THE SOFTWARE DEVELOPERS MAGAZINE

EXE

OCTOBER 1997

£3.50

Is it real or is it BeOS?

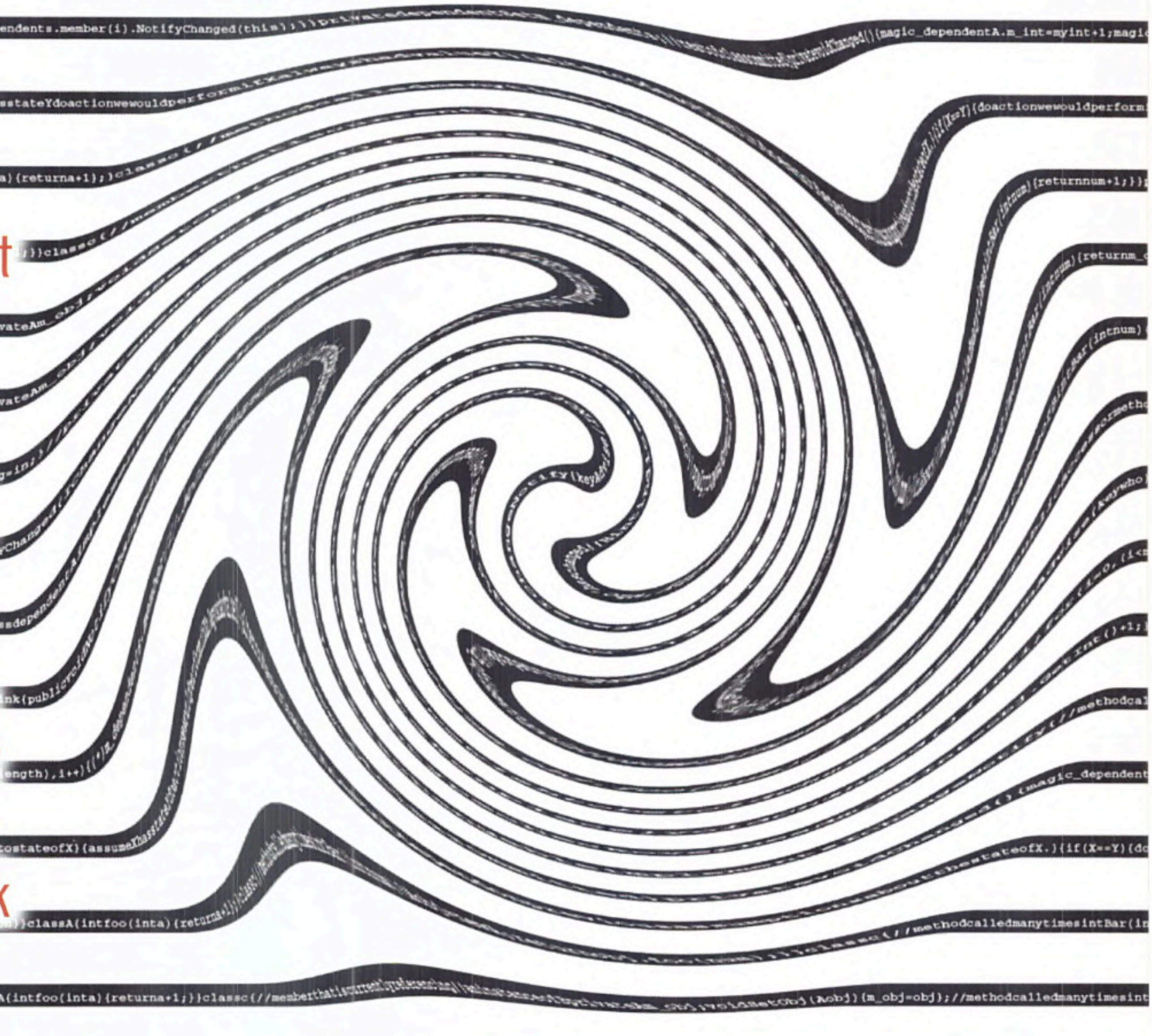
Going Windows with J/Direct

From coding to typesetting

Talking low-level to Windows NT

Knuth's back

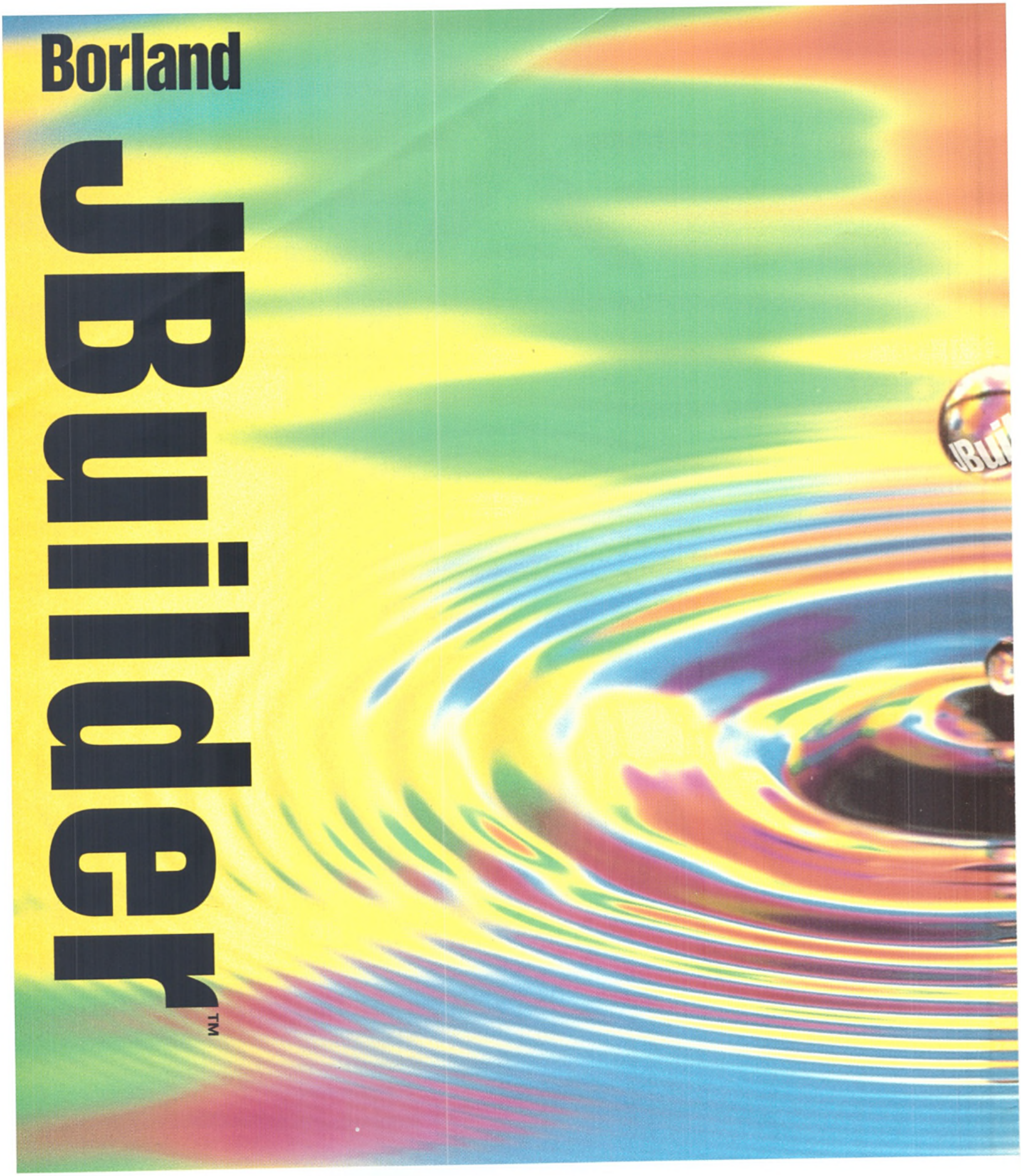
Basic Constituents for ActiveX

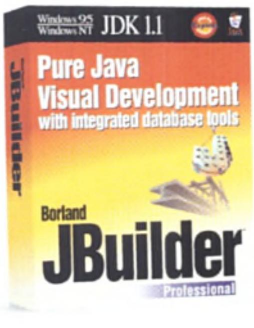


Self-modifying code is good for you

10

9 770268 687015





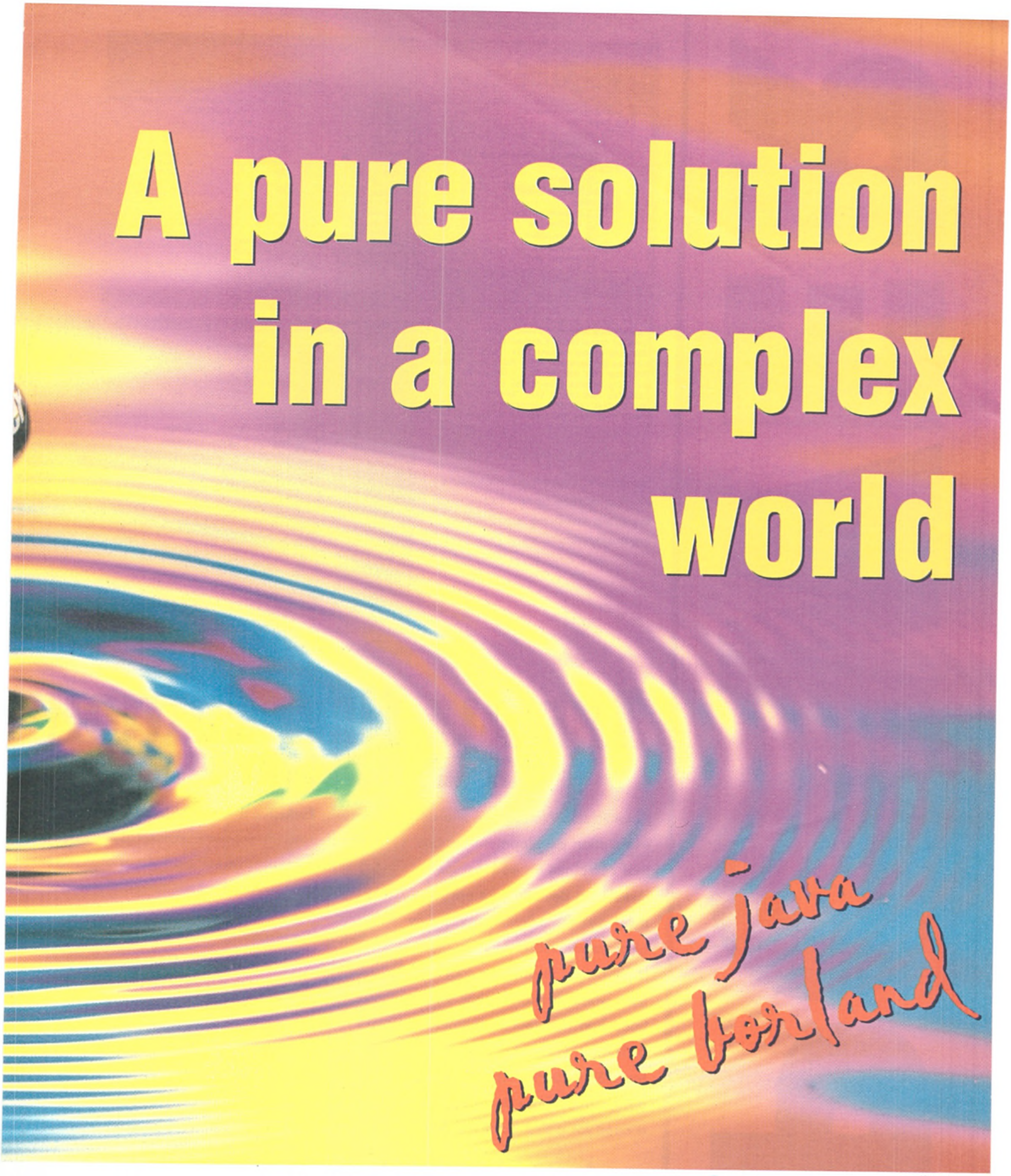
With JBuilder, Java's in its Element

Creating platform independent Java applications has never been quicker or easier. JBuilder combines the productivity of Borland's renowned RAD technology with industry standard Pure Java™.

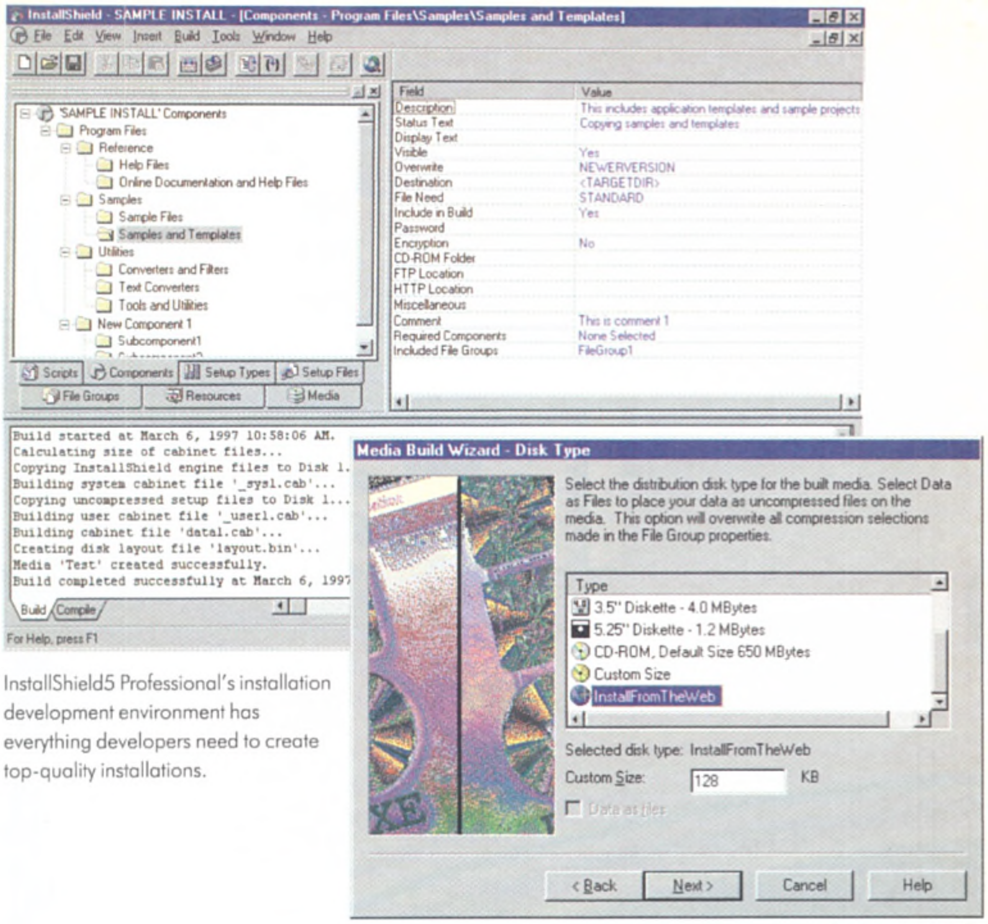
Only Pure Java code is truly platform-independent. That's why Borland collaborated with Sun Microsystems, the creators of Java, to equip JBuilder™ with the latest Sun JDK 1.1.3 architecture as its development platform, guaranteeing that JBuilder will create 100% Pure Java source code.

Now Java's Really Making Waves

For true productivity, JBuilder *Professional* includes over 80 JavaBeans, with full support for JDBC


A pure solution in a complex world
pure java
pure borland

POWER



'SAMPLE INSTALL' Components
- Program Files
  - Reference
    - Help Files
    - Online Documentation and Help Files
  - Samples
    - Sample Files
    - Samples and Templates
  - Utilities
    - Converters and Filters
    - Text Converters
    - Tools and Utilities
  - New Component 1
    - Subcomponent1

Field	Value
Description	This includes application templates and sample projects
Status Text	Copying samples and templates
Display Text	
Visible	Yes
Overwrite	NEWERVERSION
Destination	<TARGETDIR>
File Need	STANDARD
Include in Build	Yes
Password	
Encryption	No
CD-ROM Folder	
FTP Location	
HTTP Location	
Miscellaneous	
Comment	This is comment 1
Required Components	None Selected
Included File Groups	FileGroup1

```
Build started at March 6, 1997 10:58:06 AM.
Calculating size of cabinet files...
Copying InstallShield engine files to Disk 1.
Building system cabinet file '_sys1.cab'...
Copying uncompressed setup files to Disk 1...
Building user cabinet file '_user1.cab'...
Building cabinet file 'data1.cab'...
Creating disk layout file 'layout.bin'...
Media 'Test' created successfully.
Build completed successfully at March 6, 1997
```

Select the distribution disk type for the built media. Select Data as Files to place your data as uncompressed files on the media. This option will overwrite all compression selections made in the File Group properties.

Type
- 3.5" Diskette - 4.0 MBytes
- 5.25" Diskette - 1.2 MBytes
- CD-ROM, Default Size 650 MBytes
- Custom Size
- InstallFromTheWeb

Selected disk type: InstallFromTheWeb

Custom Size: 128 KB

□ Data as files

InstallShield5 Professional's installation development environment has everything developers need to create top-quality installations.

Media Build Wizard determines the target media for your final build.

THE ALL NEW EDITION
OF AN INDUSTRY STANDARD

InstallShield5 Professional is a new installation development system that combines an easy-to-use installation development environment with the reliability and power of proven InstallShield technology. It's the only installation toolkit from which you can author, layout, build and test your installation. Built on a world of experience, InstallShield5 Professional was designed for those developers with a need for a first-class installer.

Features Exclusive to InstallShield5 Professional

- Installation visual development environment contains all tools
- Project Wizard establishes framework of installation project
- Project Workspace Panes store and manage all properties
- Color Script Editor visually organizes installation elements
- Function Wizard auto-inserts complete function statements
- Compiler message window displays errors during compile
- Visual C++-like interface provides standard look and feel
- Multi-tiered file layout system gives maximum flexibility
- Drag-and-drop file layout allows for easy file organization
- Components depict users' view of application organization
- Quick build option for testing; full build option for final
- End-user installation interface supports multimedia files
- AVI video, WAV/MIDI sound, 256-color BMP images

InstallShield Software Corporation

www.installshield.com

InstallShield5
PROFESSIONAL EDITION

Grey Matter	01364 654100	PTS Direct	01928 579900
Highlander	0181 316 5001	QBS	0181 9568000
Rogue Wave	01734 880224	System Science	0171 8331022
		Systemstar	01707 278300

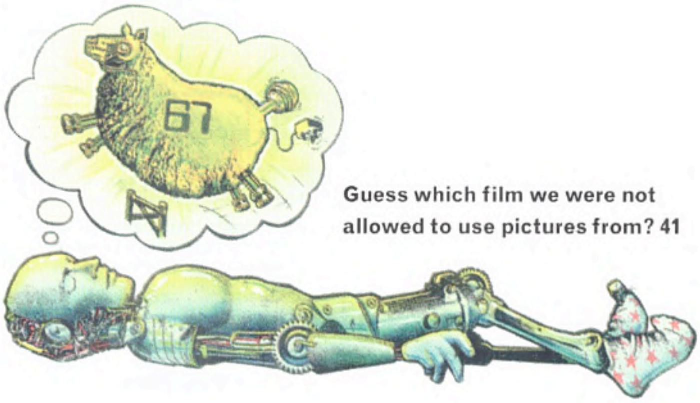
InstallShield is a registered trademark of InstallShield Corporation.

ENQUIRY NO. 02

From high-level to low-level 25

Java, Java.... west of Krakatoa 63

Programming with case and types 31


Guess which film we were not allowed to use pictures from? 41

Editor: David Mery **Features Editor:** Neil Hewitt **Contributing Editor:** Will Watts **Production Manager:** Kate Adams **Advertisement Manager:** Mark Parker **Display Sales Executive:** Ian Sinclair **Office Administrator:** Jacqui Ramrayka **Group Art Director:** Colin McHenry **Publisher:** Declan Gough

EXE: The Software Developers' Magazine is independent and not affiliated to any vendor of hardware, software or services. It is published by: **Centaur Communications Ltd, St Giles House, 50 Poland Street, London W1V 4AX**. **EXE Advertising/Editorial/Production Telephone:** 0171 287 5000 **Fax:** 0171 437 1350 **Advertising email** markp@dotexe.demon.co.uk **Subscriptions** Tel: 0171 292 3706 Fax: 0171 439 0110 email: execirc@centaur.co.uk . EXE is available by subscription at £35 per annum (12 issues) in the UK: see subs card within this issue. The magazine is published around the 1st of the month. To subscribe or if you have a subscription query, please call 0171 439 4222 or write to The Subscriptions Manager, EXE, (address above). We can invoice your company if an official company order is provided. Back issues are available at £3.50 each. 'A Subscription implies that this journal will be sent to the subscriber until one of the three expires' (AG Macdonell.) **Editorial.** Address all editorial enquiries and comments to The Editor, EXE, (address above) or email to editorial@dotexe.demon.co.uk. We welcome letters, opinions, suggestions and articles from readers. These may be edited. Information contained in EXE is believed to be correct. If errors are found, we will endeavour to publish a clarification in the next issue.

Repro & Typesetting: Atelier Dataset Ltd **Printer:** St Ives (Roche) Ltd. **Front Cover Illustration:** After Bridget Riley **ISSN:0268-6872**

EXE ONLINE
http://www.exe.co.uk

Generic OCXs & DLLs

Bar Code

Bar Code Pro	£149
d-Barcode OLE Developers Kit	£145

Charts & Graphs

Chart FX Enterprise 3.0(16&32)	£247
First Impression OCX 2.1	£204
Graphics Server 5.0	£235
Olectra Chart	£247

Comms

CommTools/DLL	£215
Greenleaf CommLib 5.2 (DLL)	£235
Sax Comm Objects Pro	£345

Data Grids & Spreadsheets

DataTable	£135
Formula One OCX 4.0	£204
Spread 2.5	£212

Database

ActiveX Component Suite	£243
Classic Data Control for Btrieve	£175
CodeBase 6.2 (Unlimited-Client)	£295
VBtrv Toolbox Controls 4.1	£209

Graphics

Fastgraph for Windows	£189
GeoPoint	£204
InterAct	£275
TWAINWizard	£89

Images

ImageGear Std ActiveX 6.0	£1250
ImageMan ActiveX Suite 5.0	£322
LEADTOOLS ActiveX16/32 Pro	£375
VectorFX	£229
Victor Image Proc for Win32	£390

Internet

Distinct Visual Internet Toolkit 2.0	£249
dsSocket OCX 1.6	£68
Sax Webster OCX (32-bit)	£99
SocketTools	£175

Multi-Function

OLETools	£163
Stamina	£141
WinX Component Library	£135

Reports

Crystal Reports Pro 5.0	£285
Report FX Enterprise	£240

Spell Checkers

Sentry Spell-Checker Engine	£235
VisualSpeller OCX 2.0	£102

Special Purpose

Calendar Widgets	£99
Eval-O-Matic Suite	£188
Input Pro	£109
Schedule/OCX	£177

Text Editors

ALLText HT/Pro OCX	£305
HighEdit Pro	£399
TX Text-Control Collection OCX	£267

GREY MATTER

Prigg Meadow, Ashburton
Devon TQ13 7DF

Prices do not include VAT or other local taxes but do include delivery in mainland UK. Please check prices at time of order as ads are prepared some weeks before publication. This page lists some products – call us for a complete price list. ORDER BY PHONE WITH YOUR CREDIT CARD

(01364) 654100

FAX: (01364) 654200
www.GreyMatter.co.uk
maildesk@GreyMatter.co.uk

DATABASE

CA-Visual Objects 2.0 Std	£82
Clarion for Windows Pro	£320
Omnis 7 Workgroup Starter	£399
Paradox (Win32) 7.0	£92
Personal Oracle for Win95	£305
PowerBuilder Desktop 5.0	£175
Sybase SQL Anywhere 5.5	£177
Visual dBase	£405
Visual FoxPro Professional 5.0	£403

COBOL

VisualAge COBOL Std OS/2	£605
VisualAge COBOL Pro OS/2	£1265

PROGRAMMER'S TOOLS

CASE Tools

EasyCASE Pro	£694
ERwin/Desktop	£395
ERwin/ERX	£2784
With Class Enterprise 3.0	£280

Help Tools

AnswerWorks	£515
Doc-to-Help 2.5(Win32)	£293
Documentation Studio	£499
Help Magician Pro 95	£207
HelpSite	£196
HyperText Suite	£292
NextHelp	£131
RoboHELP 4.0	£304
Visual Help Pro 3.1	£156
WinHelp Office 4.0	£446

Installers

Eschalon Setup Desktop	£175
InstallSHIELD Express 2.0	£174
InstallShield5 Professional	£565
PC-Install (Win32 & Win16)	£150
Wise Installation Enterprise	£422

Macro Languages

BasicScript VBX	£390
Cypress Enable (32- & 16-bit)	£741
Sax Basic Engine Pro	£345

Shells

MKS Toolkit with preferred subs	£310
Thompson Toolkit (Win&DOS)	£112

Version Control

MKS Source Integrity (with sub)	£465
PVCS Version Manager 5.2	£420
SMS	£445
TLIB 5.04	£157
Visual SourceSafe 5.0	£399

C & C++ FOR WINDOWS

Charts & Graphs

3D++ Class Library	£290
GraphiC/Win 7.0	£360
Real-Time Graphics Tools	£300

Comms

COMM-DRV/LIB 16.1	£124
Fax C++ SDK for Win16/95	£905

Compression

Compression Plus	£202
Crusher! Win32	£225
Greenleaf ArchiveLib 2.1	£210
PKWare Data Comp Lib Win32	£225

Database

DISAM96 for Win95	£715
HyperView++	£999
ProtoGen+ Client/Server Win	£1370
Raima Engine + EADS 4.0	£740
Report Writer Visual Coder	£75
Velocis + EADS (Offer)	£250
Visual SQL	£958

Images - Bitmap

Ad Oculos 3.0 Win32 & Win16	£475
Image SDK Plus for NT	£595

Maths & Stats

IMSL C Numerical Lib w/Sub	£650
Math.h++ (with support)	£803
Linpack.h++ (with support)	£803

System Level

WinDK	£487
WinDriver	£570
WinRT Developer's ToolKit	£242
WinSCSI-32	£129

Tools

BoundsChecker C++Builder	£310
BoundsChecker Visual C++ 5.0	£434
C-Cover	£557
CC-Rider 6.0 (Win32)	£326
Exemplar Setup Toolkit Pro	£130
Thales (MS Basic to C/C++)	£2760

DELPHI

Delphi 3 Standard	£81
Delphi 3 Professional	£382
Delphi 3 Client/Server	£1377
Abbrevia	£117
ABC for Delphi	£114
Async Pro 2.1 for Delphi	£110
BoundsChecker 5.0 for Delphi	£310
Charting Tools for Win - Delphi	£164
Component Create	£162
Eschalon Power Controls 2.0	£135
Essentials	£43
Helping Hand 3.0	£104
HyperTERP/Pro	£320
ImageLib Win32&Win16	£109
InfoPower 3.0	£164
InnoView MultiLanguage Std 2	£144
List & Labels for Delphi 4.0	£295
Mobius Draw Kit	£99
Mobius FastSprites	£105
Orpheus 2.1 (32 & 16-bit)	£110
Pumpkin Project Manager 2.0	£86
SysTools	£98
Transform: Component Expert	£103
VB2D Standard	£164
VisualPROS 1.1	£125
WinG Sprite Kit	£99

SMALLTALK

IBM Smalltalk for Win	£515
VisualAge for Smalltalk Base	£3197

JAVA

Jamba 2.0	£199
Java WorkShop	£75
JBuilder Standard	£85
JBuilder Professional	£399
JDesignerPro	£623
Parts for Java	£70
SuperCede Java	£99
SuperCede Java/ActiveX	£169
SuperCede Database	£455
Visual Café	£145
Visual Café Pro	£395
Visual J++ Professional 1.1	£70
VisualAge for Java Professional	£66
VisualAge for Java Enterprise	£1250

VISUAL BASIC ADD-ONS

3d Graphics Tools 5 VB4 (32bit)	£134
ActiveThreed	£99
Applet Designer Professional	£332
ClassAction (VB4)	£105
ClassAssist	£173
CodeReview Pro	£276
Communications Library 3.5	£162
Crescent Internet ToolPak 3.01	£132
Designer Widgets 2.0	£99
ERwin/Desktop for VB4	£395
FailSafe Pro	£259
Fax Plus 2.0	£202
Helping Hand 3.0	£104
Into Code (VB4)	£55
List Pro	£173
PDQComm 3.21	£124
Developer's Suite NetWare 5.0	£362
List & Labels for VB3/4 4.0	£295
QuickPak VB/J++	£150
Spyworks Pro Subscription 5.0	£187
True DBGrid Pro 5.0	£195
VB Assist 5	£130
VB Language Manager Pro	£115
VBTools	£99
Visual Expert Developer	£250

VISUAL BASIC 5

Visual Studio 97 includes the equivalent edition of Visual Basic 5 as well as Visual C++, J++, FoxPro & InterDev

Visual Basic Enterprise 5.0	£918
Visual Basic Professional 5.0	£378
Visual Basic Learning 5.0	£70
Visual Studio 97 Enterprise	£1125
Visual Studio 97 Professional	£765

C++ COMPILERS

Borland C++ 5.0	£242
C++Builder Standard	£67
C++Builder Professional	£389
C++Builder Client/Server	£1267
Learn to Program w/C++Builder	£39
Power++ Developer 2.0	£308
Power++ Professional	£617
Power++ Enterprise	£1233
Salford C/C++ Dev Bundle	£295
Symantec C++ 7.5	£75
Visual C++ 5.0 Learning Edition	£72
Visual C++ 5.0 Professional	£377
Visual C++ 5.0 Enterprise	£919
VisualAge for C++ 3.5	£309
Watcom C/C++ 11.0	£206

News & Views

JDESIGNERPRO

The Fastest Way to Build Bean-Based Business Apps in Java?

JDesignerPro from BulletProof is probably the most advanced tool for building and deploying single- & multi-tier database systems in Java. JDesignerPro is a truly visual enviroment for building elaborate and elegant systems without having to write a single line of code. Complex relationships are displayed as simple visual metaphors - database programming has never been so much fun!

- **Tab Menu System** ties the menus to the access control mechanism and allows instant deployment - just assign your new module to a menu location and it is instantly accessible by users
- **Interaction Manager** lets you link any object to another and specify action events, completely visually
- **Instance Manager** let you import any Java component, including Beans, and creates a live instance at design time
- **Layout Manager** is 100% Pure Java, supports Java Beans and leverages the enhancements in the latest JDK
- **Generates SQL** for inserts, updates & deletes as the fields are created. You don't even type in the field names!
- **Project Management** - just install JDesignerPro on a single server, then run it remotely through a browser. It tracks the screen designs for any number of developers, and they can reload & modify their projects anywhere
- **Generates SQL** for inserts, updates & deletes as the fields are created. You don't even type in the field names!
- **Powerful Grid Class** for viewing data tables with spreadsheet functions
- **Filtering, Sorting & Searching** classes give you immediate power for browsing
- **Java Charting** with a number of classic chart styles to look at your data
- **Build E-Mail Windows** on any screen
- **JAGGServer** is BulletProof's Java database engine that allows highly scalable, very fast database access for Java clients. You can also program into JAGGServer to add logic and rules. It also handles the delivery of Java client screens to the user as they are requested

Check our web site for full details on this exciting product - £623 + VAT

HYPERVIEW++

Add a GUI Builder to Your MFC Apps

HyperView++ extends MFC with fully customisable GUI builder and form/dialog/view editor classes that you can embed in your applications. Easily reprogram the editor, its objects and controls, its attached actions, etc., to suit your needs. Now with VBScript (Cypress Enable), ODBC and drop-in ActiveX support! HyperView++ is a great alternative to VBA 5.0 for MFC developers.

- OLE/ActiveX Support - event handling, browsing of methods and properties
- OLE Automation of forms, objects, properties, variables, etc
- Compilable as MFC Extension DLL or ActiveX Control
- Built-in VB compatible Basic Scripting engine for views
- ODBC/DAO connectivity, all other standards can also be used
- Editor supports Win, ActiveX & custom controls, lightweight GUI objects, etc
- Full Featured Editor supports spacing, undo, multi-select, etc

You get full source code, and pay no run time royalties. Unusually, HyperView++ is licensed at £999 + VAT per project, not per developer.

DEBUGGING TOOLS

BoundsChecker Standard 5.0	£259
GreatCircle Standard	£268
MemCheck	£205
Soft-ICE/Win95 3.0	£372

EDITORS

Codewright Prof 5.0	£180
Kedit for Windows	£129
Multi-Edit for Windows 7.1	£137
Visual SlickEdit (Win) 2.0	£205

✉ ENQUIRY NO. 03

The Object is dead, long live the Component

Anderson Consulting confirmed it in May this year and The Butler Group has now followed suit – the future of software development architecture lies in components. Object technology has become too much of a religion, too rigid and prescriptive, dictating how every last bit of code should be written.

Today's object based software development could be compared to building piece by piece with individual bricks, while the component approach combines much larger chunks of software as the building blocks – like constructing a building from pre-fabricated blocks. The benefits of this approach can be seen from a software development angle that also makes pragmatic business sense.

Components can be viewed as groups or bundles of objects defined largely by the end application for which they are designed. In practice, components can be employed to provide a standard GUI screen interface to secure the user's password and grant entry to a defined list of other applications – a fairly routine application. Alternatively, components can be employed within a database to undertake a series of SQL queries and distribute the analysis across a range of platforms throughout the organisation. In essence, organisations can select from a combination of ready made 'off-the-shelf' components, for routine tasks such as GUI user entry and verification, and bespoke components for the more complex, distributed tasks such as SQL data mining or Web deployment.

Object-based programming has not enjoyed widespread success because it is too fine-grained and because, in many respects the tools are too technical. The evolution of second-generation client/server tools and development of 3 and 4GLs brought the complex nature of the object-oriented approach into focus.

Organisation-wide software development becomes an issue of component integration.

Client/server tools such as NatStar, SuperNova's Application Developer, Forte and Dynasty, complete with application partitioning may have addressed some of the software integration issues brought up by their predecessors. But what they gained in power and flexibility they certainly sacrificed in terms of ease of use. In reality, to a generation of programmers brought up on Cobol and C, second-generation client/server tools were a bridge too far.

Faced with the prospect of having to retrain teams of programmers in an increasingly diverse set of tools and languages ranging from legacy Cobol to the latest Java, component-based architecture provides a solution to placate both the IS and financial departments.

In an ideal world, programmers would select a combination of off-the-shelf and bespoke components generated from a variety of tools determined only by the application and the programming skills at their disposal. Visual Basic for Windows-based applications and Java for Web deployment, for example. Rather than being driven entirely by the nature of the legacy system or toolset, programmers could select 'best of breed' tools to build components for deployment. Organisation-wide software development becomes an issue of component integration. The shift away from the rather granular concept of individual objects towards components certainly represents one step towards addressing the issue of global integration, that is, the universal ability to combine components generated by any and all tools.

Talk of development tool 'standards', however, is likely to remain just that. The ActiveX/JavaBeans debate highlights the ambiguity regarding unified standards. The ActiveX marketplace for components has been valued at over £180 million, so market forces combined with Microsoft's well-documented muscle should ensure its proliferation – but where does this leave JavaBeans, also supported by Microsoft?

In common with Anderson Consulting, The Butler Group supports a third solution besides purchasing off-the-shelf applications or developing systems from scratch – the use of re-useable, modifiable software components. Future software applications will be assembled from existing components held in a repository, providing access for developers to as wide a range of components as possible. The future lies in *component glue* software. According to the Butler Group, such glue would have essential properties: the ability to create applications without writing code; the ability to combine components developed with various tools into one application; strong support for industry standards; technology independence; and integration of legacy systems.

Faced with the necessity for an increasing variety of applications to be distributed across a growing array of platforms, the concept of object-based development is no longer valid. In today's environment, the OO approach is too fine-grained and, in many cases, the tools are too technical. Today's programmers are looking to select best of breed client/server tools for the job in hand. Provided that components can be easily integrated, programmers can then effectively select tools simply on the basis of the application itself and the programming skills at their disposal.

Component glue, therefore, remains the key technology enabling applications to be built by mixing and matching software built with different tools. Only then will developers be able to concentrate on the business requirements without being constrained by the technology.

David Barkaway
Technical Manager
SuperNova
(www.supernova.com)

WHY PAY MORE?

FOR THE BEST PRICES & SERVICE IN EUROPE CALL COMPONENTSOURCE

PRODUCT DESCRIPTION	PRICE*
3D Modelling Tools	
Visual 3Space Control V2.4 (32Bit ActiveX)	£189.00
ActiveX Control Creation	
Basic Constituents V1.0 (32Bit)	£89.00
Artificial Intelligence	
Amzi! Prolog+ Server Personal V4.0 (32Bit)	£140.00
Barcode Tools	
C39 Tools V3.0 Single User Licence	£70.00
dBarcode OLE Developers Kit V3.5 (16&32Bit DLL/OCX)	£105.00
dBarcode -2D Developers Ed. V1.0 (32Bit OCX/DLL)	£225.00
dFont Barcode Fonts for Windows - from	£50.00
UPC Tools V3.0 Single User Licence	£70.00
Button & Cursor Design	
FarPoint ButtonMaker V1.0a (16&32Bit ActiveX/DLL/VBX)	£45.00
Calendar Controls	
Calendar Widgets V1.00c (16&32Bit ActiveX/OCX)	£79.00
iCalendar V1.0 (32Bit ActiveX)	£39.00
Code Tools	
Code.Print Pro. for Delphi V3.01 (16&32Bit Delphi 1/2)	£60.00
Code.Print Pro. for VB V3.02 (16&32Bit VB3/4/5)	£60.00
NuMega CodeReview Pro. V5.0 (16&32Bit)	£199.00
NuMega VB Dev. Suite (CodeReview Pro./FailSafe Pro.)	£299.00
Pretty Printer V5.0 (16&32Bit VB3/4/5)	£49.00
Communications - Libraries	
App-Link V2.0 RADX Control Set (16&32Bit ActiveX/VBX)	£80.00
Crescent PDQComm V3.21 (16&32Bit ActiveX VB5 Ready)	£95.00
SuperCom V3.5 (32Bit W95/WNT)	£205.00
Communications - Message Brokering	
Active Gateway SDK V1.3 (32Bit ActiveX/DLL)	£175.00
Communications - NetWare	
Developers Suite for Netware V5.0 (16&32Bit)	£235.00
Compression Tools	
Crusher! Data Comp. Toolkit V3.12 (32Bit DLL)	£165.00
Data Entry Controls	
FarPoint Input Pro. V2.0 (16&32Bit ActiveX/DLL/VBX)	£69.00
ProtoView WinX Component Library V1.0 (32Bit ActiveX)	£105.00
Data Import & Export Tools	
OpenExchange Dev.Kit V1.0x (16&32Bit OCX/VBX)	£270.00
Data Modelling	
Logic Works ERwin/ERX V3.0 (16&32Bit)	£2,430.00
Database Tools	
Total Access Agent V1.03 (Access2/Access95)	£95.00
VideoSoft VSDATA V1.0 (32Bit ActiveX)	£115.00
Debugging & Testing Tools	
MemCheck for Windows V5.0 (Win 95/NT)	£125.00
NuMega BoundsChecker V5.0 VC++ (W95/NT)	£330.00
NuMega BoundsChecker V5.0 Delphi (W95/NT)	£235.00
Vermont High Test Plus V3.0 (16&32Bit Win 3.1.1/95/NT)	£140.00
Development Tools	
Leszynski Naming Conventions (LNC)(16&32Bit)	£25.00
Diagramming Controls	
AddFlow ActiveX Control V1.0.1 (32Bit ActiveX)	£69.00
MetaDraw V1.0 (16&32Bit OCX)	£145.00
ProtoView InterAct V1.5 (16&32Bit ActiveX/OCX)	£215.00
DirectX Controls	
iDraw V1.0 (32Bit ActiveX)	£99.00
iSound V1.0 (32Bit ActiveX)	£99.00
VideoSoft VSDIRECT V1.0 (32Bit ActiveX)	£110.00
Error Handling	
NuMega FailSafe Pro. V5.0 (16&32Bit)	£199.00
Explorer	
ProtoView Data Explorer V1.0 (32Bit ActiveX)	£105.00
Facsimile Tools & Controls	
FaxMan V2.11 (16&32Bit DLL/OCX/VBX)+ 10,000 runtime	£260.00
Find & Replace Utilities	
SPEED Ferret for Access Bundle 97/95/2.0 (16&32Bit)	£70.00
Graphic Filters	
DXF-IN for VB V2.52 (16&32Bit VB3/4)	£95.00
Graphing Tools & Controls	
Graphics Server V5.0 (16&32Bit ActiveX/DLL/VBX)	£170.00
Olectra Chart V5.0 (16&32Bit ActiveX/DLL)	£140.00
Pro. Essentials V2.0 (16&32Bit ActiveX/DLL/VBX)	£190.00
Grid Controls	
APEX True DBGrid Pro. V5.0 (16&32Bit ActiveX) VB4/5 upgrade	£109.00
Data Widgets V2.0b (16&32Bit ActiveX/OCX)	£79.00
Objective Grid V5.0 for MFC (16&32Bit)	£199.00
ProtoView DataTable V3.1 (16&32Bit ActiveX/OCX)	£105.00
VideoSoft VSFLEX V2.0 (16&32Bit ActiveX)	£90.00
Help Authoring	
ForeHelp V2.96 (32Bit Win95/NT)	£185.00
ForeVB V2.0 (Add-In for VB5 32Bit)	£45.00
Helping Hand V3.11 (32Bit)	£79.00
SOS Help! Info Author V2.3 (Win3.1/95/NT3.51)	£160.00
VideoSoft VSDOCX V1.0 (32Bit)	£145.00
Imaging	
ImageFX PhotoPRO V1.0 (32Bit ActiveX)	£165.00
ImageMan ActiveX Suite V5.04 (16&32Bit ActiveX/OCX)	£255.00
iTWAIN OCX V1.1 (32Bit OCX)	£99.00
LEADTOOLS ActiveX32 Pro. V8.0 (32Bit ActiveX)	£205.00
Victor Image Processing Library V4.11 (32Bit DLL)	£235.00
Installation	
Exemplar Setup Toolkit Pro. Ed. for VC++ 5.0 V1.3 (32Bit)	£95.00
InstallShield Express V2.0 (32Bit)	£165.00
PC-Install for Windows Pro. V5.0 (16&32Bit)	£230.00
Setup Factory V4.0 (32Bit)	£99.00
WISE V5.0 Enterprise Edition (16&32Bit)	£279.00
WISE Installation System V5.0f (16&32Bit)	£99.00
Instrumentation Controls	
Instrumentation ActiveX Lib. V3.01 (32Bit ActiveX)	£135.00
RSAnimator V1.0 (32Bit ActiveX)	£99.00
RSToolPak I V1.00.19 (32Bit ActiveX)	£95.00
RSToolPak II V1.00.19 (32Bit ActiveX)	£119.00
ISAM	
VB/ISAM 4032 for VB (16&32Bit DLL)	£99.00
List Controls	
FarPoint List Pro. V2.0 (16&32Bit ActiveX/DLL/VBX)	£115.00
Lotus Notes Integration	
VBLink for Lotus Notes V4.2 (16&32Bit ActiveX/OCX)	£545.00
Mapping & GIS Tools	
GeoView LT V5.0 (32Bit ActiveX) - Royalty free	£139.00
MapInfo MapX SDK V2.0 (32Bit OCX) includes1 Run-Time	£895.00
Maths, Stats and Financial	
Eval-O-Matic Suite V2.0 (32Bit ActiveX/VCL)	£115.00
Multi-Media	
ImageFX FXPic/VBX V4.0 (16&32Bit ActiveX/VBX)	£175.00
ImageFX FXTools Pro. V4.0 (16&32Bit ActiveX/VBX)	£199.00
O-O & Class Tools	
ClassAction V1.0 (16&32Bit)	£65.00
ClassAssist V1.0 (32Bit)	£155.00
On-Line Analytical Processing (OLAP)	
DynamiCube V2.0 (32Bit ActiveX)	£235.00
Outline Controls	
The List Collection V3.0 (16&32Bit OCX/VBX)	£115.00
TList Pro. V3.0 (16&32Bit ActiveX/OCX)	£99.00
Postcodes & Addressing	
Postcode Plus V2.0 (16&32Bit DLL)	£450.00
Printing Tools	
VideoSoft VSVIEW V2.0 (16&32Bit ActiveX/OCX)	£90.00
Productivity Controls	
Kwery Control Pak 1 V1.1 (32Bit ActiveX for Access 95)	£75.00
Mabry Mega Pack 2 (16&32Bit OCX/VBX)	£95.00
OLETools V5.1 (32Bit ActiveX)	£109.00
Crescent QuickPak VB/J++ V1.0 (16&32Bit ActiveX/DLL)	£120.00
Vantage Control Set V2.0 (16&32Bit ActiveX/VBX)	£75.00
VBTools V5.01 (16Bit VBX)	£59.00
Productivity Tools	
VBAssist V5.0 (32Bit VB5.0)	£115.00
Report Writers	
Crystal Reports Pro. Upgrade V5.0 (16&32Bit)	£119.00
VideoSoft VSREPORTS V1.0 (32Bit ActiveX)	£90.00
Software Licensing Tools	
Protection Plus Pro. V4.04 Enterprise (16&32Bit DLL)	£240.00
softSENTRY V1.11 Full Version (16&32Bit)	£275.00
Sound & MIDI Controls	
Mabry MIDI Pack (16&32Bit OCX/VBX)	£55.00
Spell Checking	
Sentry Spelling-Checker Engine V4.21 (16&32Bit DLL)	£140.00
VideoSoft VSSPELL V1.0 (16&32Bit ActiveX/VBX)	£90.00
Spreadsheet & Data Entry	
FarPoint Spread V2.5 (16&32Bit ActiveX/DLL/VBX)	£150.00
SQL Tools	
sp_Assist V1.1 (32Bit)	£370.00
Structured Storage Tools	
StorageTools V1.0 (16&32Bit ActiveX)	£65.00
Tab Controls	
FarPoint Tab Pro. V2.0 (16&32Bit ActiveX/DLL/VBX)	£60.00
Your Basic Tab V2.0 (16&32Bit VB3/4/5)	£79.00
Telephony & TAPI Products	
VBVoice for TAPI V2.0 (32Bit OCX)	£220.00
Visual Voice for TAPI 2-Line V2.0 (32Bit ActiveX)	£285.00
Text Controls	
HighEdit Pro. V5.2 English Ed. (16&32Bit DLL/OCX/VCL)	£310.00
TX Text Control V5.01 (16&32Bit ActiveX/DLL)	£205.00
ALL Text HT/Pro. V4.0 (16&32Bit OCX)	£195.00
Timers	
ExacTicks V1.1 (16&32Bit DLL)	£75.00
User Interface	
AceToolbar V1.2 (16&32Bit ActiveX)	£45.00
ActiveBar V1.019 (32Bit ActiveX)	£70.00
ActiveThreed V2.01 (32Bit ActiveX)	£79.00
Designer Widgets V2.02 (16&32Bit ActiveX/OCX)	£79.00
Eidesys SPLIT Suite '97 V2.0 (16&32Bit ActiveX)	£49.00
ReSize V2.5 (16&32Bit ActiveX/OCX)	£25.00
VideoSoft VS-OCX V5.0 (16&32Bit ActiveX/OCX)	£60.00
Version Control	
GP-Version V2.11 (16&32Bit) Single User Licence	£60.00
Versions 2.0c (32Bit)	£50.00
VersionStamper V5.0 (16&32Bit ActiveX/VBX)	£75.00
Windows API Tools	
SpyWorks Pro. V5.0 Incl. Subs. (16&32Bit ActiveX)	£130.00
WinAPI oblets V1.0a (32Bit)	£89.00
WWW & Internet Tools	
Astra Site Manager V1.01 (Win95/NT)	£235.00
Crescent Internet ToolPak V3.01 (16&32Bit ActiveX)	£95.00
Distinct TCP/IP SDK V2.0 (32Bit DLL)	£290.00
Distinct Visual Internet Toolkit V2.0b (32Bit ActiveX)	£175.00
IdleY ISAPI V2.0 (32Bit ActiveX/DLL)	£119.00
IDSMail OLE Server V4.1 (Win 3.1/95/NT)	£235.00
iPOP3 V1.0 (32Bit ActiveX)	£39.00
iSMTP V1.0 (32Bit ActiveX)	£39.00
J-BOTS V1.0 (32Bit for Microsoft FrontPage 97)	£20.00
Mabry Internet Pack (16&32Bit ActiveX/VBX)	£70.00
NobleNet Web V1.5 (16&32Bit)	£565.00
PowerTCP Internet Toolkit for C/C++ V3.0 (16&32Bit DLL)	£280.00
PowerTCP Internet Toolkit for Delphi V3.0 (16&32Bit)	£280.00
PowerTCP Int. Toolkit for VB4/5 V3.02 (16&32Bit ActiveX)	£280.00
X-Works (WebLink) for VB/VFP V2.0	£165.00

* Prices exclude VAT

© 1997 COMPONENTSOURCE LTD. • ALL RIGHTS RESERVED

Component Source™

The Definitive Source of Software Components

Microsoft Certified Solution Provider

www.componentsource.com

+44 (0) 118 9581111

This is just a selection from our full product range. For information on all our products call us now!

To buy a product call us FREE on 0800 581111

ENQUIRY NO. 04

www.componentsource.com

Microsoft, Windows, Windows 95, Windows NT and ActiveX are registered trademarks of Microsoft Corporation. All product names and trademarks referenced are the property of their respective owners.

Can the Web run on DESL?

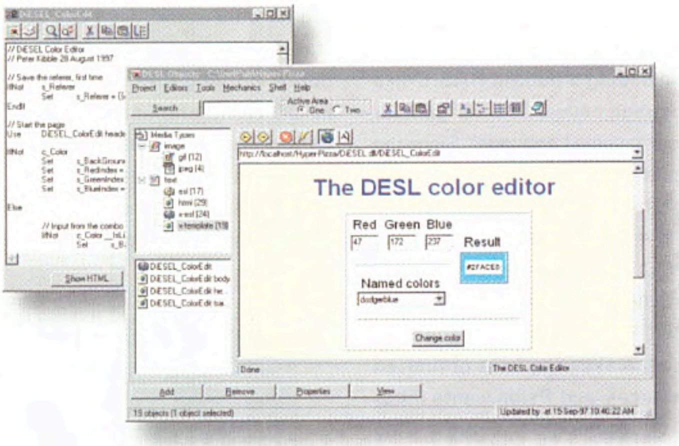
At the moment there are two methodologies in Web development; the visual and the programmatic. The visual method, exemplified by tools such as FrontPage and HotMeTaL Pro, takes a WYSIWYG approach. The programmatic method developed out of 'Visual Notepad' – the joke name given to the tendency of many Webmasters to write all the HTML code by hand. The nearest thing to a traditional code-based development tool for Web applications has been Microsoft's Visual InterDev, which in its current release still leaves a great deal to be desired – especially as it has no direct support for server-side scripting or debugging.

DESL (pronounced Diesel) from UK Internet consultants Q4, is designed to be the first Web development tool aimed directly at traditional codeheads. The product consists of an interpreted language – itself called DESL – together with a full IDE comprising HTML and code editors, debugger, object manager, and change management facilities. The server-side environment is built with ActiveX components running on Internet Information Server on Windows NT. Web applications are built in the DESL language, which Q4 claims is easy to learn and was designed specifically for the job rather than cobbled together from existing languages, and as you would expect can interface with ODBC data sources and incorporate server-side Java applets or ActiveX controls.

DESL also maintains strict session criteria, so any attempt to spoof or change IP address during a session will be rejected. EXE recently saw a demonstration of the product in beta, and we were impressed.

The following licences will be available: DESL engine only (for runtime purposes) priced at £500 for unlimited useage, Standard Edition priced at £750 including 3 developer licences, Professional Edition priced at £1250 which adds ODBC/RDO capabilities and includes 5 developer licences, and the Enterprise Edition which adds support for Unix Web servers and includes 10 developer licences. DESL is scheduled to ship in November.

t 0181 317 3733 w www.q4.com



```
// DiESEL Color Editor
// Peter Kibble 28 August 1997

// Save the referer, first time
IfNot   s_Referer
        Set     s_Referer = (I
EndIf

// Start the page
Use     DiESEL_ColorEdit heade

IfNot   c_Color
        Set     s_BackGroun
        Set     s_RedIndex =
        Set     s_GreenIndex
        Set     s_BlueIndex =

Else

        // Input from the combo
        IfNot   c_Color__IsLi
                Set     s_B
```

http://localhost/HyperPizza/DiESEL.dll/DiESEL_ColorEdit

The DESL color editor

Red 47 Green 172 Blue 237 Result #2FACED

Named colors: dodgeblue

Done — The DESL Color Editor

15 objects (1 object selected) — Updated by at 15-Sep-97 10:40:22 AM

ORBs for free

The Oracle and Olivetti Research Lab (ORL) has announced the availability of a 'free' CORBA ORB. In this case, it is also free in the Stallman sense as the ORB is being released under the GNU General Public Licence. ORL claims the ORB is significantly faster than the competition because of its native platform thread support.

Offering the product free may pull the rug from under competitors like Visigenic which has been making a lucrative business licensing its technology to development tool vendors.

The ORB – called OmniORB2 – is a native IIOP implementation which will be compatible with offerings from Visigenic and Iona. Versions are available for Windows 95/NT, Linux and flavours of Unix.

w www.orl.co.uk

Cold Fusion happened

Not to be confused with the unproved 'fusion in a glass' experiments of Professor Martin Fleischmann (or the rather more promising supercold fusion experiments of a Japanese team based here in the UK), Cold Fusion is a Web development tool from Allaire Software.

Like DESL (see above), Cold Fusion specialises in Web application development on the server side where client HTML pages are generated from a database rather than held as flat files. However, Cold Fusion is not aimed at codewrights – all the server-side logic is stored in templates which are incorporated into raw HTML within the visual design environment. Cold Fusion is available from UK distributors Grey Matter priced from £446.

t 01364 654100

w www.greymatter.co.uk

JBuilder is go at last

Borland's long-awaited JBuilder has finally shipped, fulfilling the promise made by CEO Del Yocam that it would do so this quarter. All versions including the client/server edition should be available from resellers as you read this.

w www.borland.com

OpenGL developers will be pleased to hear that **Silicon Graphics** has released a revised version of the OpenGL libraries (and development kit) for **Windows 95** and **NT**. MMX support is included, as is support for **DirectDraw 5.0** in 2D applications.

www.sgi.com/technology/opengl

Need to add **FTP** functions to your application? Then look no further than **Conduit**, a Windows DLL which provides high-level functions for achieving FTP transfers. **16-bit only** for now, with a **32-bit** version coming soon.

www.modmed.com/conduit

More 2D graphics with **Carnac**, a **C++** and **Java** graphics toolkit for fast 2D object-based drawing. Version 1.1 includes a full **SDK** and support for **24-bit** images and textures. One suggested application is real-time data visualisation.

www.int.com

Protect your software with **SoftSentry** from **20/20 Software**. SoftSentry is a software-only protection system for licensed or **demo** software. Claimed features include the ability to **imprint** an executable at installation so it will only run on **one** particular machine.

www.componentsource.com

Following on from the launch of **Web Integrity** (see EXE, September 1997), **MKS** has announced availability of **Mac** and **Unix** clients for the software, although the server is still NT and Unix only. See the MKS Web site for more information on **upgrades**

www.mks.com

Launch of an EPOC-making event

For years, the Psion Organiser series was unique in the PDA world in including a built-in BASIC-like language called OPL (Organiser Programming Language). Since the inception of the Series 3 it has been possible to write directly to the APIs of the Psion operating system, EPOC. The 32-bit version of EPOC on which the Series 5 is built, EPOC32, is a multi-threaded OS which makes the best of limited resources, and Psion wants third-party developers to start writing EPOC32 applications for the Series 5 and other EPOC devices which will include mobile phones and smaller task-specific PDAs.

To this end, the company has launched a Web site, EPOC World, which aims to act as a central resource for all Psion developers working on the platform. EPOC World promises the latest SDK revisions, online discussion groups, an EPOC knowledge base, downloadable code samples and tools, and training programs.

Individual developers can get access to most of the site by joining the EPOC World program; prices start at £200 per year for C++ based EPOC32 developers, and £75 for those using OPL. An extensive 'licensee-only' area will also be available for companies licensing the EPOC technology for their own products. In the meantime, you can try out the public areas of EPOCWorld for free.

Given the unprecedented demand for the Series 5 (they sold out completely within the first week and at the time of writing are on long-term back-order!) now may well be the time to get into EPOC development.

w www.software.psion.com/EPOCWorld

Embedded environment from SDS

Yet more embedded development news with the announcement by specialists Software Development Systems (SDS) of a customised version of their SingleStep development environment tailored to the Motorola's latest embedded PowerPC processors, the 750 and 740.

As well as in-circuit debugging and tracing tools, SingleStep for PowerPC includes a PowerPC 750/740 emulator for Unix and Windows 3.x/95/NT – the SingleStep Advanced PowerPC Emulator – which allows developers to test their software in a simulated environment before the first track on the PCB is etched.

SingleStep targets a variety of industry-standard realtime operating systems including Nucleus, OSE, RTEK, RTXC, and VxWorks. SingleStep for PowerPC should be available immediately.

w www.sds.com

The incredible shrinking Eiffel

The latest Eiffel release from Interactive Software Engineering (ISE) is Embedded Eiffel, a version of the language targeted at embedded systems development. Featuring the same EiffelBench development environment found on its cousins, Embedded Eiffel has been equipped with a number of features designed for the particular needs of the embedded development environment.

First of these is a tailored version of the Eiffel runtime scaled down to fit in less than 110 KB of ROM and 70 KB of RAM. The necessary libraries and tools to perform host-target debugging and development have been ported to the VxWorks embedded OS, and ISE claims that this version can be easily ported to other embedded operating systems. A scaled-down version of the Eiffel garbage collector allows for automatic collection even at the embedded level.

A special version of CECIL – the C to Eiffel Call-In Library – is supplied which allows for the inclusion of existing C/C++ source code into the Embedded Eiffel project. Tuning tools to aid per-implementation optimisation are also included.

Traditionally, the civilian embedded development market has been the province of the C, C++, or assembler programmer (things are different in military development where specialised languages are often used). However, Eiffel has made a name for itself by being both more fundamentally object-oriented than C++, and producing faster and more compact code than many C compilers. ISE clearly hope to win over this potentially lucrative slice of the development pie to its side of the argument.

Embedded Eiffel is just the latest in a series of releases from ISE aimed at popularising the Eiffel language including the low-priced Personal Eiffel and the Eiffel 4.1 point release. Information about all Eiffel versions, as well as downloadable trial and retail versions of the software, can be found on ISE's Web site. Embedded Eiffel should be shipping as you read this; no pricing information was available.

w www.eiffel.com

InstallShield has launched a **Java** version of its eponymous **installer** aimed at closing the loophole whereby Java applications which run cross-platform still need a plethora of **platform-specific** install programs. More information from *www.ptsdirect.co.uk*

Version 3.2 of **SoftICE**, the kernel-mode debugging package for Windows 95/NT from **NuMega Technologies**, features support for the current beta version of **Windows 98** and **multiprocessor** NT boxes. It finally allows SoftICE to reside in a **window** on the desktop rather than as a character mode application. *www.numega.com*

Continuing its programme to port leading embedded RTOSes to the **ARM** platform, Advanced Risc Machines has announced the availability of the **CMX-RTX** RTOS for ARM processors including the ARM 'Thumb' multimedia chip. *www.arm.com*

Oracle's Designer/2000 application development platform has entered **beta** for version 2.0. Oracle claims the **IDE** has been made more intuitive with the use of **wizards** and **drag-and-drop** functions. The beta is **not** available for download, so contact Oracle to participate in the programme. *0118 924 0000*

The latest version of **Symantec's Visual Café Pro** is available to download at the company's Web site; version **1.0e** features native support for **Informix**. A preview version of an as-yet-unnumbered **JDK 1.1-compliant** release is also available. *www.symantec.com*

They say wisdom comes with age.

Now it comes in a box!

£139

The Wise Installation System 5.0 is wisdom in a box. It is powerful, intuitive to use and completely customizable. The Wise Installation System, a Windows-based installation editor, creates professional setup programs in hours, not days. It creates a single installation that runs in Windows 3.1x, Windows NT, or Windows 95. It comes complete with both Win16 and Win32 versions, creates shortcuts/shell links for use with Windows 95, handles nested components and has a full uninstaller with rollback.

The Wise Installation System Enterprise Edition combines the award winning Wise Installation System with multi-generational patching, web based installation support, and a repackager for converting existing installations.

£420

Now shipping... **The Enterprise Edition**

Your complete answer to all your software distribution needs!

Great Lakes Business Solutions

ENQUIRY NO. 05

Deliver Fast Applications Anywhere

£1090

PowerBuilder 5.0 Professional offers ground-breaking, next-generation technology for business application development. It delivers the power you need to create high-performance, widely deployable applications. PowerBuilder now offers distributed, compiled objects, faster applications via compiled code, an Internet-ready environment, plus an object browser, full Windows 95 compliance, and OLE 2.0 support.

Powersoft
Open Tools from Sybase, Inc.

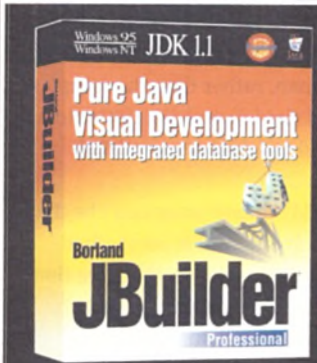
The powerful, web-enabled DBMS

Sybase SQL Anywhere Professional gives you the performance, reliability and integrity of an enterprise-wide server - at a fraction of the cost. Specifically designed for developers who need to deploy desktop and workgroup applications on client/server platforms or their corporate intranet.

from £190

Sybase
High Performance Database Servers from Sybase, Inc.

ENQUIRY NO. 06



Borland JBuilder

Out Now

"JBuilder looks set to deliver a knockout blow to existing Java tools as well as prove once and for all the viability of writing large applications in the language."
- EXE Magazine, August 1997

Professional Edition £389
QBS Special £249 upgrade price from Borland Delphi.
Proof of ownership required.

Borland
Making Development Easier

ENQUIRY NO. 07

NEW Version 2

Unlock important corporate information - *wherever it is stored*

Publish reports anywhere from anywhere

ARPEGGIO connects to any Distributed Relational Database Architecture (DRDA) or ODBC source in the enterprise.

New Key Features

- Embed charts and graphs anywhere within a report
- Point and click HTML export capabilities
- Year 2000 support
- Application Scheduler to time reports daily, weekly, monthly, quarterly and annually

ARPEGGIO Viewer - user access to pre-defined queries and reports - ***free distribution***.

ARPEGGIO for the Desktop - Data Access, Report Designer and Data Organiser modules enable users to easily create their own queries, data transfers and reports - call for pricing.

ARPEGGIO for the Developer - enables professional developers to seamlessly integrate query and report publishing capabilities into custom applications - call for pricing.

WALL DATA®

ENQUIRY NO. 08

QBS Software Limited
11 Barley Mow Passage
London W4 4PH
Phone: 0181 956 8000
Fax: 0181 956 8010
BBS: 0181 956 8011

Call now for further details and FREE software catalogue

You can now pick up detailed information, demos and evals on hundreds of developers' products from our file archive on the new qbs software ftp site: http://ftp.qbss.com. Or hotlink direct to there from the QBS homepage at:

http://www.qbss.com

QBS SOFTWARE

* prices, subject to change and exclusive of shipping plus VAT

Low bandwidth

Back from holiday, Jules gets a new toy. It's not very big, it's perfectly safe, and it annoys nobody – why would anybody want such a useless item?

It's funny, isn't it, how nobody is neutral about mobile phones. They've been around for so long now, and they're sufficiently cheap that everyone who wants one has got one, that you'd think we'd have learned to live with them by now. But, it doesn't seem to be so; what's the polite thing to do if your phone rings during a meal, for example? The tones on mobile phones are hard to localise, such that whenever a phone rings in a crowd fifty people reach for their pockets, yet why have so few phones got vibrators inside them? And finally, should one take a phone to the pub at night – most publicans would say definitely not.

The real problem, though, is the telephone-like nature of a mobile phone. When the phone rings, you've got to stop what you're doing, even stop in the middle of a conversation, to answer it. Now, you have to be polite to the person who is interrupting your life without any consideration for what you happen to be doing – politeness meaning you have to spend five minutes or so greeting the caller and closing the conversation, before you can even begin receiving the message and negotiating its contents. That's expensive, and can ruin an already-running conversation. What's more, while using a phone in a bus is frowned upon, and using a phone in the underground or inside steel-framed buildings is chancy or downright impossible, using a mobile on a motorbike is a silly idea, and even using one in a car is increasingly being legislated against. Finally, the ability to call someone anytime, anywhere, and say 'Hello, hello ... can you hear me?' seems of dubious value at best.

I've noticed that I use my phone in two ways. One is simply to receive messages; 'I'm going to be late for our meeting', for example, or (more commonly) 'You're late!'. The other is to hold a conversation which involves some degree of discussion or negotiation. The messages are by far the lion's share of my use, and when I do need to negotiate, I much prefer to devote my full attention to it, organise my thoughts beforehand, and so on, so I use the phone only to make an appointment to have the discussion later.

Most large organisations have made the same distinction between these two kinds of communication for years, so they provide their employees with telephones, and with email facilities or (in the old days) with memo pads. The point is, of course, that a memo is not just a poorer, narrower-bandwidth form of communications; it's frequently better because it allows the recipient to prioritise his work, and to place the messages into filing cases or give them physically to someone else, and it forces the sender to be clear about what it is he's trying to say – memos rarely extend beyond fifty words.

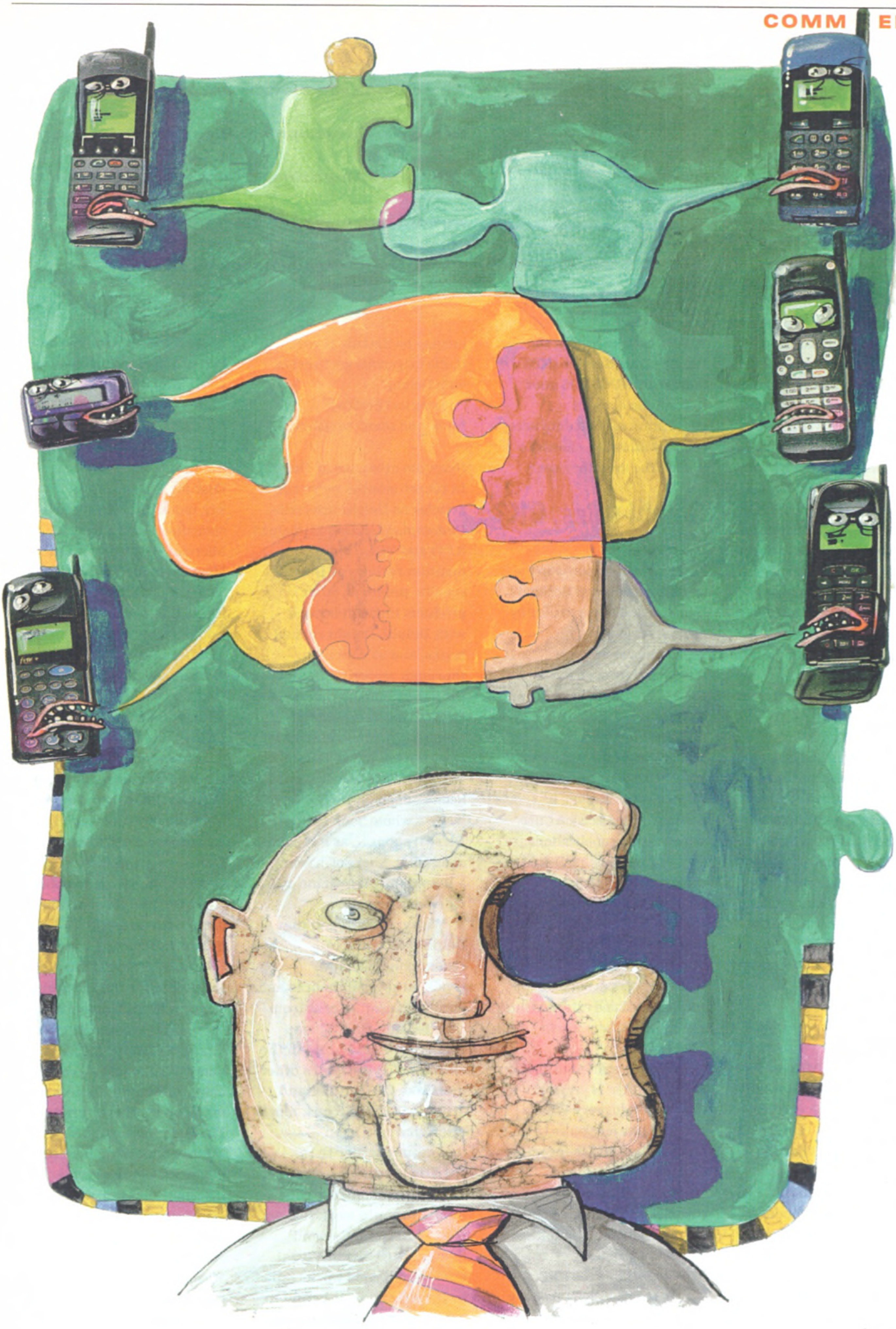
Though silly legislation around email is beginning to limit its use in certain litigious companies, memos and email are increasingly under-used because they are being provided to people who are uncomfortable with the written word. That's a shame, because some kind of mobile memo (other than Interflora, of course) is well worth having.

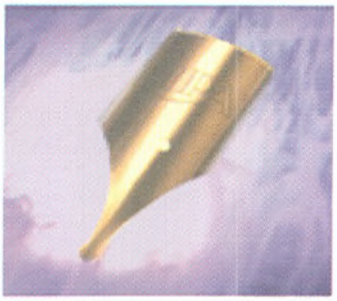
Well, I've discovered what I think is a remarkably well kept secret. I've just bought a pager. I diverted both my landline and my mobile to it for when I don't answer them.I'm finding more and more that I'm relying on it, to the extent that my mobile rarely gets switched on. I can clip the pager to my bike (or to the dashboard when I'm in a car), and when a message comes in I can read it right away – the screen is sufficiently big that most messages will fit in one page, yet there isn't so much information that I can't read it at a glance. (I know my phone has a message facility, but it's rendered useless because the display can only show a dozen characters). I can program my pager to vibrate silently (which was a somewhat unnerving experience until I got used to it), I don't need to keep worrying about charging it up, and I can selectively delete and shuffle messages around, entirely unlike an answering machine. It's so small that nobody can tell when I'm wearing it, and it has no value to a thief (an important consideration, considering my last mobile phone was cloned). I don't even have to worry about bad reception, because the system will keep sending the message until the pager gets it. This, quite simply, is what I'd needed all along.

I'm using a bureau service to collect the messages. At first I was disturbed by the typos on the messages which arrived; names would be spelled wrong, funny contractions would appear. After a while I realised that none of these things made the messages any less comprehensible, and most made them better able to fit into a page. The errors and inaccuracies are actually part of what makes this system work as well as it does. Where most digital systems are very intolerant of errors (dropout on a digital phone is a truly horrible noise, for example, and in July I wrote about dropout on a hard disk), here's a system which tolerates errors very well, because it's designed to be low-bandwidth, lo-fi, and interpreted by the immensely flexible mind of man, rather than the complex and rigid digital discriminators used by telephones and disk drives.

My phone usage can't be that much different to most other people's, and it seems to me that messaging systems are a far more efficient use not just of users' time, but also of the radio bandwidth. How come nobody makes mobile memo pads, like the mobile phones? You could build stuff into laptops and palmtops, you could have Microwriter-type keypads on the back of cigarette packet sized pagers, and you could build voice recognition or handwriting recognition or even gestural interpreters into the same cases we use now – 100% accuracy is not, after all important for what would otherwise take the form of a handwritten note. ■

Jules can now be paged on 01459 110967. You can still email him as jules@cix.co.uk, or call him on 01707 662698, but neither of those will be any good when he's out on his motorbike, or if he's not answering his phone.





We welcome short letters on any subject relevant to software development. Please write to: The Editor, *EXE Magazine*, St. Giles House, 50 Poland Street, London W1V 4AX, or email editorial@dotexe.demon.co.uk

MSFT's TLAsz

Dear Sir,
Has anyone else noticed that Bill's camp has gone decidedly schizoid recently? Do they not understand that VBX, OCX, OLE, and ActiveX are all the same thing? And so are ODBC, ADO, RDO, and ODBC-direct. And more recently DAO, OLE DB, ADC, JET and… RDS? Do they think that we can't figure it out?

Time was when a new design or a new release was just that. Mostly the same as the old, with a new twist. There was a little honesty and integrity involved: 'Sure we messed up, but we've thought of a better way. Try this.' Everyone understood the rules.

Now Microsoft heralds each new release of a few dull drivers as an exciting new technology. Either they know what they're doing and they think we're too dumb to figure it out, or marketing truly believes the tech boys are inventing gold every minute, and they're too stupid to figure it out themselves

I've always been a staunch supporter of Microsoft because I've always seen Bill as the underdog. Trying to fight the multiprocessor, multithreaded, distributed processing Unix crowd with DOS and Windows 3 took some balls. Now Bill has read the Unix manuals, are his conjugal responsibilities taxing his sanity?

Is this a case of 'The Emperor Has No Brain'?
Alan Ambrose
alan@alan-ambrose.com

Straight from the Microsoft Web site: 'New Name for ADC--Remote Data Service (RDS) – ADC has now been integrated with ActiveX Data Ojects (ADO) to provide date remoting [sic] within the same programming model as ADO. This makes it easier to design, code, and deploy both Web-based and LAN-based applications. To clarify the relationship of ADC to ADO, ADC is now known as the Remote Data Service (RDS), a feature of ADO. RDS goes beyond the current generation of Web data access tools by allowing clients to update the data they see.'

Delphi rules, OK?

Dear Sir,
Just a note in response to the criticism levelled at Borland's Delphi 3, and in particular its pricing structure.

The enormous price differential between the Professional and Client/Server editions is justified not just by the database connection capabilities of the latter but the extra VCLs that ship with the Client/Server version, in particular the Decision Cube. When creating applications derived from existing Delphi components it's well worth spending some time exploring the possibilities of subclassing the Decision Cube. This component is difficult to master in its native form but once encapsulated and extended into a user-defined component… well, what can I say!

I recently started to develop in Delphi after 14 years of C/C++ skulduggery. What a relief! If like me you're tired of bit-twiddling and re-inventing the wheel then switch to Delphi. Its object model is far superior to C++, and of course, it's not another Microsoft bloatware product.

When will we see EXE (which seems to deny the very existence of the said product, judging by the lack of coverage in the magazine) get to grips with Delphi?

Long live Borland!
Name and email supplied.

Deny the existence of Delphi? In recent months we've run articles on Delphi internals, Undocumented Delphi, Delphi RTTI and VCLs, not to mention reviews of Delphi-specific products, other products that can be used with Delphi, Delphi-related books, and of course a thorough review of Delphi 3 itself. Remember Anders Hejlsberg (the creator of Delphi)? We interviewed him after the launch of Delphi 2. We have Dave Jewell, who is keen enough on Delphi to write a whole book (*Instant Delphi*) about it, as a regular columnist. Short of becoming a Delphi-only magazine, what more could we do?

We agree with you that Delphi has much to recommend it. As we reported several months ago, even Microsoft uses it from time to time. Should we concentrate on Delphi to the exclusion of Borland's other tools like C++ Builder and JBuilder? EXE's focus, more often that not, is on techniques rather than specific products; unless otherwise stated, the code we publish should work just as well in Borland C++ and Symantec C++ as in Visual C++. Delphi code works only in Delphi and so is bound to have a more limited appeal.

As for Delphi having a better object model than C++… well, we'll leave that for others to comment on.

Tangy processor

Dear Sir,
Err, unless they produced two computers, the Tangerine Microtan 65 (mentioned in Mayhem in the last issue) was a 6502-based machine, not Z80-based. But to point that out would be picky, wouldn't it. 8-)
Matthew Jones
mattjones@cix.co.uk

Well, what can I say? You are picky… and mostly correct!

After much research – isn't the Web wonderful? – I found that the CPU of the Tangerine Microtan 65, released in 1979, was indeed a 6502 and not a Z80 as we printed. This 6502 was running at a whopping 0.75 MHz and the machine came with no less than 1 KB of RAM. However, Tangerine Computer Systems Ltd did all the design for the Tangerine Tiger, a desktop machine with three processors: a Z80 for CP/M, a 6809 for I/O, disk and printing, and a graphics chip. Unfortunately, this machine was never produced. The Oric was designed by the same company, then an R&D house for Oric Products International Ltd. – Ed

More information can be found at www.tardis.ed.ac.uk/~alexios/MACHINE-ROOM/Tangerine_Microtan-65.html and at www.ensica.fr/~frances/oric/BOOK/chapter1/html.

Controlling distributed development across the Web is a reality with StarTeam

The easy to use version control and defect tracking tool

StarTeam is the first Windows® 95 and Windows NT™ SCM Integrated Team Environment™ for collaborative development.

Seamless functionality provides departmental and geographical integration throughout the development process.

StarTeam Workstation Professional 2.1 & StarTeam Workstation 2.1 includes:

- Version control
- Defect tracking with process management
- Threaded Conversations
- Build management
- Audit logs, reports & charts
- Central project repository
- New lower price

StarTeam VirtualTeam Server 2.1 includes:

- Internet, dial-up, WAN or LAN access to projects.
- Web browser access to projects via dynamically generated HTML pages
- Client/Server architecture to reduce network traffic
- Enhanced security

CONTEMPORARY
software

Talking Your Language

ORDER TODAY

Call: 01344 873434 **Email: sales@contemporary.co.uk**

Fax: 01344 872228 **Website: www.contemporary.co.uk**

StarTeam®
The Integrated Team Environment™

ENQUIRY NO. 09

PCWEEK LABS ANALYST'S CHOICE OCTOBER 7, 1996

PCWEEK ITX CORPORATE IT EXCELLENCE APRIL 1996

StarBase®

Taking advantage of

Although generally thought of as bad practice, self-modifying code can be extremely useful. When used for dynamic optimisation, it allows for layers of logical abstraction, without the accompanying overhead. Robert Ennals looks at how to inline virtual function and dynamically unroll loops.

We generally associate self-modifying code with hackers writing programs which are hard to prove and hard to maintain. It is also condemned by Intel: the Architecture Optimisations Manual lists 'Avoid self-modifying code' as number 12 in a list of ways to produce faster executables. This is due to the overhead from pipeline and cache flushes when code is modified. However, there is a good side to self-modifying code. When applied by a carefully proven dynamic optimiser it can lead to incredible performance increases. If the creation and management of self-modifying code is left to a dynamic optimiser then one can have the speed benefits without the problems.

The never ending optimiser

A dynamic optimiser is different to a static optimiser in one important way. While a static optimiser creates optimised machine code and then leaves it alone, a dynamic optimiser never stops optimising the code and may change the resultant code as conditions change. Listings 1.1, 1.2, and 1.3 illustrate the basic operation of dynamic optimisers.

The most common type of dynamic optimisations, the *timid optimisations,* are those which produce optimised code that will function correctly, whatever state the machine is in. Although a timid optimisation will correctly function for any state, we can optimise it so that it is faster when certain assumptions about the state of the machine are true. It will however often be slower when the assumptions are false.

There is a more complex type of dynamic optimisation – *fierce optimisations* – where we produce code that will not always function correctly. This code will only function correctly when certain assumptions about the state of the machine are true. This code will often be significantly faster when these assumptions are true, at the cost of not working at all when they are false. In order for the code to always work and the optimisation to be invisible, the optimiser must catch any attempts to perform actions that would change the state of the machine such as to break the assumption that the generated code relies on. When it catches such an event, it must change the optimised code that relies on the assumption so that it behaves correctly with the new machine state. This can be done by attaching notification code to any code that could break the assumption. With some languages, such notifications are very hard to attach, but in other languages, such as Java, where there is no address space, attaching them is relatively easy.

Most fierce dynamic optimisations provide a speed benefit by moving an overhead from the code that depends on the assumption, to the code that causes the assumption to be broken. If the code that changes it is executed rarely and the code that depends on it is executed often, then a major performance benefit can result from moving the check.

Inlining the uninlinable

One of the simplest, but also most useful fierce dynamic optimisations is dynamic inlining of code. Static inlining is a very common optimisation on conventional static optimisers. It consists of moving code from a called function that is known never to change, into the function that called it. For small functions, the overhead of the call can be significant compared to the time spent in the function and so the resultant speed up can be significant. An optimiser supporting static inlining might

```
Code that depends on state of X:
{
     inspect X
     do action relevant to state of X
}
```

Listing 1.1 – Basic structure of code before fierce dynamic optimisation.

```
Code that depends on state of X:
{
     assume X has state Y
     do action we would perform if X always had value Y
}

On Changing X:
{
     alter code so that it does not depend on
     previous assumptions about the state of X.
}
```

Listing 1.2 – Basic structure of code that is being fiercely dynamically optimised.

```
Y is a very common state of X.
Code that depends on state of X:
{
     if(X == Y)
     {
          do action we would perform if X had value Y
     }else{
          work out what we should do
          do relevant action
     }
}
```

Listing 1.3 – Basic structure of code that is being timidly dynamically optimised.

self-modifying code

optimise Listing 2.1 into Listing 2.2. This static inlining optimisation relies on the fact that the called code is always the same.

Dynamic inlining lets one inline called code, even if the called code is not always the same. We can for example inline virtual functions and dynamically linked functions. A dynamic optimiser might optimise Listing 3.1 into Listing 3.2. When the code being called is not inlined, an overhead is incurred every time the code is called, so that the code can run when the code being called changes. Dynamic inlining allows us to remove this overhead in exchange for an overhead when there is a change to which function will be called.

Let's look at the code in Listing 3.1 in more details. If I regularly invoke a virtual method of object pointer m_obj, every time I am suffering an overhead in order for my code to be able to call the correct method code whatever object pointer m_obj points to. If the object m_obj changes rarely and the method is called frequently, then one may gain a speed improvement by dynamically inlining the code in the called method. The speed improvement is gained by removing the requirement for the code to work when m_obj references a different object. However our program needs to continue to work when m_obj is set to reference a different object, so the code for Bar() must be regenerated if m_obj is changed such as to invalidate the existing code.

```
class A
{
     int foo(int a) {return a + 1);
}

class c
{
     // method called many times
     int Bar( int num )
     {
          A obj = new A;
          return obj.foo(num);
     }
}
```

Note that the optimiser can easily prove that obj will always be an instance of A and so can rely on the fact that obj.foo always calls the same code.

Listing 2.1 – Code suitable for static inlining.

```
class c
{
     // method called many times
     int Bar( int num )
     {
          return num + 1;
     }
}
```

The implementation of obj.foo has been inlined into Bar and obj has been optimised away as it is an unused object with no construction side effects.

Listing 2.2 – Code after static inlining.

The optimiser does this by inserting extra code into the method that changes m_obj so that it tells the optimiser that the code for Bar() needs to be regenerated. The optimiser will respond to this notification by either generating code that depends on the new contents of m_obj, or producing code that can work for any value of m_obj and removing the notification code from SetObj(). By inserting extra code into the SetObj() method we are adding an extra overhead to changing the reference. This overhead is likely to be significant, especially as many processors require a complete flush of the instruction cache if code is modified. In order for this optimisation to be beneficial, the gain in speed from removing the virtual method call overhead in Bar() must be greater than the loss in speed from adding a significant overhead to SetObj().

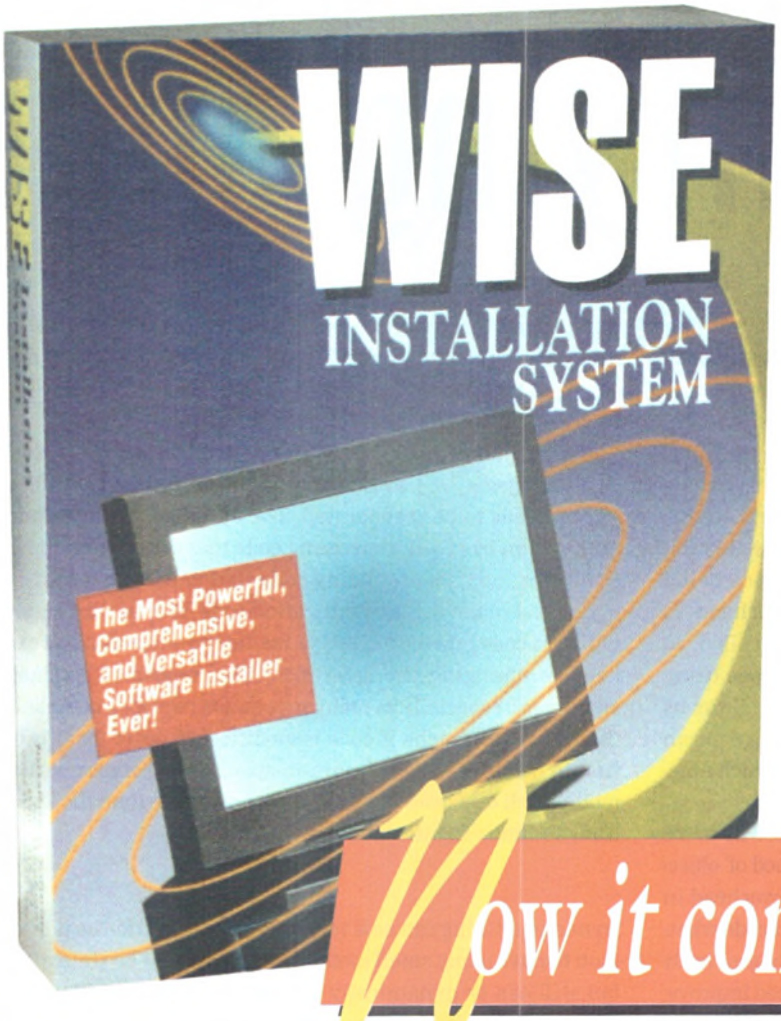
Get Set go

Dynamic inlining can be a very useful optimisation as it is very common to have many small methods which cannot be statically inlined, but still can be dynamically inlined. For example often well structured code has lots of very small Get and Set methods that do very little (Listing 4 gives an example). When they are implemented as statically linked non-virtual methods they can be statically inlined and there will be no overhead. However if one wants to dynamically link to an object, or call such methods on an object via a pointer which may point to an instance of any of several derived classes, static inlining will not be possible.

These Set and Get methods are a very useful alternative to public member objects as they allow the object to react to changes in its members and so keep itself consistent. These methods tend to be very small, often simply setting and returning the value of an object held internally. As these methods are so small, the overhead of the call will often be very significant compared to the overhead of the actual action. When Get and Set methods are used in inner loops, the time spent in them can end up being a significant proportion of the time a program takes to do something. I have come across people who insist on using public data members instead of public accessor methods just because of this overhead.

Fortunately, this overhead can be avoided with dynamic inlining. Often the pointer will point to the same object most of the time and the methods will be invoked extremely frequently. Even when the object being pointed to changes, it will often still refer to an object that does exactly the same thing in its Get and Set methods, thus allowing the inlined code to be updated relatively rarely.

Get and Set methods are not the only example of small dynamically linked methods that benefit from dynamic inlining. Graphics routines are another good example. In order to achieve device independence, routines for doing things like writing pixels to the screen tend to be dynamically linked. They also tend to be executed very often and almost always call the same code. On top of this, they also


WISE
INSTALLATION SYSTEM
The Most Powerful, Comprehensive, and Versatile Software Installer Ever!

They say wisdom comes with age.

Now it comes in a box!

The Wise Installation System 5.0

is wisdom in a box. It is powerful, intuitive to use and completely customizable. The Wise Installation System, a Windows based installation editor, creates professional setup programs in hours, not days. It creates a single installation that runs in Windows 3.1x, Windows NT™ or Windows 95®. It comes complete with both Win16 and Win32 versions, creates shortcuts/shell links for use with Windows 95, handles nested components and has a full uninstaller with rollback.

The Wise Installation System Enterprise Edition combines the award winning Wise Installation System with multi-generational patching, web based installation support, and a repackager for converting existing installations.

Now shipping...

The Enterprise Edition

Your complete answer to all your software distribution needs!

Great Lakes Business Solutions

http://www.glbs.com to download a fully functional demo.

QBS SOFTWARE

QBS Software Limited
11 Barley Mow Passage
London W4 4PH
Phone: 0181 956 8000
Web: www.qbss.com

Component Source™

ComponentSource
27-37 Vachel Road
Reading, RG1 1NY
Phone: 0800 58 1111
Web: www.componentsource.com

Microsoft, Windows, and the Windows logo are registered trademarks of Microsoft Corporation.

Designed for Microsoft® Windows®95

ENQUIRY NO. O10

```
public interface A
{
     public int foo(int a);
}

class B implements A
{
     int foo(int a) {return a + 1;}
}

class c
{
     // member that is currently referencing
     // an instance of B
     private A m_obj;

     void SetObj(A obj) {m_obj = obj};

     // method called many times
     int Bar( int num )
     {
          return m_obj.foo(num);
     }
}
```

Note that a static optimiser cannot prove what code will be called by `m_obj.foo(num)` and so would not be able to inline `foo`.

Listing 3.1 – Code suitable for dynamic inlining.

```
class C
{
     // member that is currently referencing
     // an instance of B
     private A m_obj;

     void SetObj(A obj) { alert_optimiser; }
          // optimiser needs to change inlined code
          // if m_obj is set to point to something
          // which isn't an instance of B

     int Bar(int num)
     {
          return num + 1;
     }
}
```

The dynamic optimiser has produced code that relies on `A` always being an instance of `B`, even though it cannot prove that this is always the case. If this assumption is broken, the code will change. We might also inline the calls to `Bar`, removing yet another layer of overhead.

Listing 3.2 – Code after fierce dynamic inlining.

```
class C
{
     // member that is currently referencing
     // an instance of B
     private A m_obj;

     void SetObj(A obj) { m_obj = obj };

     int Bar(int num)
     {
          if(m_obj = magic_common_obj)
          {
               return num + 1;
          }else{
               return m_obj.foo(num);
          }
     }
}
```

Note that `magic_common_obj` is set by the dynamic optimiser to an object that `m_obj` often points to. This code always checks before calling the inlined code and so will always behave correctly. Thus the optimiser does not have to re-optimise `Bar()` every time `m_obj` is changed.

Listing 3.3 – Code after timid dynamic inlining.

tend to be very small. All this makes them a good place to apply dynamic inlining.

Speeding up patterns

Another very common and very useful programming pattern that can be sped up a lot through dynamic optimisation is multicasted notification. This pattern works as follows:

Objects which other objects can depend on hold a list of objects that depend on them. When another object needs to depend on the first object, it calls a method of the first object, adding itself to this list. When the first object is changed, it iterates through this list notifying the objects that depend on it that it has changed.

Listing 5 gives an example of possible Java interfaces for change notification multitasking. They are based on the interfaces used by Microsoft in OLE. Listing 6 gives an example of a very common way of implementing the function that multicasts the change notifications to dependent objects.

When objects are notified by a notification multicast, they usually do one of two things, depending on whether they are lazy or eager. Lazy objects will invalidate their current contents and leave regeneration until the object is needed. Eager objects will regenerate their contents to be consistent with the change in the object they depend on. Usually large, rarely used objects will be lazy, and small, frequently used objects will be eager. Leaving generation for later saves generating data that will not be used, and regenerating the object at notification-time saves the overhead of checking if the object exists when it is requested.

Let us take as an example a file browser similar to the windows explorer. One might have an object representing the selected item in the tree on the left. Several other objects, such as the tree display, the contents display, the status display, and any other related items, would depend on which item was selected. The object that represented the selected item would not know which other objects depended on it at compile-time. The dependent objects would register themselves with the object representing the selected item at runtime and would update their contents when this object notified them that it had changed. This model is very useful as it does not require objects to know what other objects depend on them at compile-time. This allows other objects to bind to the object they depend on without requiring it to be recompiled.

Another useful example is graphics output. A program might have a bitmap that it regularly writes pixels to. It could provide a notification system whereby bitmaps that depended on this bitmap were told when a new pixel was added to the main bitmap so that they could add relevant pixels to themselves. In this case, the change notification method on the dependent objects would be given a 'hint' saying which pixel had changed.

This notification model is very widely used. It is the standard way for objects to depend on each other in OLE and in many other systems. It is becoming increasingly common to make class hierarchies which contain only `const` methods and notifications. Thus preventing any object from ever being allowed to write to any other object directly or to know when other objects would need to be written to. Instead of writing to other objects, one changes the value of one of one's sub-objects. If other objects need to be changed to reflect this, they will have asked for notifications from this sub-object.

The only problem with change notification multicasting is that when one uses static compilation it can carry quite a large overhead

```
class A
{
// --- public accessor methods ---
     public thing getThing() {return m_thing;}
     public void setThing(thing in) {m_thing = in;}

// --- private member data ---
     private thing m_thing;
}
```

Note that very little is actually done in the `Get` and `Set` methods

Listing 4 – Common use of Get and `Set` methods.

```
public interface IchangeNotify
{
     key Advise(INotifySink adviseme);
          // we wish to be told when you change

     UnAdvise(key who);
          // we no longer wish to know when you change
}

public interface INotifySink
{
     void NotifyChanged(IChangeNotify who, CHint hint);
          // The object referenced by 'who' has changed
          // Hint indicates what kind of change has
          // taken place

}
```

Listing 5 – A possible interface for objects supporting change notification. Based on the interfaces used by OLE.

```
void Changed()
{
     int i;

     for(i = 0, (i < m_dependents.length), i++)
     {
          m_dependents.member(i).NotifyChanged(this);
     }
}
```

Note how all dependent objects are notified from the same line.

Listing 6 – A typical change notification routine.

Instances of `dependentA` and `dependentB` have registered with the `advise` method (not shown) of `changable`.

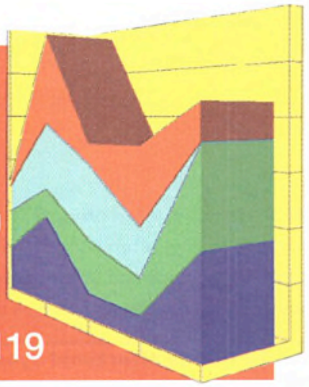
```
class dependentA implements INotifySink
{
     public void NotifyChanged(IChangeNotify who)
     {
          changable obj = who;
               // dynamic cast with type check overhead
          m_myInt = obj.GetInt() + 1;
     }

     private int m_myInt;

     // rest of class omitted
}

class dependentB implements INotifySink
{
     public void NotifyChanged(IChangeNotify who)
     {
          m_valid = FALSE;
     }

     private bool m_valid;

     // rest of class omitted
}

class changable implements IChangeNotify
{
     // method called when object has changed
     private void Changed()
     {
          int i;

          for(i = 0, (i < m_dependents.length), i++)
          {
(*)            m_dependents.member(i).NotifyChanged(this);
          }
     }

     private dependentSet m_dependents;

     // rest of class omitted
}
```

If instances of both `dependentA` and `dependentB` have registered for notifications from an instance of `changable`, then every time the asterisked line is executed, the virtual function called will be different, causing a mispredicted jump on many processors.

Listing 7.1 – A possible dependency arrangement.

with it. As objects do not know what other objects need to be notified at static compile-time, a static compiler can apply very little optimisation to change notification routines. In order to be ready for the needs of future objects it is considered good style to support notification of changes on as many objects as possible, including those that no objects currently depend on. Consequently, whenever one changes anything there is likely to be an overhead for notification. Even if no objects depend on the change one is making there is the overhead of checking to confirm that no objects depend on the change, the memory overhead of having to have code in memory to do this checking, and the memory overhead of storing the list.

When objects do depend on others, the overhead of changing the object that is depended on is much larger. Multicasting a change notification involves iterating through a list



www.qbss.com

LOW PRICES VAST RANGE EXPERT SERVICE

Visual Basic

Addsoft *(Upgrades available)*	
Gantt VBX/OCX	£189
Resource Manager VBX	£189
Schedule VBX/OCX	£189
Apex *(Upgrades available)*	
TrueGrid Pro VBX	£99
True dBgrid OCX v 5	£225
MyData Control	£115
Bennet-Tec	
Alltext Std/Pro	£115/265
TList OCX or VBX Std/Pro	£115/165
Crescent *(Upgrades available)*	
ClassAction	£95
DBpak	£120
FTP Internet Data Control	£70
Internet ToolPak	£125
PDQComm VBX/OCX	£85/130
PowerPak Pro OCX	£445
PowerPak Enterprise	£445
QuickPak Pro VBX/OCX	£125/149
QuickPak VB/J	£149
Scope/VB	£75
VB Appframework	£120
VBNet	£130
Desaware *(Upgrades available)*	
SpyWorks 5 Pro (incl upgrade subs)	£185
StorageTools OCX	£90
Version Stamper VBX/OCX	£99
Farpoint	
ButtonMaker OCX	£70
Input Pro	£99
List Pro	£129
Spread VBX/OCX	£210
Tab Pro VBX/OCX	£90
GreenTree	
ActiveX Suite	£375
DataList	£95
DataMask	£95
DataTree	£95
DataView	£95
XGrid	£95
ImageFX	
Fractal FX VBX+OCX	£150
FXPic VBX + OCX	£205
FXTools/VB Pro VBX + OCX	£280
Planet FX ActiveX control	£200
Vector FX OCX	£245
Luxent (formerly DFL)	
Business Graphicserver Std/Pro	£165/299
Docs&Images Std/Pro	£165/299
MagicMenus Std/Pro	£65/89
Audio&Video Std/Pro	£165/299
Vectors&Layers Std/Pro	£165/299
Power Pack Std/Pro	£330/530
ProtoView	
Data Explorer	£135
DataTable OCX or DLL	£135
DataTable OCX & DLL & VBX	£265
InterAct Diagraming OCX or DLL	£275
InterAct Diagraming OCX & DLL	£345
WinX Component Library	£135
Sheridan	
3D Widgets	£75
Calendar Widgets	£90
ClassAssist	£169
Component Suite (Cal+Data+Des)	£235
Data Widgets	£90
Designer Widgets	£90
Sp_Assist	£415
VBAssist	£125
VideoSoft *(Upgrades available)*	
VS Data ActiveX	£115
VS Direct	£145
VSFlex VBX/OCX	£89/115
VSFlex upgrade from VB5	£45
VS Reports ActiveX	£115
VSVBX/VSOCX	£40/75
VSView OCX	£115
Visual Components *(Upgrades available)*	
Codebank for VB4	£99
dbComplete NEW	£375
First Impression ActiveX	£195
Formula One ActiveX	£199
Geopoint 32 bit ActiveX GIS	£199
SystemTools	£55
Visual Speller ActiveX	£99
VisualWriter ActiveX	£199
WebViewer	£199
Miscellaneous	
3D Graphics Tools 16/32	£119/139
Active Bar	£110
CodeReview Standard/Pro	£149/299
ErgoPack for VB, VC++ and Delphi	£225
Erwin Desktop for VB	£395
FailSafe Standard/Pro	£149/255
ForeVB	£69
Helping Hand (Help for VB)	£99
HighEdit Proactive	£275
Intelligent Paper	£125
LeadTools 7 OCX 16 bit or 32 bit	£275
List & Label forVB	£275
PowerBasic Developer Kit (PBDK)	£115
Rockwell RSAnimator	£159
Rockwell RSToolpak I/II	£149/185
TX Text VBX/OCX/combo	£199/199/299
VBCompress 4.0	£85
VB Language Manager 3.0	£199
VBtrv Btrieve from Classic Software	£175
Vision StoryBoard Pro	£145
Visual Expert	£250

OCTOBER NEWS

dbComplete from Visual Components. We've been eagerly waiting for this excellent package for months - and now it's available. In the opinion of our techies, dbComplete is one of the most useful database component libraries - especially for Visual Basic folk - to come out in years. Nine ActiveX controls are included; the three primary ones being: Outline Data Grid Control presents data from multiple queries in an Explorer-like, outline format; hosts third party data controls. Data Source Control - and ODBC-based control that manipulates master-detail data. Data Bound Control Library - six data-aware ActiveX components (Combo; Calendar combo; Picture combo; Masked edit; Option group; Check box) that can be used with either or both the above-mentioned Outline Data Grid Control and Data Source Control. £375.

SoftDev/WebDev 1997. 12-13 November, Olympia 2, London. We'll be there - stand 250 - with our partners for the show WallData (demonstrating Arpeggio *live!*) and Visual Components (demonstrating dbComplete (see above)). Tickets? Please call 0181 240 5042.

Crystal Reports v 6. Latest version of the industry-leading report writer. New web components provide advanced Web reporting capabilities. Smart Preview #1: Smart Navigation™ allows drill down and search in runtime reports. Smart Preview #2: Event & Callback support allows interactive user response. Automation Server has over 500 properties, methods, events. Active data driver: integration within VB, reporting off runtime data. Many more superb new and existing features.

Delphi

Abbrevia 1.0	£129
ABC for Delphi Pro VCL	£135
ACE Reporter Pro	£165
Apiary Dev's Suite for Netware	£365
Apiary NetBIOS Custom Control	£79
Apiary OCX Expert	£189
Apollo 4 Standard/Prof.	£129/199
Apollo Source Code Kit	£65
Async Professional 2.0	£129
BoundsChecker 5 for Delphi	£325
Component Create	£135
DBPower Standard	£99
Delphi 3.0 Standard/Pro	£79/369
Delphi 3.0 Client/Server.	£1329
Delphi 3.0 Professional Upgrade.	£239
Delphi 3.0 Client/Server Upgrade.	£1089
Direct Access	£189
Dr Bob's Wizards v2	£49
Dynazip 16 bit/32 bit/combo	£175/210/289
Eagle CDK 16/32	£189/209
Essentials Vol 1 VCL library	£50
ImageLib 16 & 32 bit	£145
ImageLib@theEdge	£99
Infinity Report VCL	£89
Infinity Security Component	£89
Infinity MAPI Component	£89
InfoPower 3.0 VCL/+Source	£150/250
Innoview Multilanguage Std/Pro	£125/750
King Calendar	£75
List & Labels for Delphi	£275
Memory Sleuth	£40
MK Query Builder (incl. source)	£149(299)
Piparti Reporting Std/Pro	£175/350
Pumpkin Project Manager	£69
Orpheus 2.0	£129
Raize Components NEW v 1.6	£135
Quick Reports 2 with source	£79
Shoreline DialogPROS	£185
Shoreline VisualPROS	£115
Titan for Btrieve /+source	£225/379
TOLEAutomation Client	£39
TRPKCrypto Single User	£95

Inter Networking Tools

Arpeggio *live! Starter Kit*	£1175
Borland IntraBuilder Std/Pro	£69/349
Borland IntraBuilder Client/Server	£1279
Catalyst SocketTools	£190
Dart PowerTCP	£450
Distinct TCP/IP SDK 16 or 32 bit	£375
Distinct Internet Toolkit Visual Edition	£199
Distinct Runtime Ext./Ext.+ Winsock	£49/99
Mabry Internet Pack/(inc. source)	£99/(275)
Net-Install from 20/20 Software	£225
Sax Webster Control16/32	£105/105
SoftSENTRY	£499
Wise WebDeploy	£129

C/C++

Borland C++ v 5.0/Dev Suite	£225/315
Borland C++ Builder Professional	£389
Borland C++ Builder Client/Server	£1279
Borland C++ Builder Pro/CS Upg	£239/1089
BoundsChecker Pro C++	£449
C-Vision from Gimpel	£179
CodeBase 6.0	£239
CodeSQL 6.0 5 user/Unlimited	£350/2420
Greenleaf Comm++	£205
Greenleaf Datawindows	£225
High Edit SDK/PRO	£275/375
HiProf from TracePoint	£475
InfoPower++/ with source NEW	£150/250
Leadtools7 WinPro 16/32 bit DLL	£575/725
MKS Toolkit	£329
Object Master	£195
Power++ Dev/Pro/Ent	£339/679/1369
Power++ Dev upgr. From Optima	£90
PCLint from Gimpel	£179
Stingray Objective Diagram 32	£330
Stingray Objective Grid (16/32 bit)	£299
Stingray Objective Toolkit (16/32 bit)	£375
VBtrv for C++	£245
Visual Coverage	£475
Watcom C/C++ CD only/docs	£240/90

Microsoft

Office 97 Pro (Incl Access)	£449
Office 97 Dev. Upgrade from 97	£230
Visual Basic 5 Pro/Enterprise	£369/939
Visual Basic 5 Upgrade Pro/Ents	£185/529
Visual Basic 5 Comp. Upgrade Pro	£219
Visual C++ 5 Pro/Ents	£365/925
Visual C++ 5 Upgrade Pro/Ents	£235/535
Visual FoxPro 5 Pro	£395
Visual FoxPro 5 Upgrade	£225
Visual J++	£75
Visual Studio Pro / Enterprise	£765/1135
Visual Studio Upg Professional	£395
Visual Studio Upg Enterprise	£775

Client/Server Databases

Borland *(Upgrades available)*	
InterBase 4.0 Workgroup Server	£625
InterBase 4.0 Additional single licence	£120
InterBase 4.0 Additional ten licences	£990
InterBase 4.0 Additional twenty licences	£169
Paradox 32 bit	£99
Paradox 32 bit client/server	£449
Visual dBASE Client/Server	£369
Visual dBASE Compiler	£229
Extended Systems *(Upgrades available)*	
Advantage Database Server	
CA VO/CA Clipper Interface Client	£75
ODBC Interface Client	£195
SDE 2.0 Interface Client	£215
Sixty day Server Evaluation Pack	£70
2 User Server Development Kit	£215
5 User Server Deployment Kit	£695
10 User Server Deployment Kit	£1145
25 User Server Development Kit	£1895
100 User Server Deployment Kit	£3445
Microsoft *(Upgrades available)*	
Access 97 (or v 2)	£265
SQL Server NT 6.5 incl. 5 user lic.	£1035
SQL Server NT 6.5 incl. 10 user lic.	£1420
SQL Server NT 6.5 incl. 25 user lic.	£2845
SQL Server NT 6.5 incl. 50 user lic.	£5685
SQL Server NT 6.x Twenty Client Lic.	£1895
SQL Server NT 6.x Workstation	£385
Powersoft/Sybase/Watcom	
PowerDesigner AppModeler	£695
PowerDesigner AppModeler Desktop	£195
Sybase SQL Anywhere Single User Svr	£190
Sybase SQL Anywhere 4 User	£395
Sybase SQL Anywhere 8 User	£650
Sybase SQL Anywhere 16 User	£1115
Sybase SQL Anywhere 32 User	£1975
Sybase SQL Anywhere Unlim.	£3300
Oracle	
Personal Oracle 7	£305
WorkGroup Server (5 users)	£1150
Developer or Designor 2000	£3350
Programmer 2000	£835
Power Objects Standard	£335
Power Objects Client Server	£1675

Notes

Domino Starter Pack Vs 4.5	£895
Domino Server Vs 4.5	£689
Desktop Client on CD/licence only	£85/65
DocWatch	£599
Full Client on CD/licence only	£305/265
SequelNum	£119

Power Builder

Advanced Developers Toolkit	£135
PowerDesigner Appmodeler/D'top	£645/195
Component Pack	£call
PowerBuilder 5 Desktop/Pro	£195/1190
Enterprise	£3195
FUNCky for PowerBuilder	£159
PowerFrame App. Fr. Library	£305
PowerFrame Navigator Object	£99
PowerFrame Object Analyser	£99
PowerFrame TabFolder	£99

Clipper

Argos - Clipper to Delphi migration	£149
Blinker Linker	£199
Clip-4-Win	£195
dGE Graphics	£189
FUNCky 2.5	£199
NetLib Network Library	£299
NovLib Network Library	£199
Scripton PostScript Library	£129
SilverClip SPCS Comms	£249
Summer '93 Code Optimiser	£159
Telepathy Clipper Comms	£169

Java

Codebase 6 with Java Docs	£239
InstallShield 5 Java Edition NEW	£325
Jamba Professional	£99
JBuilder Professional NEW	£389
Java Workshop	£79
JFactory for Win95/NT	£165
JMoney for Win95/NT	£90
JTools for Win95/NT	£80
JWidgets for Win95/NT	£90
Stingray Objective Blend	£195
Stingray Objective Grid for Java	£265
Symantec Cafe	£119
Symantec Visual Café / Pro	£159/349

FoxPro

Cryptor	£299
Foxfire Developer's Edition	£225
FoxFix for DOS and WIN	£149
ReFox Decompiler	£295
xCase for FoxPro & V.FoxPro	£295

Tools

Aegis v 3 (1 user)	£199
Arpeggio DTop/Developer	£395/515
Arpeggio *live!* Starter Kit	£1175
Barcode Lib. DOS/WIN/95	£389/249/519
CBI Quick 16 bit/32 bit	£1535/2250
Crystal Reports v 6 Std/Pro NEW	£95/249
Crystal Reports v 6 Pro upgrade NEW	£149
Crystal Reports 6 Pro 5 user/upgNEW	£799/559
Crystal Info	£1195
DBest Barcodes for DOS/Windows	£199/345
DataDirect Dev Toolkit 2	£425
DataDirect ODBC Pack	£310
DemoQuick Sim. Plus 16/32 bit	£375/595
DemoSHIELD 32 bit	£315
Doc-To-Help 16 or 32 bit	£295
Documentation Studio	£495
ED for Windows v 3.7	£145
ForeHelp v 2	£275
Graphics Server Developers Kit	£195
HelpSite Help to HTML	£225
Installshield 5 Full/Upgrade	£625/375
Installshield Express Pro	£175
Interactive Brochure	£225
Interactive Catalog	£375
MKS Source Integrity	£499
MS SourceSafe 95 (platform)	£379
Novlib Network DLL	£199
MultiEdit Pro/Evolve	£135/109
PC-Install Win Pro 16/32 bits	£295
PC-Install DOS&WIN super combo	£375
PVCS Version Manager	£420
R&R Report Writer v8 xBASE NEW	£229
Wall Data Rumba	£133
SoftICE	from £329
SoftSENTRY	£499
SOS Help! for Win Info Author	£185
Telephony Toolbox TAPI/TSAPI	£475/975
Track Record/ + per seat	£279/179
xBase++ for OS/2	£350
Wise Installation System ver 5.0	£139
SmartPATCH for Wise	£129
WebDeploy for Wise	£129
Wise Enterprise	£429

Please check our ftp site
http://ftp.qbss.com
or BBS 0181 956 8011
for info, demos and evals

Call us today

QBS SOFTWARE

QBS Software Limited
11 Barley Mow Passage, London W4 4PH
Ph: +44 181 956 8000, Fax: +44 181 956 8010
BBS: +44 181 956 8011, Email: orders@qbss.co.uk
Web: http://www.qbss.com, L/server : info@qbss.co.uk

ENQUIRY NO. O12

*Prices, correct at time of going to press, are subject to change and do not include shipping and VAT.
All trademarks recognised.

```
private void Changed()
{
      magic_dependentA.m_int = myint + 1;
      magic_dependentB.m_valid = FALSE;
}
```

The dynamically created code can access private members of the client classes as although the code is physically in the `Changed()` method of the object depended on, it is logically in the `NotifyChanged()` method in the dependent objects and access restrictions only apply to the logical code position.

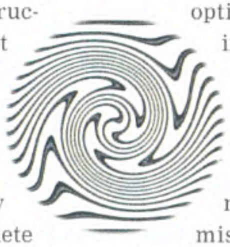
The references `magic_dependentA` and `magic_dependentB` have been dynamically created by the optimiser and reference items from the `m_dependents` object. The original loop has been unrolled.

Listing 7.2 – Dynamically optimised version of changable::NotifyChanged().

of pointers to objects and invoking a virtual method on each of them. We have the overhead of a virtual method call, and one that will often involve a mispredicted jump.

Unrolling the ununrollable

If more than one object is being notified (as is frequently the case) then every time we invoke a virtual method, we will be jumping to a different place to where we did last time from the same instruction. Many processors predict that any jump will always go to the same place it went to last time and so will always mispredict the jump. The overhead of a mispredicted jump can be very large as it often requires a complete pipeline flush. As the method being called usually does something very simple, such as setting a bit to say an object is now invalid or generating a small object, the overhead of the mispredicted jump will often be larger that that of the actual work being done.



Static loop unrolling can relieve this a little by providing more than one place where virtual functions are called from. Unless there are less dependent objects than there are unrolled stages the benefit will be minimal, and static unrolling increases code bloat as the static optimiser is unlikely to know where unrolling is sensible. It will either unroll too little everywhere and get no speed benefit at all when there are too many dependants, or unroll too much everywhere and bloat even those objects which do not have any dependent objects.

As a result of the aforementioned overhead, change notification multicasting is not used as much as it should be and systems that make intensive use of it are often criticised as being slow. Fortunately, dynamic optimisation can remove the overhead.

A dynamic optimiser can unroll the notification multicast loop and inline the code from the methods that would have been called. It can also optimise away notification functions for objects with no dependants. In order to maintain correctness in all conditions, it will insert code in the `Advise` and `UnAdvise` methods to cause the code to be regenerated when the set of dependants changes. A dynamic optimiser could optimise Listing 7.1 into Listing 7.2.

Generally the list of objects needing to be notified changes rarely and the notification code is called frequently, thus allowing major benefits. With dynamic optimisation, programs using notification for changing programs can be made to run as fast as objects which allow objects to call `mutor` functions on each other.

Dynamic unrolling of notification loops can make code feasible that would otherwise be unreasonably slow. Without dynamic optimisation, the graphics notification example given earlier would be far too slow to be of any practical use, despite its being a logically good model. Dynamic optimisation could make this graphics example as fast as writing the pixels to the bitmaps directly and allows many other good programming patterns to be used.

Next month: optimisation of dynamic binding. This series will be concluded by the dynamic optimisation of an interpreter into a compiler. ■

Robert Ennals is researching dynamic optimisation. He can be reached by email via the forwarding address ennals@iname.com which should follow him around when his real email address changes. He can also be reached by post at 19 Belgrade Road, Hampton, Middx, TW12 2AZ.

Help Authoring Tools From The Experts

OCC Have All The Answers

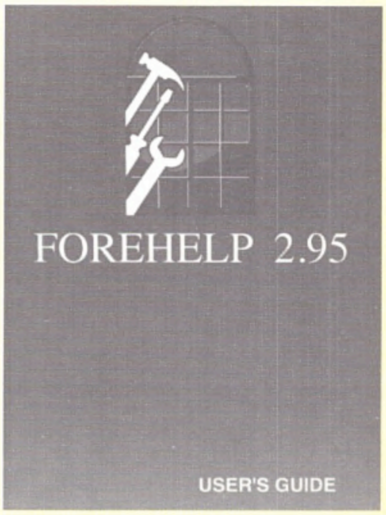
Help Magician Pro™
Professional Edition
Help Magician Pro
Hypertext Authoring System For Microsoft Windows
10
SII Software Interphase Incorporated

Help Magician Pro

Outstanding value, stand alone WYSIWYG tool. Test your Help file as you're developing. Develop Help and HTML files with one source.

ForeHelp 2.1/2.96

Award Winning WYSIWYG Help Authoring Tool. Grapher, Reporter and Navigator tools help manage and organize projects.



doctohelp
Single-Source Documentation and Help Authoring System for Windows
International Version 2.0
single source
insert

Doc-to-Help 1.7 & 2.5

Create commercial quality printed documentation and online Help from a single, easily maintained source.

RoboHelp 3.0/4.0

Only RoboHelp features a unique visual design and layout of your Help system for you. Just click your mouse, follow the on screen prompts and fill in the blanks with text and graphics.

HDK

Create simple Help and very large document systems with one tool. One product/ one project for ALL Windows platforms.

FREE 90 DAYS SUPPORT.
WE WILL BEAT ANY ADVERTISED PRICE.
SALES :01865 793 077

Oxford Computer Consultants
Littlegate House
St Ebbes
Oxford, OX1 1PS
Tel: 01865 793 077
Fax: 01865 793 124
www.oxfordcc.co.uk

OXFORD COMPUTER CONSULTANTS
sales: 01865 793077

Oxford Computer Consultants
Barclays Venture Centre
Sir William Lyons Road
Coventry, CV4 7EZ
Tel: 01203 690 931
Fax: 01203 411 727
E-mail: sales@oxfordcc.co.uk

ENQUIRY NO.045

How close would you like to get to Microsoft technologies?

With QA Training, you can get even closer.

QA Training remains the leading edge Microsoft technical training provider. Our close relationship with Microsoft means that we really do understand their technology best and are in a prime position to deliver the skills you need.

We've been training companies of all sizes on Microsoft products since Windows was first released and the superior knowledge and experience of our instructors is well recognised throughout the industry. In fact, most impressive of all, our instructors work alongside Microsoft's to help deliver training courses to Microsoft themselves.

QA offers an amazing range of over 65 Microsoft courses, covering products from Windows NT and the rest of the Back-Office Suite to Visual Basic, plus all the very latest releases. So if you want to stay ahead of your competitors make sure you use QA for your Microsoft training.

You don't even have to take our word that we are the premier training supplier. Computing has voted QA 'Best Training Company' in their 'Awards for Excellence' every year since its inception.

If you are searching for the best training in Microsoft technologies, get in touch with QA.

CALL US NOW ON 01285 655888 OR RETURN THE COUPON

Please send me QA's complete course catalogue for July – December 1997.

Name: Mr/Mrs/Ms/Miss ____________________ Company Name ____________________

Company Address ____

____________________ Postcode ____________________

Tel ____________________ Fax ____________________

QA

QA Training Limited, Cecily Hill Castle, Cirencester, Gloucestershire GL7 2EF Facsimile: 01285 640181 *http://www.qatraining.com*

ENQUIRY NO. 014

Taking on Goliath

NT device drivers are small. NT itself is huge. Yet, a badly-written device driver can bring the OS crashing to its knees. With an eye to avoiding the Blue Screen of Death, Chris Cant shows us how to do it right first time.

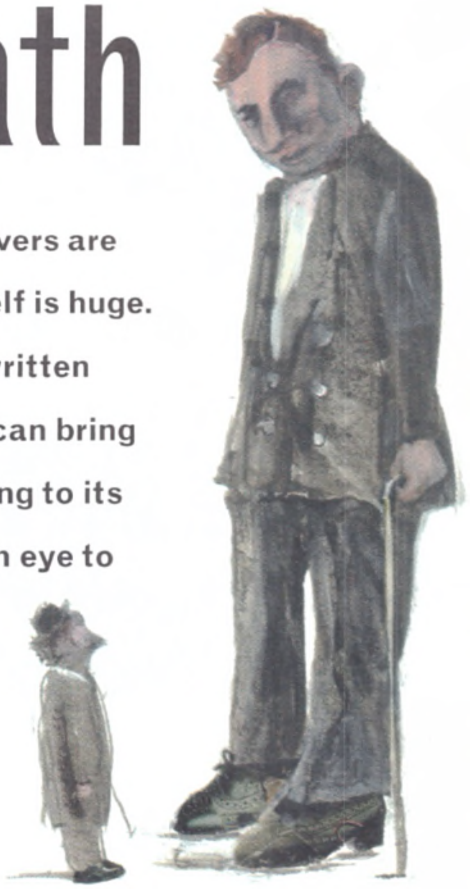


Illustration: David Humphries @ Monster

Of all the kinds of software, a developer can be called upon to write, device drivers are generally regarded as the worst. Since drivers often run at a high CPU privilege level, they are able to crash even the most stable systems with ease. In some cases device drivers need to be written in assembler rather than a higher level language, with all the development baggage that entails. In any case, writing a driver is a serious business.

The first question to ask when faced with the task of writing an NT device driver has to be whether a new driver is actually needed. There may be other tools or techniques which achieve the same end with less effort. For instance, generic device drivers available can be customised with a script. Talking to a proprietary dongle or graphics card will usually need a device driver. Before we can really tackle the issue of device driver development, we should look at an overview of Windows NT's internal architecture.

- A device driver is a trusted part of the NT kernel.
- In between the kernel and user programs is the Win32 subsystem which implements the Win32 API and manages the screen, etc.
- A Virtual DOS subsystem allows 16-bit DOS programs to run and a Windows on Windows subsystem handles all 16-bit Windows applications.
- Drivers are controlled by the I/O Manager and talk to the electronics using NT's Hardware Abstraction Layer (HAL).
- Device drivers are therefore *not* Win32 programs so the Win32 API and standard C routines such as `printf()` must not be used. Instead, drivers can call various NT kernel routine types as described in Table 1. Note that not all kernel routines have these prefixes.

See the box *Understanding hardware* overleaf for a more detailed explanation of the basic issues involved in driver design.

DDK kernel support

Both non-paged and paged memory can be allocated in various guises, depending on how crucial it is that it's available. There are in fact only minor differences between NT 3.51 and NT 4.0 device drivers specifications; NT 4.0 provides just a few extra data structures, API routines and call options.

Simple drivers will talk straight to the hardware. However more sophisticated drivers fit into a hierarchy. For example, the `parport` low-level driver exists simply to arbitrate between access requests for the parallel ports. The `parallel` printer class driver sits on top; it uses `parport` to get exclusive use of parallel port but then talks to the hardware directly. If your driver talks to the parallel port then it should use the same technique. The documentation recommends that the parallel port is just grabbed for each read and write. However, this does not seem appropriate for some applications, so you might want to allocate a port when a device is opened and release it when the handle is closed.

Other I/O areas are arranged in layers. NT's generic SCSI port driver does its job using SCSI mini-port drivers. So to write a driver for a new SCSI card, just the documented interface defined for a mini-port driver is needed. A similar approach applies to video card drivers. File system drivers, however, seem to be undocumented.

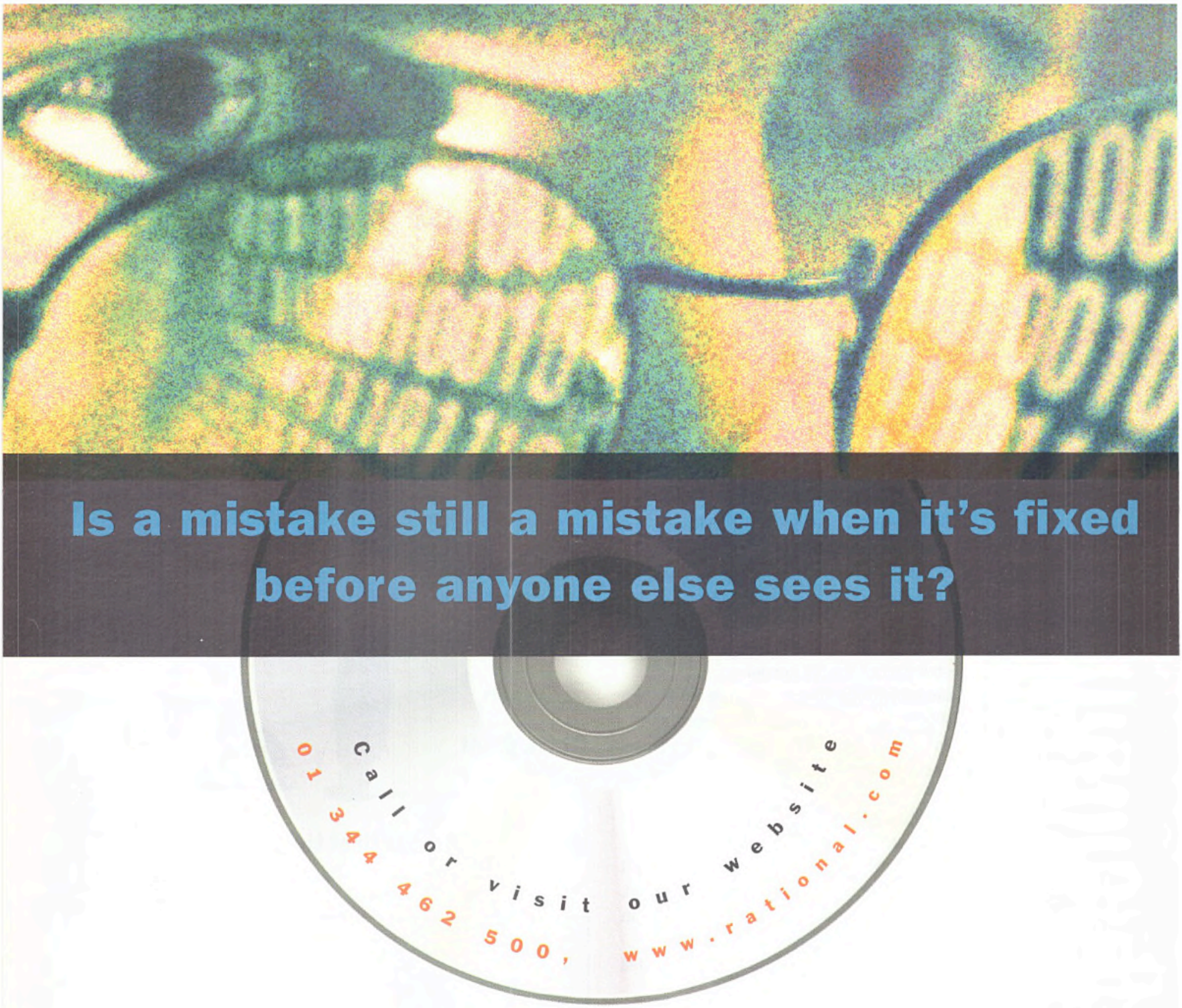
Filter drivers can sit unseen above a driver, intercepting all I/O requests. A filter driver could, for example, transparently add com-

`Ex...()`	Executive Support
`Hal...()`	Hardware Abstraction Layer
`Io...()`	I/O Manager
`Ke...()`	Kernel
`Mm...()`	Memory Manager
`Ob...()`	Object Manager
`Ps...()`	Process Structure
`Rtl...()`	Runtime Library
`Se...()`	Security Reference Monitor
`Zw...()`	Other Routines

Table 1 – Common driver call types.

IRP	Function	Win32 Call
`IRP_MJ_CREATE`	Request for a handle	`CreateFile()`
`IRP_MJ_CLEANUP`	Cancel any pending IRPs	`CloseHandle()`
`IRP_MJ_CLOSE`	Close the handle	`CloseHandle()`
`IRP_MJ_READ`	Read data from device	`ReadFile()`
`IRP_MJ_WRITE`	Write data to device	`WriteFile()`
`IRP_MJ_DEVICE_CONTROL`	Control operation	`DeviceIoControl()`
`IRP_MJ_INTERNAL_DEVICE_CONTROL`	Control operation from other drivers	
`IRP_MJ_SHUTDOWN`	System shutting down	`InitiateSystemShutdown()`

Table 2 – Some common IRP types.


Is a mistake still a mistake when it's fixed before anyone else sees it?
Call or visit our website
01344 462 500, www.rational.com

Bugs. Memory leaks. Bottlenecks. Every developer faces them. But with Rational's suite of developer diagnostic products, no one will know they existed in your code. Purify® quickly locates a broad spectrum of execution errors and memory leaks before your code leaves your desktop. Quantify® pinpoints performance bottle-necks by providing accurate, repeatable, comprehensive timing data. And PureCoverage™ identifies untested code. But don't take our word for it—the proof is in the products. For a free evaluation, simply visit our website, call us on 01344 462 500 or e-mail us at info@rational.com

ENQUIRY NO. O15

RATIONAL
SOFTWARE CORPORATION

pression or encryption to a file system or device handler without altering the underlying driver design.

To Win32, a driver appears as one or more file devices. So an application can open a device file, read and write to it, and then close the handle. A driver can also handle DeviceIoControl() requests to do any sort of I/O. Microsoft's NT drivers have various I/O control requests defined. A DongLpt driver for a dongle on the parallel port might provide devices \\.\DongLpt1 for the first parallel port, \\.\DongLpt2 for the second, etc. \\.\DongLpt1 is just a Win32 symbolic link to the real hidden NT kernel device name \Device\DongLpt0. Note that kernel device numbers are zero-based by convention. (See the panel '*DOS device support*' for information on supporting DOS and Win16 applications with your NT drivers.)

Be prepared

Arm yourself with a suitable compiler such as Visual C++ and get a Microsoft Developer Network Level 2 (or above) subscription. Install both the Platform SDK and NT DDK. The DDK examples are particularly useful as they are the source code of many of Microsoft's actual NT device drivers. They show how real device drivers are put together and can yield some undocumented features. For example, there is documentation for the parport driver interface. However only by looking at its source code will you find an extra internal device I/O control request.

Brace yourself for a development style that went out with the ark: the command line. You may also need several NT machines available for deep debugging.

Most people will just write their driver in plain C and use Microsoft's command line utility build. *Free* (retail) and *checked* (debug) versions of drivers can be built. The rebase utility will strip out all remaining symbols from a final release retail version. A Microsoft technical note says that you can write in C++ and debug from the Visual C++ IDE. I have not tested this approach.

There is a lot to learn before starting coding. Drivers need to be well designed from the start. Remember that data areas can be accessed by several different parts of your driver, which may be running at more or less the same time (or even at the same time if in a multi-processor machine). Several Win32 programs could slew off lots of overlapped I/O requests. So think re-entrant and avoid global variables. And remember that all strings must be Unicode.

A driver design might not take the expected course. The NT parallel port printer driver does not use interrupts, for example. The parallel port interrupts often conflict with other devices, so it uses a system thread to poll the printer.

Before getting started, it is worthwhile tuning your NT start-up. Cut out anything that is unnecessary as your computer will probably need quite a lot of reboots. It is easy to get device names left behind when a test driver unloads. There are quite a few alterations needed in the registry; some changes are only recognised at boot time. However, once past the initial stages, new versions of a driver can be installed using the Control Panel Drivers window. Hopefully you will not see NT's bugcheck Blue Screen of Death too often.

Do not be afraid to copy techniques from existing code: that's what all the DDK source is there for.

Driver structure

An NT driver has one standard entry point, DriverEntry(). This discovers any hardware, creates any devices and loads up a table of other entry points. Apart from the DriverUnload() routine, all the other main calls will be the result of I/O Request Packets (IRPs), discussed below. Various callback routines can be set up; as well as interrupt handlers, there are deferred procedure calls, and completion and cancel routines. A useful convention is for all a driver's routines to have a common name prefix, eg DongDispatchOpen(), DongDispatchWrite().

Understanding hardware

If you are new to device drivers then your first job will be to fully understand how the hardware works. If talking to another device at the far end of a link, try to get its source code as well. If they have not done it yet, get your hardware engineers to finalise their spec!

Standard parallel and serial ports have a well-defined interface. However, some devices at the other end of the wire may not. There are several flavours of parallel port configurable in the BIOS set up, for example unidirectional or bidirectional. Unidirectional parallel ports can be made readable by reading data 4 bits at a time via the status port.

I/O devices may be on the motherboard or on one of the various types of bus. A device may need to be serviced by the processor, or DMA transfers can be set up. Perhaps the card can even act as a bus master itself.

Hardware may or may not produce interrupts, and signals may be level sensitive or edge triggered. Allow time for data values to become stable on output lines before toggling a signal wire. Very slow inputs may need to be de-bounced. Devices have a habit of timing out, running out of paper, not being turned on, causing framing errors, CRC errors, overrunning their buffers, etc.

Plug and Play allows hardware devices to be configured automatically by the operating system. Plug and Play is not a part of NT yet so extra effort is needed to set up such devices at the moment.

For each routine, note carefully which interrupt level it runs at. All the IRP dispatch routines run at PASSIVE_LEVEL while interrupt routines run at one of the processor-specific DIRQLs. In between, deferred procedure calls run at DISPATCH_LEVEL.

There are different things you can and cannot do at each interrupt level. If running at DISPATCH_LEVEL or higher, you must not touch paged memory. And note that these routines are usually not running in the context of a user's thread. Interrupt routines need to run very quickly and so usually ask a deferred procedure call to do any post-transfer processing.

There is a data structure for each device, however most working variables will be in the associated device extension defined by the driver. Besides IRPs and devices, there are many kernel data structures. Low down, there is UNICODE_STRING and LARGE_INTEGER. Controller, adapter (DMA) and interrupt objects can only be used by one device at a time. Zone buffers, look-aside lists and linked lists are different ways of organising memory.

Use spin locks to guard access to data areas. The Cancel spin lock is used to protect access to the cancel fields of an IRP. It is a useful cheat to use this Cancel spin lock to guard data areas in all the dispatch points. A minor point to note is that the status values returned to the I/O Manager are not identical to the values Win32 programmers see, ie a non-obvious mapping occurs.

Common IRPs

As described above, a separate handler is required for each IRP. Not all IRP function codes need be implemented, but create, cleanup, close and read or write are a useful minimum (see Table 2 for a list of the common IRPs). An IRP has a header area followed by several stack locations. Each

stack location holds a function code and various parameters, eg for the read, write and device I/O control functions.

When the first driver processes an IRP, there will be only one stack location. If it passes the IRP to a lower level driver to process, then the next stack location is allocated. Note that the new stack location can have a different function code. As an example, a transport network layer driver could accept data transfers of any length. The lower level driver might have a maximum transfer size, so the transport driver will keep calling the lower level driver until all the data is sent.

Alternatively, a higher level driver can allocate whole new IRPs. The transport driver could therefore allocate all the necessary IRPs and send them all off to the lower level driver at once. Obviously, it would need to check carefully that all the IRPs completed successfully.

With buffered I/O, the I/O manager copies any user write data into non-paged memory automatically (and vice-versa). With direct I/O, the I/O manager locks the user buffer into physical memory for the duration. Direct I/O is slightly more complicated to use but has less overhead.

Installation

The driver executable should be put in the appropriate Drivers directory, eg `\WINNT\System32\Drivers\DongLpt.sys`. Various entries are needed in the registry to make NT realise a driver is there. Use REGEDT32 instead of RegEdit as it can handle all the necessary registry types.

DongLpt's main driver registry key is `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\DongLpt`. In here are several standard values, eg if `Start` has a value of 2 then the driver is loaded automatically on re-boot. A driver's `Group` and which groups and drivers must be loaded before it does can be specified. A driver may well have a `Parameters` sub-key. This could have an `ErrorLogLevel` value which indicates the level of event log messages required.

For the final cut, you can write a script file or write an installation program by hand. The Win32 Service Control Manager functions can be used to install, start and stop drivers, so a reboot will usually not be necessary.

It is best to write a driver in stages, checking that each part works before moving onto the next. First, get it to load and unload. Then find the hardware, allocate it on load and release it on unload. Make the devices and symbolic links on load and release them on unload.

Now start handling IRPs. Initially just dispatch these to be run straight away in your Start I/O routine. Then they could be put in a driver's own internal queue for processing later. Now catch interrupts. Time-outs can be caught with the basic one second I/O timer or custom timers can be set for any time interval. In NT 4.0, these can fire repeatedly, while earlier versions were one-shot.

Microsoft recommends not tying up the processor for more than 50 microseconds at a time. For lengthier interactions, consider using a design based on system threads. Event, mutex, semaphore and timer objects may be used to coordinate thread activities. Ensure that the thread is told to exit when it is not needed, as the kernel will not stop it if you forget.

For debugging, I found that writing messages to NT's event log was the easiest way to find out what the driver was up to, a bit like putting `printf` statements in. Write the event logging with the first bit of code. Messages for the event log are stored in an `.mc` file and compiled into a resource using the `mc` utility. Source code level debugging is possible between two NT machines. The development machine runs retail NT and the `windbg` utility. Debug info is passed along a serial line to the other computer running the NT checked kernel build and the test driver. Have a proper network connection available for transferring files.

DOS device support

An NT device driver can be accessed from legacy DOS or Win16 programs, provided certain rules are followed.

Only standard DOS device names can be used, eg `LPT1`, `COM1`, etc. Note that `LPT1`, etc. is output only, while `COM1`, etc. is bidirectional.

A symbolic link must be set up from a DOS device name to an NT kernel device name. NT's own parallel and serial drivers will try to allocate any appropriate device names. One simple approach is to make `COM9` link to the desired driver. This corresponding device could talk to the parallel port.

If you really need your `DongLpt` driver to allocate `LPT1` for example, then you need to stop NT allocating it. In fact, what you need to do is allocate `LPT1` before NT does, by setting up the driver group load order correctly.

The NT parallel port arbitrator `parport` driver is in the group `Parallel Arbitrator`. The parallel class driver `parallel` is one of several drivers in group `Extended Base`. If `DongLpt` loads after `parport` and `Parallel Arbitrator` but before `Extended Base` then it can reserve the name `LPT1` before `parallel` does. Note that this technique implies that the driver must start at boot time and so will reserve `LPT1` for the entire NT session. In contrast, making a link from `COM9` allows a driver to be started only when needed.

So far I have not found out how to allocate `AUX` before NT does.

Non-paged memory is a precious resource. The `alloc_text` pragma can be used to mark appropriate routines as pageable, and initialisation routines as discardable. The DDK has a test suite to check drivers in stressful situations. Once you're happy with its performance, a driver can be submitted to Microsoft's compatibility labs for certification testing.

Example – the DongLpt driver

As an example I have written some initialisation code for a `DongLpt` driver which talks to a dongle on the parallel port. This code is available on EXE OnLine or directly by ftp at *ftp.exe.co.uk/pub/exestuff/9710_NTdriver.* After initialising its event log, `DriverEntry()` sets the other entry points for the driver. For each of the parallel ports that NT has found, `DongCreateDevice()` first creates an NT kernel device. This driver uses buffered I/O. Then the device extension is initialised.

`DongCreateDevice()` then links to the corresponding NT `parport` device to retrieve information about the port. Finally, the appropriate Win32 symbolic link name is created. `DongGetPortInfoFromPortDevice()` builds a new Internal Device I/O Control IRP for `parport` to retrieve the port information. The routine uses a notification event to wait for `parport` to complete processing of the IRP. Various hardware details are stored on return, then `parport`'s routine `TryAllocatePort()` is called directly by `DongLpt` when it wants to do some I/O. `FreePort()` makes the port available again.

As intimidating a task as it might seem on first inspection, once you have prepared the ground and learned the necessary techniques, writing NT device drivers is a relatively straightforward process. Certainly a useful skill to have in your repertoire. ■

Chris Cant runs PHD Computer Consultants Ltd. You can reach him at chris@phdcc.com, or go to his Web site at http://www.phdcc.com/.

IBM
WINDOWS NT

TRACK & TRACKWeb™

from Soffront Software, Inc.

New version 4.0

Beat the Competition!

Track all of your development and technical support.

- Bug reports
- Code changes
- Test cycles
- Support calls
- New features
- Releases
- Documentation
- System configurations

There's nothing like TRACK & TRACKWeb. Save time, money and dramatically increase customer satisfaction.

Easy to submit, modify and query. Your entire team can start using TRACK without any training. Not possible? Just download the evaluation copy from www.soffront.com, install it on the LAN and let everyone try it!

SQL database support. The fast dBaseIV engine supports 100+ users and popular SQL databases such as Microsoft SQL, Oracle , Sybase.

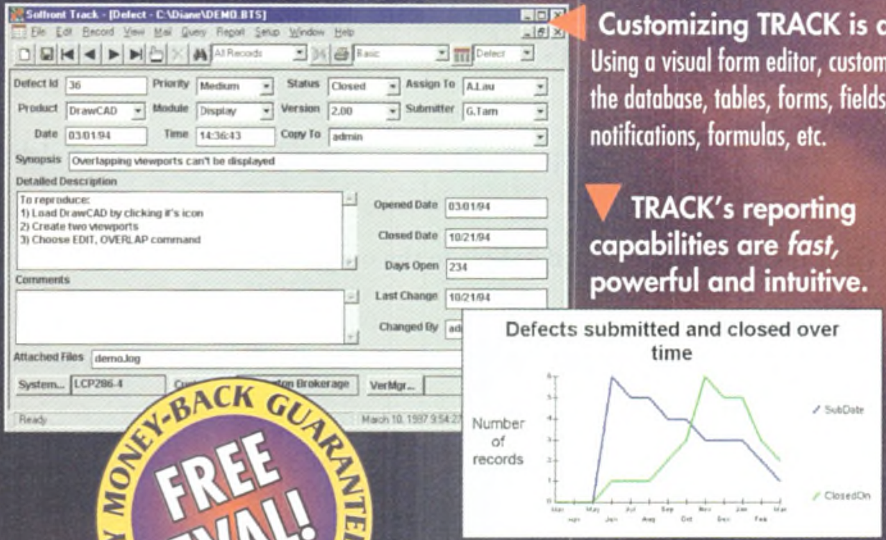
Global project management becomes reality. Get real-time interaction to geographically dispersed developers, QA and support personnel.

Links to customer database. TRACK integrates defect tracking and Help Desk.

Multiple security levels. Set up your desired level of security for commands, projects, tables, fields, data, and more!

Track code changes. TRACK integrates with popular version control systems.

Automatic notification using numerous mail systems. TRACK keeps everyone up-to-date when new bugs are entered or pending bugs are modified. You'll know when a bug was entered, who's working on it, and when it's likely to be fixed.



Soffront Track - [Defect - C:\Diane\DEMO.BTS]

Defect Id: 36 | Priority: Medium | Status: Closed | Assign To: A.Lau

Product: DrawCAD | Module: Display | Version: 2.00 | Submitter: G.Tam

Date: 03/01/94 | Time: 14:36:43 | Copy To: admin

Synopsis: Overlapping viewports can't be displayed

Detailed Description:

To reproduce:
1) Load DrawCAD by clicking it's icon
2) Create two viewports
3) Choose EDIT, OVERLAP command

Opened Date: 03/01/94
Closed Date: 10/21/94
Days Open: 234
Last Change: 10/21/94
Changed By: ad

Comments:

Attached Files: demo.log

System...: LCP286-4

Ready | March 10, 1997 9:54:27

30-DAY MONEY-BACK GUARANTEE

FREE EVAL!

Customizing TRACK is a joy. Using a visual form editor, customize the database, tables, forms, fields, notifications, formulas, etc.

TRACK's reporting capabilities are *fast*, powerful and intuitive.

Here's what the users and press say about TRACK and TRACKWeb.

What Users Say:

"The ability to get users up and running with TRACK in a very short period of time is invaluable. Most people are able to use TRACK without ever reading the documentation."
— Greg Bryant, Principal Software Engineer, NEC Technologies, Inc.

"Using TRACK we're able to look at exactly what happened on a project and come up with accurate time estimates for future releases."
— Kirsten Thompson, Quality Assurance Analyist, Pepsi International

"After using TRACK for several months, we have found that it has streamlined our problem reporting and tracking efforts on a corporate basis."
— Rory Roybal, Director of Engineering, Phoenix Technologies, Ltd.

The Press Agrees:

"TRACK automates interactions with related tools."
— Peter Coffee, PC Week, March 4, 1996

"TRACK wins hands down in the flexibility, power and ease-of-use areas."
— Data Management Review, January 1996

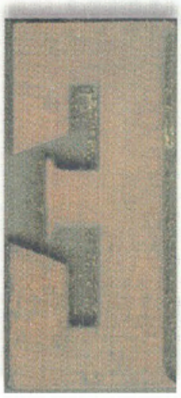
SOFFRONT® TRACK

for Windows, NT, 95 and popular Web browsers.

www.soffront.com

SOFFRONT Software, Inc. Internet: info@soffront.com • http://www.soffront.com
Ph 800-763-3766 / 408-263-2703 P.O. Box 361936, Milpitas, CA 95036 • Fax 408-263-7452

ENQUIRY NO. O17

LATX typesetting as a programming language

LaTeX is a system for typesetting documents that replaces the notion of What-You-See-Is-What-You-Get with standard ideas from conventional programming languages. Mark Harman demonstrates the power of this inheritance.

Anyone who has used a What-You-See-Is-What-You-Get (WYSIWYG) editor, wordprocessor or DTP package will have probably had two frustrations:

WYSIWYG always seems to be a lie – What you see is rather *similar* to what you get, or what you see is *almost always* what you get, or what you see *would* be what you got had your printer had the right fonts, but what you see is seldom *exactly* what you get.

WYSIWYG is restrictive – Suppose you can't get to see what you want? What you see is what you get has the additional, and tacit implication that what you cannot manage to see you also cannot get. How often have you wanted something to look slightly different (or perhaps very different) but your editor will not allow it?

LaTeX (pronounced Lateck) is a typesetting system which does not follow the WYSIWYG approach. Instead, it is inspired by programming languages. It inherits all the advantages of programming languages and some of their disadvantages. Instead of composing a document 'at the screen' (the WYSIWYG approach), a LaTeX document is a *program* which tells the LaTeX system how to create the document. The program is compiled using a LaTeX compiler to produce a document which can be printed or viewed.

This may sound a little odd to someone familiar with the WYSIWYG approach, but anyone who enjoys (or appreciates) the power and flexibility of a high-level programming language, will soon find that LaTeX is simply a better way of designing documents.

In this article, I will explain a little of the LaTeX language, enough to allow you to download a free LaTeX system and to write some normal documents. There will not be time to cover all of the features of LaTeX (this would take a whole book), but I hope to leave you with a strong feeling for ways in which writing a document could, using LaTeX, be a similar activity to writing a program.

A simple LaTeX source file

Figure 1 contains a simple LaTeX source file. The first line is a predefined LaTeX command. All commands start with a backslash character. The command in the first line establishes the global properties of the document to be typeset. The document style 'article' is the style used for a short article. Other styles include book, report, thesis and so on. Each document style changes global parameters which describe the layout of your document. For example, in the book environment, running headers are produced giving the chapter title and author on alternate pages.

Like all good programming languages, all this is, of course, entirely configurable. However, like most programming languages, the more flexibility you want, the more you will find you need to know of the underlying programming language. Fortunately the default settings for all of the LaTeX environments give very pleasing results, so it is possible to get a long way without having to know too much of the underlying language. What you get is *likely* to be what you want, and if it is not, then at least you will be able to change it.

The text of the document itself is contained within the commands `\begin{document} ... \end{document}`, `\begin{}` and `\end{}`

```
\documentstyle{article}
\begin{document}
hello world
\end{document}
```

Figure 1 – A Simple LaTeX document.

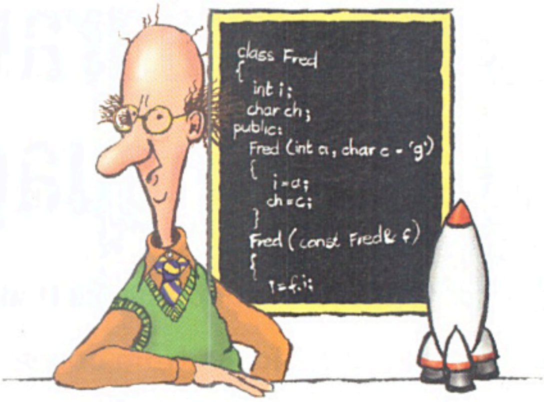
```
\documentstyle{article}
\begin{document}
\section{Introduction}
This is a fairly short document and this is its
introduction.

\section{Rationale}
The document is so short because it is simply an
example.
```

Figure 2 – Sections.

```
\documentstyle{article}
\begin{document}
\section{Introduction}
\label{intro}
This is a fairly short document and this is its
introduction.

\section{Rationale}
A brief introduction to this document can be found in
section \ref{intro}.
\end{document}
```

Figure 3 – Symbolic references.

There's C++ training...



...then there's our C++ training...

...Visual C++ Programming and VC++ Programming & Design for NT and Visual C++ for Windows and ActiveX Programming with Visual C++ and Advanced C++ Programming and COM & DCOM Programming and Java Programming for C++ Programmers and C++ Foundation and Advanced MFC Programming with Visual C++ and Java Programming and a full range of courses in Visual Basic and courses in MS Office and Delphi and Fortran and...

Richfords: A programme to suit you.

At Richfords we don't try to cover everything in IT - unlike other training companies. We stick to what we're good at: **providing focused training for Windows Developers.**

Our course leaders (who still work in the real world - and face the same challenges as you) write, deliver and maintain Richfords' own integrated courseware. And our pre-course assessment process ensures that your skills and objectives are right for the course, and shared by the class.

At Richfords, you'll find the right training solution.

For a full schedule of courses in London & Bristol call:

070 10 70 60 61

RICHFORDS

WINDOWS DEVELOPER TRAINING

South Bank Technopark, 90 London Road, London, SE1 6LN. Fax: 070 10 70 60 62

ENQUIRY NO. O18

are commands which open and close an environment. All documents (and fragments of documents are typeset within an environment. We can also nest environments, as we shall see later.

Space characters are unimportant to LaTeX; one space is a good as one hundred. New lines can also be inserted anywhere, but two or more new lines are used to denote the point at which one paragraph finishes and another starts. When you print out the final document, LaTeX will left and right justify the text (inserting hyphens, where pure justification would lead to unattractive output).

A large document usually consists of a number of sections, which may contain subsections. A new section is introduced into LaTeX using the `\section` command, and a new subsection with the \subsection command. The LaTeX source code in Figure 2 describes a document with two sections, the titles of which will be 'Introduction' and 'Rationale'. Notice that, in the source code we do not need to give the sections a section number. LaTeX will do this for us when it compiles the document. Thus, 'Introduction' will be section number 1 and 'Rationale' will be section number 2. If we were to swap the order in which the two sections occur (by cutting and pasting the source code), then 'Rationale' will become section 1, while 'Introduction' will become section 2 – automatically.

An important question now arises: How would I cross-reference from one section to another? For example, suppose I want to refer to the section 'introduction' in the section 'rationale'. The way that this is achieved illustrates the first advantage we gain from the LaTeX way of doing things.

Symbolic references

Because a LaTeX source file is a program, you can use symbolic names to refer to parts of the document. This makes cross-referencing a pleasure, as the cross-reference is a logical entity, referring to some named part of the document. If this named part of the document should be moved, then all we have to do is recompile.

To introduce a symbolic reference, we use the `\label{}` command, and to refer to it we use the `\ref{}` command. Figure 3 illustrates this. `\label{intro}` introduces a symbolic name, label, whose value depends upon the context in which the `\label` command appears. In this case, since the `\label` command is used in the first section of the document, the value assigned to 'intro' will be 1. The `\ref{}` command simply produces the value of the label. Now if I move the introduction to a new position, for instance, after the 'Rationale' section, the value of intro will change to 2, and the cross reference in 'Rationale' will thus point to the new location of the section 'Introduction'.

This style of writing forces us to think of the document at a logical level rather than at the physical level. It would be foolish to write 'as we saw earlier in section \ref{intro}' for example, because we may move the label 'intro' to a point after the reference. Instead of thinking of our document as a monolith of text occurring in a particular order, we think of it at a higher level of abstraction, as a collection of sections which we are free to move around. We can even reuse sections from one document in another, and providing our symbolic names are unique, we will find that all the cross references work out correctly.

Some more environments

LaTeX has lots of useful predefined environments. Suppose we want to produce a sequence of points using bullets. We can do this with the 'itemize' environment, the source code in Figure 4 will produce a document which lists the three principal states of matter, one per line and each preceded by a bullet point mark. In many respects the LaTeX way of designing a document is similar to the HTML way of doing things. For example, the itemize environment is rather like the unsorted list environment in HTML.

Sometimes we want to put items into a sorted, numbered list. This is achieved with the enumerate environment. Figure 5, shows a nested sequence of enumerated items, describing the four eras of geological time and the periods within them. LaTeX uses different numbering systems for each level of nesting (Arabic numerals for level one, alphabetic characters for level two, roman numerals for level three). This, as with everything else, can be changed if we so wish.

To emphasise a portion of text, it is enclosed in the 'em' (emphasise) environment, so we simply write '\begin{em} help! \end{em} she cried.' to emphasise the word 'help' (and the exclamation mark which follows it).

Procedures

In a conventional programming language, the ability to define procedures gives the programmer considerable flexibility. In LaTeX too, we can define procedures for laying out text. The simplest form of procedure is a parameterless one. It allows us to name some portion of source code and then call it up. Suppose I am writing a document in which I want to refer to an item of fruit, but I haven't yet decided whether it is to be an apple, orange or pear. I could introduce a procedure called 'fruit', and put an arbitrary fruit name in its body. When I've finally decided which fruit I want to refer to, then I will *only* have to change the body of the procedure; all the points at which the procedure is called will then automatically take account of the change to its body.

```
\begin{itemize}
\item Solid
\item Liquid
\item Gas
\end{itemize}
```

Figure 4 – The Itemize environment.

```
\begin{enumerate}
\item Cenozoic
   \begin{enumerate}
   \item Quaternary
   \item Tertiary
   \end{enumerate}
\item Mesozoic
   \begin{enumerate}
   \item Cretaceous
   \item Jurassic
   \item Triassic
   \end{enumerate}
\item Paleozoic
   \begin{enumerate}
   \item Permian
   \item Carboniferous
   \item Devonian
   \item Silurian
   \item Ordovician
   \item Cambrian
   \end{enumerate}
\item Precambrian
\end{enumerate}
```

Figure 5 – The enumerate environment.

We make it easy

Integrating document imaging technology into any Windows application does not have to be difficult. In fact with ImageControls v2.0 and StorageControls v1.0 from Kofax it can be as simple as 1,2,3.

1 Both of these low-cost developer toolkits provide you with 32-bit ActiveX controls for adding high performance document scanning, printing, viewing, image processing and storage into applications built with Microsoft's Visual Basic 4.x and C++.

2 ImageControls contains royalty-free viewing and printing technologies with the controls to drive the industry's fastest scanners at rated speed whilst simultaneously performing bar code recognition, image clean-up, image enhancement, forms recognition and more in the blink of an eye.

3 StorageControls lets you integrate sophisticated optical jukebox management into your application to make access and retrieval of large image files lightening fast on both Windows NT and Novell NetWare networks.

Once you've developed your application, we've got the Kofax Adrenaline line-up of software and hardware accelerators and Ascent Storage run times to help you get the most out of your scanners, printers and jukeboxes.

Which is why Headway Technology and Kofax Image Products lead the way in bringing the best visual programming tools for document imaging to developers throughout the United Kingdom. Give us a call today and we'll show you how easy adding production imaging into your Windows applications can really be.

KOFAX

The Component Imaging Company.

Distributed in UK by:
Headway Technology Group
Headway House, Crosby Way,
Farnham, Surrey, GU9 7XG, UK
Tel: + (44) (0) 1252 717071
Fax: + (44) (0) 1252 741223
http://www.headway.co.uk

Headway Technology Group

A MEMBER OF THE ACAL GROUP

Offices in Belgium, France, Germany, Italy, Netherlands, Norway, Sweden, UK and USA

ENQUIRY NO. 019

In LaTeX, a procedure is called a command, and a new one is created with the command `\newcommand`. Commands are often called macros because LaTeX expands calls to these when it meets them in the body of the document. Figure 6 illustrates the use of a simple parameterless macro. When compiled, the LaTeX source in this figure produces the text 'The first apple to appear will be the first apple I shall eat.'

To be entirely fair, this *could* be achieved, perhaps more simply, with a WYSIWYG wordprocessor, simply by performing a search and replace. (Of course, this would *not* have worked had the sentence been 'the first \fruit to appear is the apple of my eye'!) This is, however, only a simple example of what we can do with LaTeX macros. They really come into their own when we provide them with parameters.

Parameters

Suppose I am writing a document about array handling. I might want to describe an algorithm for finding the largest element of an array. To make the document more generic and to save retyping large sections of it, I could produce two versions, each specific to a particular programming language, for example, Basic and C. Using commands, I can avoid using the particular syntax of arrays, or at least I can capture the syntactic differences in a single command, making it far easier to adapt my document to different programming languages.

Figure 7 illustrates this. In the command definition for `\lookup`, the `[2]` tells the LaTeX compiler that the command takes two parameters, the first is referred to as `#1` and the second as `#2`. In a call to a command, the parameters are supplied one after the other in curly brackets. So the call `\lookup{S}{2}`, will produce the text 'S(2)'. This is the Basic version of the `\lookup` command. If we replace it with the version in Figure 8, then we obtain the same document, but with array references in square brackets. This is the C version of the document. Notice that the difference between the two LaTeX source documents is precisely *two characters*, namely, the two characters which make up the difference between array referencing in Basic and C.

As with programming language procedures, it is possible to call one procedure from the body of another and to use the result of a procedure call as the actual parameter to another. So, for example, we can write `\lookup{A}{\lookup{B}{1}}` which produces either the text 'A(B(1))' or 'A[B[1]]' depending upon whether we are using the Basic or the C version of the `\lookup` command.

Variables

LaTeX has its own variables, on which we can perform simple arithmetic (more advanced forms of arithmetic are possible, but addition is usually all that is required for typesetting). I will look at two simple examples of the way in which we might use variables, both of which will be familiar to programmers; the counter variable and the flag variable.

Suppose we want to include a sequence of numbered points in a document. We can use a counter variable to number each point, and write a few simple commands to control the numbering. Figure 9 illustrates this. The counter is declared using the command `\newcounter`. It is set to a specific value using the command `\setcounter`. The command `\point` is used to printout the current point number and to step the counter (to add one to its value). The command `\the<name>`, for some counter `<name>`, causes the value of the variable to be printed. This command can be used with any variable, not just those that the user has introduced, so for example `\thesection` prints out the current value of the section variable. In Figure 9 we use the `\point` command to print out three points. A nice feature of this approach is that we can vary the order in which the points occur and the numbering will change accordingly.

Now let's see how we can use variables as flags to choose what text is produced in a document. As we shall see, the combination of flags and macros allows us to write very generic documents, which can be instantiated simply by choosing a suitable value for the flag. Consider again the problem of writing a document about arrays, where we wanted two forms of the command `\lookup`, one for Basic and one for C. It would be better if we could use a flag in our LaTeX source to indicate whether the language was to be C or Basic. All that we would then have to do is to give the flag the correct value before compiling the document. Figure 10 illustrates this.

The first thing to do is to include the option '`ifthen`' in the document style declaration. This allows us to use the command `\ifthenelse` later on. Next we declare a counter variable, '`language`', which is set to 1 if the language is to be Basic and to 0 if it is to be C. The `%` symbol is used by LaTeX for comments; any text which appears after a `%` symbol (and before the end of the line) is ignored by the LaTeX compiler. Next we set the counter to 1, using the command `\setcounter{language}{1}`, so the text we shall produce will, in this case, be specialised for Basic. This specialisation is achieved using the modified version of the `\lookup` command. The new version of `\lookup` uses the built in command `\ifthenelse` to test the value of the 'language' variable. The format of an `\ifthenelse` command is

```
\documentstyle{article}
\newcommand{\fruit} { apple }

\begin{document}
The first \fruit to appear will be the first \fruit I
shall eat.
\end{document}
```

Figure 6 – Parameterless commands.

```
\documentstyle{article}
\newcommand{\lookup} [2] { #1(#2) }

\begin{document}
To find the biggest element of the array A, store the
first element, \lookup{A}{0}, in the variable b. Next
enter a loop, controlled by the variable i, starting 1
and proceeding to the end of the array. At each point in
the loop, compare element i, \lookup{A}{i}, with the
value in b. If \lookup{A}{i} is larger than b, then
assign \lookup{A}{i} to b.
\end{document}
```

Figure 7 – Parameters: Basic version.

```
\documentstyle{article}
\newcommand{\lookup} [2] { #1[#2] }

\begin{document}
To find the biggest element of the array A, store the
first element, \lookup{A}{0}, in the variable b. Next
enter a loop, controlled by the variable i, starting 1
and proceeding to the end of the array. At each point in
the loop, compare element i, \lookup{A}{i}, with the
value in b. If \lookup{A}{i} is larger than b, then
assign \lookup{A}{i} to b.
\end{document}
```

Figure 8 – Parameters: C version.

```
\newcounter{pointnumber}
\setcounter{pointnumber}{1}

\newcommand{\point} { Point \thepointnumber
\stepcounter{pointnumber} }

\point
Some text associated with one of the points

\point
Some text associated with another point

\point
```

Figure 9 – Counter variables.

`\ifthenelse{<test>}{<then_branch>}{<else_branch>}`. It behaves just like an `if` statement in a conventional programming language. If `<test>` evaluates to true, the text in the `<then_branch>` is produced, if false, the text in the `<else_branch>` is produced.

Using this flag we might write lots of commands, each of which produces the text for a particular kind of statement, the language depending upon the value of the flag counter variable. In this way we could write a *generic* document about programming and simply set the flag appropriately to produce the specialised version of the document we want.

Figure 10 shows how we might do this. We define commands which produce Basic or C syntax for array lookup (using the `\lookup` macro, as described above), array updating and, more elaborately, a command which produces the appropriate syntax for a 'for' loop. The last of these requires some further explanation.

The difference between a for loop in C and in Basic is largely syntactic, and we can use the flexibility of LaTeX to escape from these syntactic details. The command `\forloop` uses the flag counter 'language' to decide whether to lay out the four elements of the loop in Basic or C style. This allows us to write some text about array initialisation and loops, *without* having to decide which language the target document will refer to.

Notice that in the C version of the for loop syntax the curly brackets that enclose the statements of the body of the loop are written as `\{ ... \}`, rather than as `{ ... }`. This is because the curly bracket symbols already have a meaning to LaTeX, so to get it to print curly brackets we prefix them by a backslash.

In Figure 10, we set the counter language to 1, so the output produced will be for Basic. From the source code in Figure 10, LaTeX will produce the output in Figure 11. If we want to produce a document which says the same thing about C arrays, we simply have to change the line `\setcounter{language}{1}` to `\setcounter{language}{0}`. It's that easy.

Mathematics

LaTeX is often (and rightly) praised for the way in which it allows for the typesetting of complex mathematics. Many modern mathematics, computing and other science and engineering texts are typeset using LaTeX.

Mathematical text can either be laid out 'in-line', in which case it appears in the sentence in which it is typed, or in 'display mode', in which case it appears centred on a line of its own – 'displayed' as it were. All the standard mathematical symbols and forms of text are provided for using commands. As LaTeX has been around for so long and has been used developed and enhanced by so many mathematicians around the world, it is extremely unlikely that there exists any form of mathematical output that has not been catered for by someone. A quick trawl through your bookshelf will probably reveal acknowledgements to LaTeX in several computing and mathematics textbooks, as it is often used to prepare technical books, allowing the author(s) to provide camera-ready copy for their publishers.

There is also a thriving LaTeX user community which ensures that all of this valuable information is collected, maintained and updated. All LaTeX developments are entirely backwards compatible, so there's no need to worry that your documents will somehow 'go out of date'.

Reuse

I estimate that it takes between two days and one week to become productive using LaTeX. Many readers may consider this unacceptable when compared to the lead-in time for WYSIWYG editors. Certainly, if

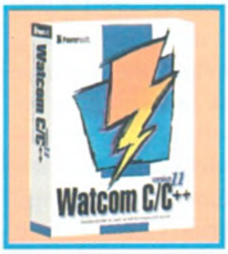
```
\documentstyle[ifthen]{article}
\newcounter{language} % set to 1 for Basic and 0 for C
\setcounter{language}{1}

\newcommand{\lookup}[2]
{
\ifthenelse{\value{language} = 1}  {#1(#2)}  {#1[#2]}
}

\newcommand{\update}[3]
{
\ifthenelse{\value{language} = 1}
        {LET #1(#2) = #3}  {#1[#2] = #3}
}

% The forloop command takes four parameters
% 1. The lower bound of the loop
%    - an integer or integral expression.
% 2. The upper bound of the loop
%    - an integer or integral expression.
% 3. The loop control variable - an integral variable.
% 4. The body of the loop - a sequence of statements.
% The flag counter language, is used to determine the
% language in which the syntax of the loop is written.
\newcommand{\forloop}[4]
{
\ifthenelse{\value{language} = 1}
{
FOR #3 = #1 TO #2

#4

NEXT #3
}
{
for(#3=#1;#3 != #2;#3++)

\{

#4

\}
}

\begin{document}
To store the value 10 in element number 3 in the array
A, we write \update{A}{3}{10}.

To initialise the elements 0 through to 10 in the array
A with the initial value 0, we can use a for loop,
starting at 0 and going up to 10. This would be written
like this

\forloop{0}{10}{i}{\update{A}{i}{0}}

\end{document}
```

Figure 10 – Flag variables.

develop to advantage

Build better applications
Add performance, functionality & versatility

CodeWright 5.0, the programmer's favourite editor, helps with tasks like; — Locating and navigating to specific code — Creating & maintaining comments — Working with version control Typing repetitive constructs —Renaming variables across hundreds of files — Working with others and their revisions. CodeWright synchronises with your Visual C++ and Delphi environments. For Windows 95 or NT and Windows 3.1 £169

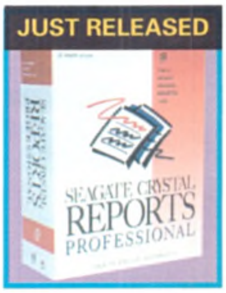


Watcom C/C++ 11.0
The high-performance, multiplatform development system.
Watcom™ C/C++ 11.0 provides a comprehensive development environment with tools, SDKs and libraries for creating powerful 16- and 32-bit applications for Windows, Windows 95, Windows NT, OS/2 Warp and more. New features include MFC 4.1 and MMX support, C++ language support for namespaces and RTTI, faster builds, and the new version of Blue Sky's Visual Programmer for Win32.

Driver::Works
Uncover the hidden world of NT Driver Development. Driver::Works includes a C++ library for NT Driver Development, extensive real-worls sample drivers, comprehensive documentation and a powerful device driver code generator. Driver::Works is designed to get new developers up to speed quickly, while providing the depth needed for the most complex driver projects. £595. Also Vtools:D device driver kit for Windows 95 –£375

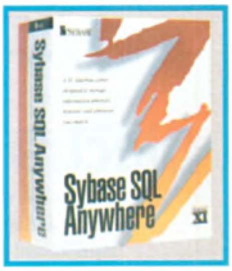
System Science News!

- **Numega DevPartner Studio** – the ideal companion to Microsoft VisualStudio 97. It contains Boundschecker for VC++ and SoftICE for all supported platforms, and CodeReview, FailSafe and SmartCheck for Visual basic - upgrades from any NuMega product for just **£375**.
- **InstallShield Java Edition** – create a single installation setup that is consistents for all Java supported platforms – a true cross platform installation package. **£call**
- Call for your **FREE** copy of our new Corporate Developer Catalogue



Crystal Reports Professional 6.0
Seagate Crystal Reports 6.0 sets the standard for Web reporting. It's access to over 30 data sources, powerful data analysis, presentation quality output, and flexibility in integrating reporting into applications has won rave reviews. Reports created with any version of Crystal reports can easily be deployed to Web servers or integrated into ActiveX Server pages for Web deployed applications. Professional 6.0 Upgrade – £189

Btrieve 6.15
Btrieve 6.15 for Netware, Windows, 3.1, NT and 95 is the de-facto standard database used by developers working with 3rd, 4th or 5th generation languages to build mission critical client/server applications. Features include multi-tasking, registry support, dynamic file expansion, true WinDLL requesters, dual mode requesters (NT and Netware), improved memory use, SFT III support, and more... Btrieve Developers Kits for Win, Win 95/NT & DOS are available. from £359



Sybase SQL Anywhere
Sybase SQL Anywhere delivers a high performance DBMS for mobile, desktop and workgroup environments. Ideal for broad deployment, SQL Anywhere requires minimal DBA support and offers interoperability with Sybase System 11. Replication technology facilitates database access for occasionally connected users. Features small footprint and easy-to-use GUI tools. Client/Server Base £197

SoftICE3
Need to debug below the surface of Windows? SoftICE 3 offers single-machine, kernel mode, multi-thread and multi-process debugging, with Win32 fault trapping and exception handling. Version 3 has support for data structures local symbols, with full support for Windows NT and Windows 95 address contexts. SoftICE NT £438; SoftICE 95 £395; SoftICE NT & 95 £499

NuMega Technologies
SoftICE3

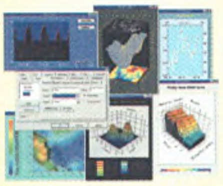
ENQUIRY NO. O20

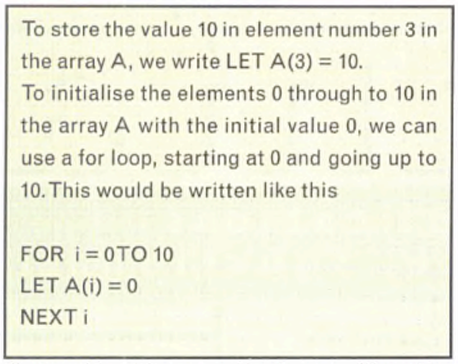
● PLEASE CALL IF THE ITEM YOU ARE LOOKING FOR IS NOT LISTED ● CALL FOR OUR COMPREHENSIVE CATALOGUE ● PRICES ARE EXCLUSIVE OF VAT ● SHIPPING TO MAINLAND UK £8.00. SAMEDAY LONDON DELIVERY AT COST ● PRICES ARE SUBJECT TO CHANGE - PLEASE CALL TO CHECK ● VISA, ACCESS, & MASTERCARD ACCEPTED WITH PHONE ORDERS

TO ORDER CALL
0171 833 1022
FAX 0171 837 6411

SYSTEM
SCIENCE
System Science, 1 Bradley's Close, White Lion St. London N1 9PN

your first choice for better development tools



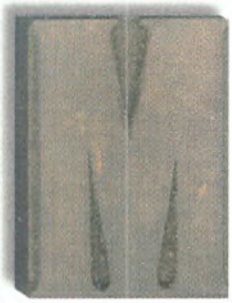


To store the value 10 in element number 3 in the array A, we write LET A(3) = 10.

To initialise the elements 0 through to 10 in the array A with the initial value 0, we can use a for loop, starting at 0 and going up to 10. This would be written like this

```
FOR i = 0TO 10
LET A(i) = 0
NEXT i
```

Figure 11 – The result of compiling the LaTeX source in Figure 10.



you only have to prepare documents such as letters and memos then LaTeX is probably not worth considering. However, if you are concerned with the production of a large amount of text and are prepared to invest in a system that could ultimately save you months of work, then LaTeX may be the answer.

One of the most intangible, yet most attractive advantages experienced by LaTeX users comes from the way in which, like a good programming language, LaTeX supports reuse. Very quickly you will find yourself building up a set of your own personal macros, which will allow you to tailor your documents to your own taste. Reusing parts of a document in another is achieved effortlessly and seamlessly. The seamlessness derives from two aspects of the LaTeX approach. The symbolic naming of parts of a document allows cross-references to be updated automatically as the document is edited. The concept of an environment means that the same piece of source text may look different when included in different contexts. Of course, this directly contradicts the WYSIWYG principle, but this is the essential strength of LaTeX. Many computing journals *[but not EXE, still battling with WYSIWYG – Ed]*, conferences and publishers provide their own LaTeX 'style files', which, when included in a LaTeX source file, automatically lay the document in the form required for publication.

Where to go next

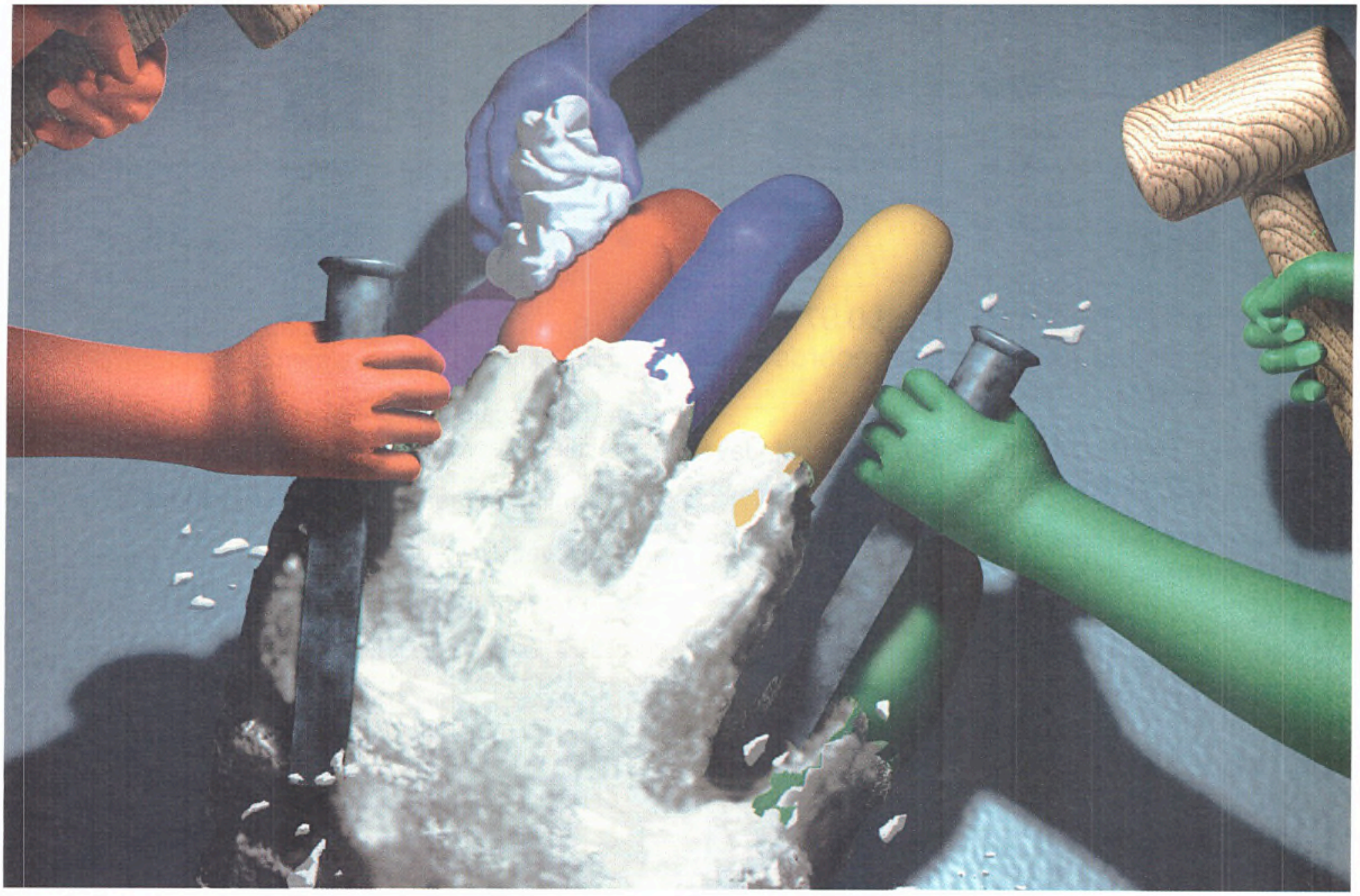
If you are interested in trying out LaTeX for yourself, versions for Windows 3.x and Windows 95/NT can be obtained (for free) from *http://www.eece.ksu.edu/~khc/tex.html.* LaTeX comes as standard on most Unix platforms, and with most Linux distributions, so if you're using one of these try typing 'man latex'. There is an FTP site containing many useful LaTeX tools, macros and related documents at *ftp.tex.ac.uk*.

There are two indispensable books on the subject of LaTeX document writing. Both are highly readable and informative. *LaTeX A Document Preparation System* by Leslie Lamport (ISBN 0-201-15790-X), describes the basic system and is a very good book to get started with. It contains enough information to write immediately most normal documents. *The LaTeX Companion* by Mike Goossens, Frank Mittelbach and Alexander Samarin (ISBN 0-201-54199-8), is more detailed, and covers all the new features added to LaTeX by the LaTeX2e project. This book is useful if you want to write a great many documents using LaTeX and to customise the language to your own tastes. It explains how to achieve all sorts of exotic effects, such as laying text out in a heart shape (perhaps useful for certain documents written just before the 14th of February). Both these books are published by Addison-Wesley.

Logical is better

The LaTeX document preparation system has evolved and improved over the years. It is extremely robust and provides features for writing documents to publishable standards containing text and mathematics. A LaTeX document is described using a programming language, which gives the LaTeX user all the power and flexibility of a conventional programming language. The style of writing forces the user to view documents at the level of their logical organisation, rather than their physical appearance. This is initially a little frustrating, but ultimately it has many advantages such as supporting reuse and creating generic documents which may have several physical instantiations. ■

Mark Harman is director of research and acting head at the School of Informatics and Multimedia Technology in the University of North London
(http://www.unl.ac.uk/~mark/welcome.html)
He can be contacted via email at m.harman@unl.ac.uk or by post to Mark Harman, Project Project, School of Informatics and Multimedia Technology, University of North London, Holloway Road, London N7 8DB.



System Architect Brings Your Company to the Future of Application Development, without Abandoning Its Past.

System Architect the Price/Performance Leader... and Your Gateway to O-O.

It single-handedly solves your company's traditional application development challenges quickly and accurately and makes the transition to Object-Oriented smooth and easy.

Using System Architect's unique Shared Repository, you can:

- use all of today's most popular modeling techniques
- share and re-use data without duplication of effort
- quickly integrate object-oriented analysis into your development projects

System Architect can be easily modified and extended to support your unique modeling and code generation requirements. System Architect deftly supports your projects, taking you to a new plateau of increased productivity and cost-effectiveness.

With System Architect, you eliminate the need for multiple products to accomplish your tasks. It's the ***off-the-shelf solution*** you can customize to fit your precise requirements.

Just Beginning Your Development Project?

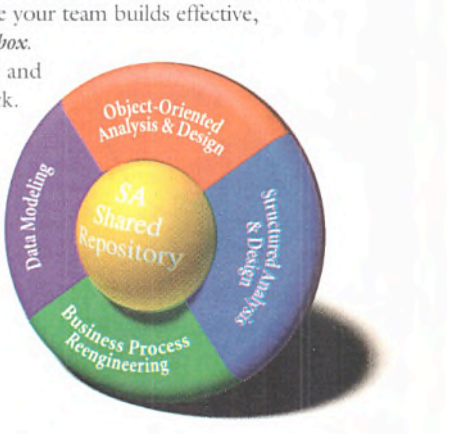
SA/Process Manager sketches the ultimate blueprint. It walks you through each stage of application development — from planning to implementation. SA/Process Manager guides you through structured analysis and design, data modeling, and object-oriented projects with clear, step-by-step advice on what to do, how to get it done, and when to do it. Customize SA/Process Manager to suit your needs, and tailor it to your unique development scenario.

Get Your Hands on the Best Family of Enterprise Development Tools Available Today.

Since 1986, Popkin Software has engineered high performance, low-cost software tools for the public and private sectors. Our award-winning software supports today's most popular methodologies.

Popkin Software supports your enterprise development needs by employing all major BPR, structured analysis and design, data modeling, and object-oriented methods in one extendable multi-user repository. And we provide comprehensive training to ensure your team builds effective, lasting solutions ***right out of the box.***

Call Popkin Software today and receive our free information pack. For more information, or to place an order, call 01926 450858. Visit our site on the World Wide Web at http://www.popkin.com.


Object-Oriented Analysis & Design
Structured Analysis & Design
Business Process Reengineering
Data Modeling
SA Shared Repository

ENQUIRY NO. O22

POPKIN SOFTWARE & SYSTEMS, LTD.

Real Tools For The Real World.

Now with Internet Model Generation.

Sequiter Software

Database Management for Programmers

CodeBase

Database Management for Programmers

Query a million records in just 0.97 of a second, or add ten thousand records in just 2.17 seconds.

- Works with C, C++, Visual Basic, Delphi, Java and soon OLE DB.
- New ActiveX data-aware controls help reduce developent times.
- Runs under Windows 95, NT, 3.1, CE, DOS, Mac, OS/2, Solaris, SunOS, AIX, SCO, Linux, UnixWare, DEC, Alpha, BSDI.

★1991-94★ DATA BASED ADVISOR Readers Choice Award

★★ 1995 ★★ Readers Choice DATA BASED ADVISOR MAGAZINE

- FoxPro, Clipper and dBASE file support including full multi-user compatibility.
- Small library size helps applications load quickly and use less memory.
- Fully scalable from stand-alone to client/server, without code changes.
- Unlimited-seat client/server included.
- Royalty free distribution.
- Full featured report writer included.

"CodeBase gives ACT! the fast database access contact management users need" -
Michael Plasterer, Director of Development, Symantec

Tel: 0181 316 5001
Fax: 0181 316 6001
http:\\www.highlander.co.uk

HIGHLANDER
PROFESSIONAL SOFTWARE DISTRIBUTION
SOFTWARE LTD

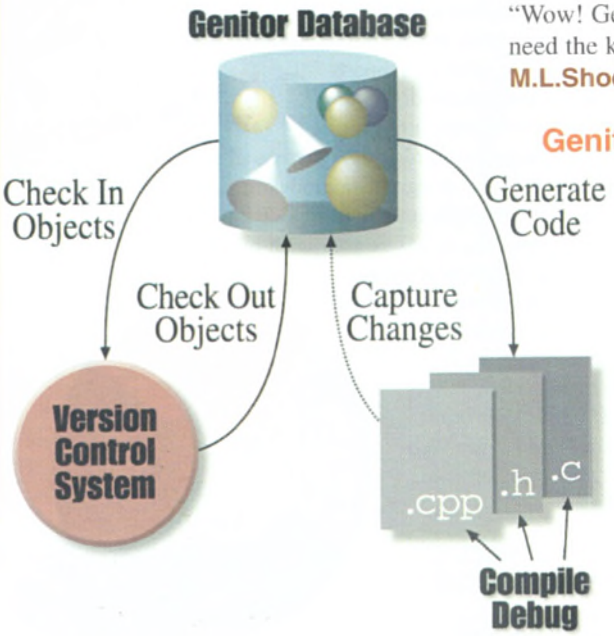
CIRCLE NO. 023

Rapid Object Development

Genitor - The Most Effective Way to Implement, Share, and Reuse C/C++ Code.

"Take a look at Genitor, and I predict you'll be impressed with how it focuses your attention on what's truly important in C++ programming."
Warren Keuffel, Software Development, April '97

"Wow! Genitor is an idea who's time has been coming for a long time. Programmers need the kind of tools that writers and accountants take for granted."
M.L.Shoemaker, Computer Scientist, En Garde Software

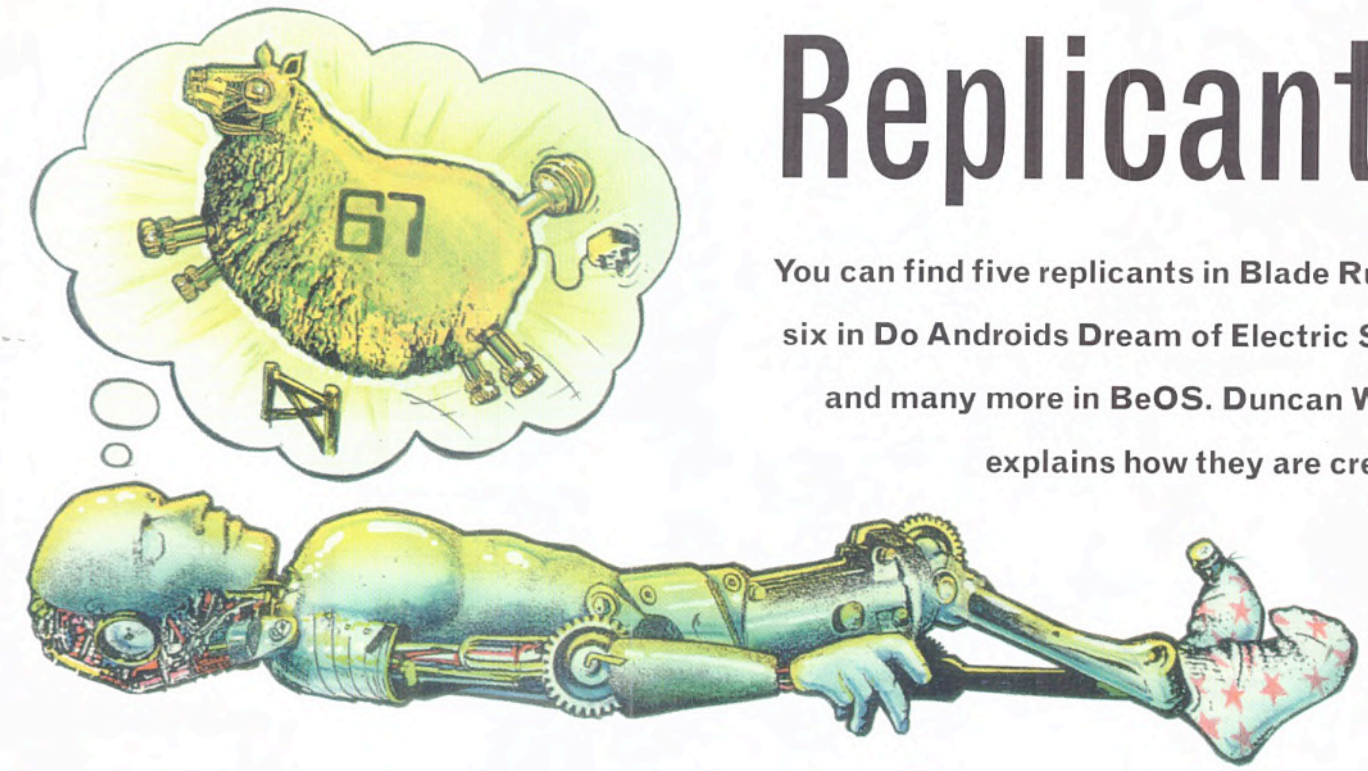


Genitor Lets You...

- Rapidly construct any C/C++ class, function, struct, union or template.
- Think at a higher level by focusing on context rather than syntax.
- Quickly locate and reuse objects developed by any member of your project team.
- Produce cleaner, more reliable code that you'll be proud to distribute.
- Instantly create up-to-the-minute documentation.
- Simplify version management and production turnover procedures.
- Boost the productivity of your existing IDE.

Tel: 0181 316 5001
Fax: 0181 316 6001
http:\\www.highlander.co.uk

HIGHLANDER
PROFESSIONAL SOFTWARE DISTRIBUTION
SOFTWARE LTD



Replicants

You can find five replicants in Blade Runner, six in Do Androids Dream of Electric Sheep and many more in BeOS. Duncan Wilcox explains how they are created.

Replicant technology is a surprisingly simple object embedding feature of BeOS. In this article we'll dissect a replicant object, a replicant container and the protocol that lets you move objects across address spaces and around applications.

Visually a replicant is a view within an application which 'belongs' to another application. From an implementation point-of-view, it's a framework and a set of rules for archiving views and passing them to other applications where they can be unarchived.

The problems that replicant technology solves include archiving an object in a package, moving the package across address spaces to a user-specified target application, locating the object implementation executable, instantiating a C++ object of a class unknown at compile-time, initialising the object with the information in the package and adding it to the target application's views.

The foundation for replicants is the object archiving framework, built around the `BArchivable` class. Archiving an object is the process of saving the minimum amount of information needed to get back to the same state when unarchiving. The `BMessage`, omnipresent information vehicle of BeOS, is the package that will contain the state information.

Every class that might have to be archived and unarchived should have `BArchivable` among its ancestors and override three `BarchivAble` methods: `Archive` to support archiving, `Instantiate` and a constructor accepting a `BMessage` to support unarchiving.

Archiving an object

Archiving an object is simply a matter of creating a `BMessage` and passing it to the object's `Archive` method:

```
BMessage data;
object->Archive(&data);
```

The first information an object should store in the `BMessage` is the data of the parent class, by explicitly calling its `Archive` method. Every class will call its parent `Archive` method, walking up the inheritance tree to the `BArchivable` version of `Archive` which only stores the object's class name.

The class name is acquired through C++ run time type information (RTTI). The Metrowerks compiler only associates RTTI with classes that have a `vtable`, but this is not a problem since `BArchivable` has a few virtual methods and hence a `vtable`, so all of its descendants will have one too.

Class specific data should then be stored in the `BMessage`. Since the data is stored with a descriptive string, just like for global C/C++ namespace, care has to be taken to avoid conflicts in the `BMessage` data namespace. Be's kits prepend an underscore to the names of all data they archive, you should choose a unique convention to prevent this problem.

A replicant is required to store the MIME signature of the executable containing the implementation of the object under the `add_on` name. The signature must be the same as that of the executable (set through the Filetypes preferences). If the executable is an application, the same signature will be used for the `BApplication` constructor.

A typical `Archive` method will look like:

```
long Replicant::Archive(BMessage *data,bool deep) const {
   inherited::Archive(data, deep);
   data->AddString("add_on",
                "application/x-my-container");
   return B_OK;
}
```

The `deep` parameter (which defaults to true) is a flag that signals if the archiving should include other owned objects that are not part of the object being archived (for example sub-views of a view).

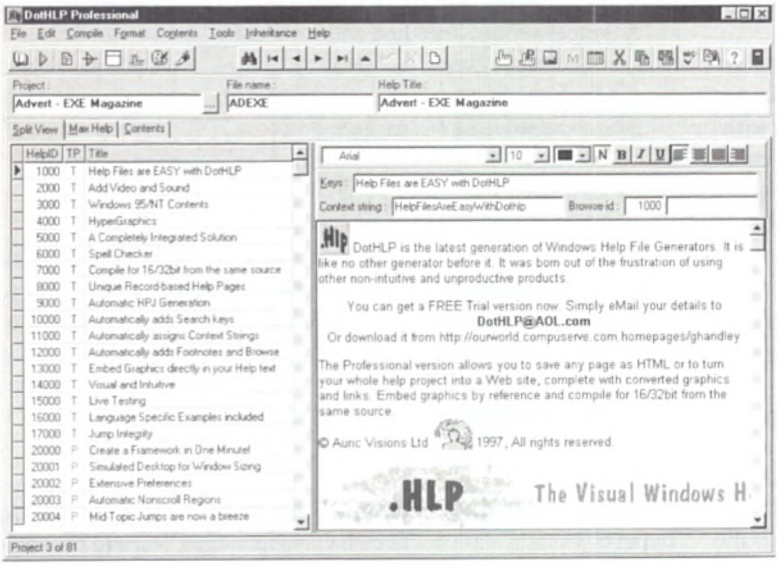
At this point, almost all pieces are in place. The `BMessage` will be *flattened* to cross machine boundaries or to be stored persistently, or it will be directly sent to another application.

Instantiating an object

This is the complex part. Starting from a `BMessage` we have to build a living object of a class not known at compile-time. Normal C++ object instantiation is through the `new` operator followed by the class name and constructor arguments. The compiler will roughly translate this to a memory allocation and constructor invocation for the specified class.

However, this approach requires the class name to be known and fixed at compile-time, because the compiler must know which constructor to call. The solution, sometimes called a *factory* function, is

Illustration: Carl Flint



DotHLP Professional

File Edit Compile Format Contents Tools Inheritance Help
Split View | Max Help | Contents

Project : Advert - EXE Magazine
File name : ADEXE
Help Title : Advert - EXE Magazine

HelpID	TP	Title
1000	T	Help Files are EASY with DotHLP
2000	T	Add Video and Sound
3000	T	Windows 95/NT Contents
4000	T	HyperGraphics
5000	T	A Completely Integrated Solution
6000	T	Spell Checker
7000	T	Compile for 16/32bit from the same source
8000	T	Unique Record-based Help Pages
9000	T	Automatic HPJ Generation
10000	T	Automatically adds Search keys
11000	T	Automatically assigns Context Strings
12000	T	Automatically adds Footnotes and Browse
13000	T	Embed Graphics directly in your Help text
14000	T	Visual and Intuitive
15000	T	Live Testing
16000	T	Language Specific Examples included
17000	T	Jump Integity
20000	P	Create a Framework in One Minute!
20001	P	Simulated Desktop for Window Sizing
20002	P	Extensive Preferences
20003	P	Automatic Nonscroll Regions
20004	P	Mid Topic Jumps are now a breeze

Arial 10
Keys : Help Files are EASY with DotHLP
Context string : HelpFilesAreEasyWithDothlp Browse id : 1000

.Hlp DotHLP is the latest generation of Windows Help File Generators. It is like no other generator before it. It was born out of the frustration of using other non-intuitive and unproductive products.

You can get a FREE Trial version now. Simply eMail your details to
DotHLP@AOL.com
Or download it from http://ourworld.compuserve.com.homepages/ghandley

The Professional version allows you to save any page as HTML or to turn your whole help project into a Web site, complete with converted graphics and links. Embed graphics by reference and compile for 16/32bit from the same source.

© Auric Visions Ltd 1997, All rights reserved.

.HLP The Visual Windows H

Project 3 of 81

BeOS status

The BeOS Preview Release for BeBox, Macintosh and other PowerPC platforms was announced at the beginning of August, check Be's Web site for more details at *http://www.be.com.*

At the Be Developer's Conference (August 4-5) and at the following MacWorld, Be demonstrated a version of BeOS running on Intel processors, scheduled for release at the beginning of 1998

to move the instantiation where the class is known at compile-time, namely in the replicant object class itself. The factory function will have to return a pointer to an object of a class that descends from a known class in the calling code. C++ will take care of casting the pointer so that the calling code gets something meaningful.

`Instantiate` is the factory function for `BArchivable`, and is declared in `BArchivable` itself as returning a pointer to an object of class `BArchivable`, that is certainly known by code calling `Instantiate`. Since `Instantiate` is a static method of `BArchivable`, it can be called without an implicit `this` pointer and hence without compile-time knowledge of the class it refers to.

`Instantiate` must be exported from the executable containing the replicant, for example with `#pragma export on` before and `#pragma export reset` after the method definition (making sure the compiler switches are set to follow the `#pragma` based export conventions).

The archived `BMessage` contains a way of getting to the object implementation: the executable signature and the class name are stored under the `add_on` and `class` names. A helper function of the Support Kit called `find_instantiation_func` will load the executable and return a pointer to a factory function of the correct class.

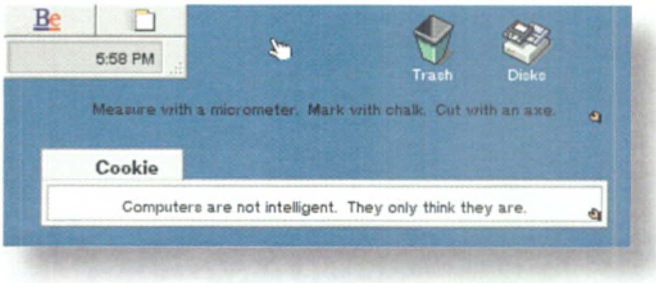
What `find_instantiation_func` has to do is deal with the symbols exported by the executable, and with C++ name mangling. Metrowerks' compiler conforms to the draft ANSI C++ name mangling standards, so an `Instantiate` method accepting a `BMessage *` of a class called `Replicant`, would be mangled as:

```
Instantiate__9ReplicantFP8BMessage
```

where `F` means it's a function and `P` means what follows is a pointer. This is the symbol `find_instantiation_func` would look for (in the executable owning the specified signature) if passed a `BMessage` containing an archived object of class `Replicant`.

Name mangling, in this circumstance, helps in allowing for more than one replicant in one executable, since the mangled symbol includes the class name and `Instantiate`, methods of different classes will have different mangled symbols.

These are excerpts from lists of exported symbols, obtained by running the `listimage` tool against Pulse, Clock and NetPositive, three applications provided with BeOS that have replicant portions:



Measure with a micrometer. Mark with chalk. Cut with an axe.

Cookie

Computers are not intelligent. They only think they are.

Figure 1 – Fortune cookies can be placed on any shelf.

Pulse:

```
TEXT Instantiate__9CPUButtonFP8BMessage
```

Clock:

```
TEXT Instantiate__13TOnscreenViewFP8BMessage
```

NetPositive:

```
TEXT Instantiate__8TResizerFP8BMessage
TEXT Instantiate__8TDraggerFP8BMessage
TEXT Instantiate__8HTMLViewFP8BMessage
TEXT Instantiate__12ProgressViewFP8BMessage
```

A typical `Instantiate` for a class called `Replicant` will look like:

```
Replicant *Replicant::Instantiate(BMessage *archive) {
  if(validate_instantiation(archive, "Replicant"))
    return new Replicant(archive);
  return 0;
}
```

The `validate_instantiation` simply checks that the class name in the `BMessage` matches the one being constructed. The constructor that accepts a `BMessage` is the third `BArchivable` method that should be implemented to build the object from the essential, archived data.

Another helper function, `instantiate_object`, will automatically find the correct `Instantiate` method through `find_instantiation_func` and call it, getting in return a descendant of `BArchivable`. Of course if the original executable disappears, the instantiation will gracefully fail and the `BView` won't be shown.

We have gone from a live object to another live object, passing through a representation (the `BMessage`) that can be moved across address spaces, across a network or stored on disk.

Draggers and shelves

Adding an `Archive`, `Instantiate` and a special constructor to a class is pretty simple. To make interaction with the system even simpler, two additional classes in the BeOS hierarchy complete the picture by letting the user drag a `BView` derived class to other applications.

The `BDragger` is a little widget that looks like a hand, and is designed to be attached to a replicant view (remember to select Show Replicants from the Be menu to show the `BDraggers` and be able to use Replicants). When the user clicks on the dragger, the dragger will create a `BMessage` of type `B_ARCHIVED_OBJECT`, fill it via the `Archive` method of the view it's attached to, and prepare for a drag. The following line will add a dragger to the view, attaching it to the specified target view:

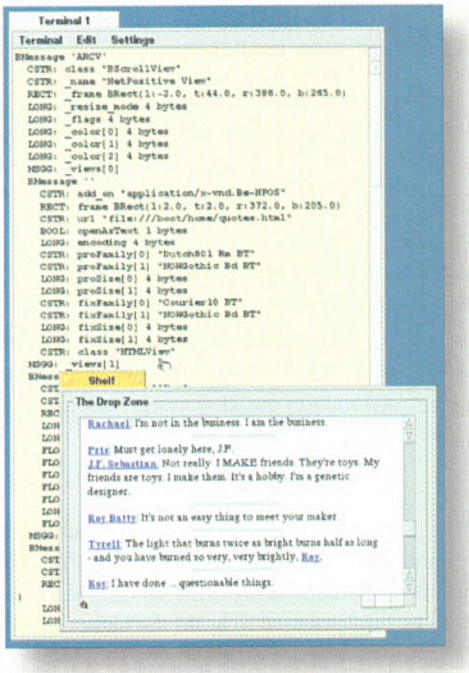
```
view->AddChild(new BDragger(bounds, target));
```

Vice-versa, instantiation of a replicant in a container can be simplified by attaching a `BShelf` to a container view. The `BShelf` which derives from a `BHandler` (basically a participant in a message handling hierarchy), will watch for `B_ARCHIVED_OBJECT` messages.

When it receives a `B_ARCHIVED_OBJECT`, the `BShelf` will go through the instantiation process, including attempting to cast the `BArchivable` object returned by `instantiate_object` to a `BView` (using RTTI and the C++ `dynamic_cast` operator) and adding the resulting `BView` to the view it's attached to. The simplest way to attach a shelf to a `BView` is through something like:

```
new BShelf(view);
```

A `BShelf` can also be associated with a stream thus trivially adding persistence to contained objects. Finally a `BShelf` descendant can override some methods that allow it to selectively reject dropped replicants based on their `BMessage`, for example when they are oversized.



```
BMessage 'ARCV'
  CSTR: class "BScrollView"
  CSTR: _name "NetPositive View"
  RECT: _frame BRect(l:-2.0, t:44.0, r:386.0, b:265.0)
  LONG: _resize_mode 4 bytes
  LONG: _flags 4 bytes
  LONG: _color[0] 4 bytes
  LONG: _color[1] 4 bytes
  LONG: _color[2] 4 bytes
  MSGG: _views[0]
  BMessage ''
    CSTR: add_on "application/x-vnd.Be-NPOS"
    RECT: frame BRect(l:2.0, t:2.0, r:372.0, b:205.0)
    CSTR: url "file:///boot/home/quotes.html"
    BOOL: openAsText 1 bytes
    LONG: encoding 4 bytes
    CSTR: proFamily[0] "Dutch801 Rm BT"
    CSTR: proFamily[1] "MONGothic Bd BT"
    LONG: proSize[0] 4 bytes
    LONG: proSize[1] 4 bytes
    CSTR: fixFamily[0] "Courier10 BT"
    CSTR: fixFamily[1] "MONGothic Bd BT"
    LONG: fixSize[0] 4 bytes
    LONG: fixSize[1] 4 bytes
    CSTR: class "HTMLView"
  MSGG: _views[1]
  BMess
    CST
    CST
    REC
    LON
    LON
    FLO
    FLO
    FLO
    FLO
    FLO
    LON
    FLO
  MSGG:
  BMess
    CST
    CST
    REC
}
    LON
    LON
```

Rachael: I'm not in the business. I am the business.

Pris: Must get lonely here, J.F.
J.F. Sebastian: Not really. I MAKE friends. They're toys. My friends are toys. I make them. It's a hobby. I'm a genetic designer.

Roy Batty: It's not an easy thing to meet your maker.

Tyrell: The light that burns twice as bright burns half as long - and you have burned so very, very brightly, Roy.

Roy: I have done ... questionable things.

Figure 2 – NetPositive is a very sophisticated replicant, with a few nested views. Part of the dump of the first (and most interesting) view is shown in the Terminal window.

A simple replicant

Cookie is a simple replicant that shows fortune cookies in a window. The full code of the application is available on EXE OnLine and directly via ftp at *ftp://ftp.exe.co.uk/pub/exestuff/9710_Replicants*. A `BDragger` attached to the `Cookie` view object lets the user drag the cookie to the desktop, to the demo Container application or to any other `BShelf`. Double clicking in the view will load another fortune cookie.

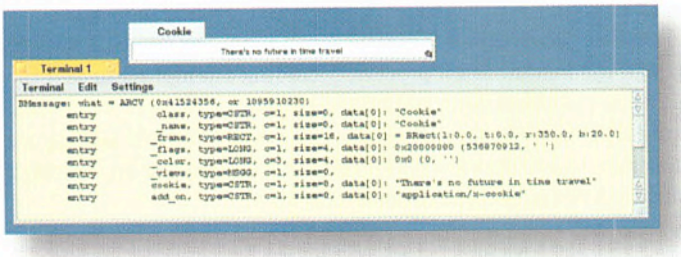
The `Cookie` class code is very straightforward. The three `BArchivable` methods are implemented to support replication. A regular constructor and destructor are implemented for use within the application. The 'engine' is the `NewCookie` method that uses Posix-like code to fetch a random fortune. Finally the `AttachedToWindow`, `Draw` and `MouseDown` methods make up the user-interface.

A `listimage` of the Cookie executable shows that the instantiation factory function is correctly exported:

```
TEXT Instantiate__6CookieFP8BMessage
```

Dumping the message contents at the end of `Cookie::Archive`, shows what is going to be sent to the container the user chooses:

```
BMessage B_ARCHIVED_OBJECT:
   class="Cookie"
   _name="Cookie"
```



```
BMessage: what = ARCV (0x41524356, or 1095910230)
        entry         class, type=CSTR, c=1, size=0, data[0]: "Cookie"
        entry         _name, type=CSTR, c=1, size=0, data[0]: "Cookie"
        entry        _frame, type=RECT, c=1, size=16, data[0] = BRect(l:0.0, t:0.0, r:350.0, b:20.0)
        entry        _flags, type=LONG, c=1, size=4, data[0]: 0x20000000 (536870912, ' ')
        entry        _color, type=LONG, c=3, size=4, data[0]: 0x0 (0, '')
        entry        _views, type=MSGG, c=1, size=0,
        entry        cookie, type=CSTR, c=1, size=0, data[0]: "There's no future in time travel"
        entry        add_on, type=CSTR, c=1, size=0, data[0]: "application/x-cookie"
```

Figure 3 – This is the message containing the archive cookie, built when the user clicks on the dragger. It will be sent to the target shelf with little additions.

```
   _frame=BRect(0, 0, 350, 20)
   _flags=0x20000000
   _color[0]=0x0
   _views=Message
   cookie="There's no future in time travel"
   add_on="application/x-cookie"
```

The names preceded by an underscore are, as mentioned, data added by the BeOS kits, `BView` contents in this case. `_color` is an array of values and only the first colour is shown, while `_views` is a flattened `BMessage` containing archived sub-views because of a deep archiving, here it's the `BDragger`.

Exploring replicants

A simple change to the Container application, shipped with BeOS in the `optional/sample-code/container/` folder will let you see what real world replicants (like NetPositive) save in the archived `BMessage`.

The patch consists in adding a message filter just after the creation of the of the Container's `TWindow`, with:

```
w->AddCommonFilter(new BMessageFilter(
   B_ANY_DELIVERY, B_ANY_SOURCE, printfilter));
```

and integrating the code in listing 2 that recursively dumps messages of type `B_ARCHIVED_VIEW`. This is useful since the deep archiving stores sub-views within sub-messages.

Pitfalls

A few common pitfalls will make a replicant simply not work or behave erratically.

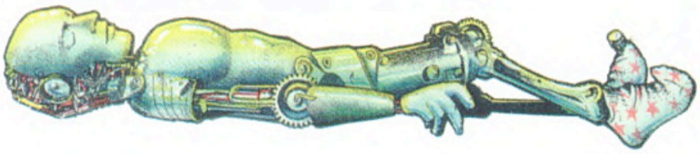
The first problem is not exporting correctly the `Instantiate` symbol, and to be sure of this both `listimage` and `pefdump` are enough. When the mangled `Instantiate` symbol doesn't show up it's usually a matter of fixing the options in the CodeWarrior 'PEF' settings page and relinking.

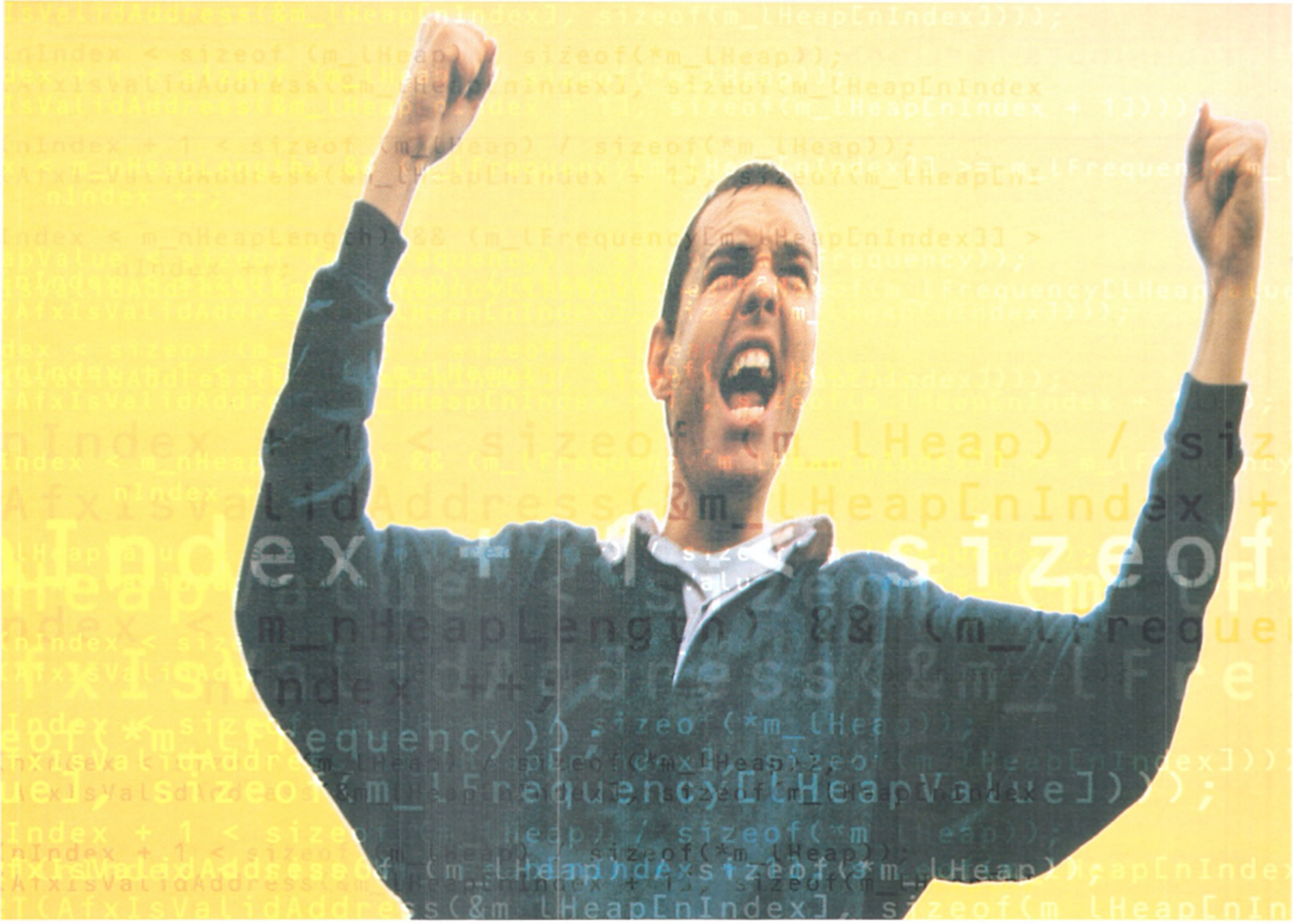
The second problem is not setting the correct MIME type signature to the executable. Notice that the file type (set in the 'PPC Project' page of CodeWarrior) must always be `x-be-executable`. The signature is set through the Filetypes application in the preferences folder.

The procedure that usually work is to link the application without resources, drag the application on the Filetypes application and set the signature exactly like the one used in the `Archive` method, save the settings and also save to a resource file, and then include that resource file in the project. On subsequent links, the signature should be correctly set.

One way to check that the signature is correctly set is to use the following program (I call it `Launch`):

```
main(int argc, char **argv) {
   BApplication app("application/x-sample-launch");

   if(argc == 2)
     be_roster->Launch(argv[1]);

   app.PostMessage(B_QUIT_REQUESTED);
   app.Run();

   return 0;
}
```




nIndex + 1 < sizeof(m_lHeap) / sizeof(*m_lHeap));
AfxIsValidAddress(&m_lHeap[nIndex + 1]
m_nHeapLength) && (m_lFrequency
sizeof(*m_lHeap));
&m_lFrequency[lHeapValue]));

```
const char *signature = "application/x-cookie";

class Cookie : public Bview {
   char *cookietext;

public:
   Cookie(BRect r) : BView(r, "Cookie", 0, B_WILL_DRAW),
         cookietext(0) { NewCookie(); }

   Cookie(BMessage *data) : BView(data) {
      char *str;
      if(data->FindString("cookie", &str) == B_OK) {
         cookietext = new char [strlen(str) + 1];
         strcpy(cookietext, str);
      }
      else NewCookie();
   }

   ~Cookie() { delete [] cookietext; }

   long Archive(BMessage *data, bool deep) const {
      inherited::Archive(data, deep);
      data->AddString("cookie", cookietext);
      data->AddString("add_on", signature);
      return B_OK;
   }

   static Cookie *Instantiate(BMessage *data);

   // ... code for NewCookie(), AttachedToWindow(),
   // Draw(BRect r), and MouseDown(BPoint) ...
};

#pragma export on
Cookie *Cookie::Instantiate(BMessage *data) {
   if(validate_instantiation(data, "Cookie"))
      return new Cookie(data);
   return 0;
}
#pragma export reset

class CookieWin : public Bwindow {
public:
   CookieWin(BRect r)
    : BWindow(r, "Cookie", B_TITLED_WINDOW,
       B_NOT_RESIZABLE | B_NOT_ZOOMABLE) {
      r.OffsetTo(B_ORIGIN);
      Cookie *v = new Cookie(r);
      AddChild(v);
      r.top = r.bottom - 7; r.left = r.right - 7;
      v->AddChild(new BDragger(r, v));
   }
   // ... code for QuitRequested() ...
};

int main(int, char **) {
   BApplication Cookie(signature);
   (new CookieWin(BRect(50, 50, 400, 70)))->Show();
   Cookie.Run();
   return 0;
}
```

Listing 1 – Portions of the Cookie replicant.

and trying to open the application that contains the replicant through its signature, for example:

`Launch application/x-cookie`

If the application doesn't open, chances are the MIME signature is not correctly set, and anyway `find_instantiation_func` won't be able to open it either.

The most important thing to consider, since it influences the design of the replicant, is that the view will be running out of the context of the original application, and it shouldn't rely on any global variable being initialized, on sibling or parent views existing or still on the application correctly handling messaging.

This is important if, for example, the replicant includes system objects that are derived from `BInvoker`. Objects derived from `BInvoker` can target a `BLooper` or `BHandler` and send a message, typically as the result of user interaction. An example is the `BButton`, that will send a message when the user presses it.

```
void dumpmsg(int32 level, BMessage *message) {
   const char *indent = "                              ";
   printf("%.*sBMessage '%.4s'\n", 2 * level,
      indent, &message->what);
   int32 max = message->CountNames(B_ANY_TYPE);
   int32 i;
   for(i = 0; i < max; i++) {
      char    *name;
      type_code t;
      int32   found;

      message->GetInfo(B_ANY_TYPE, i, &name, &t, &found);

      for(int32 j = 0; j < found; j++) {
         void *data;
         ssize_t size;
         message->FindData(name, t, j, &data, &size);
         printf("%.*s%.4s: %s",
            2 * (level + 1), indent, &t, name);
         if(found > 1)
            printf("[%d]", j);

         switch(t) {
            case B_STRING_TYPE :
               printf(" \"%s\"\n", data);
               break;
            case B_MESSAGE_TYPE : {
               printf("\n");
               BMessage nested;
               message->FindMessage(name, j, &nested);
               dumpmsg(level + 1, &nested);
               }
               break;
            case B_RECT_TYPE :
               printf(" ");
               ((BRect *)data)->PrintToStream();
               break;
            default :
               printf(" %d bytes\n", size);
               break;
         }
      }
   }
}

filter_result printfilter(BMessage *message,
   BHandler **, BMessageFilter *) {
   if(message->what == B_ARCHIVED_OBJECT)
      dumpmsg(0, message);
   return B_DISPATCH_MESSAGE;
```

Listing 2 – A recursive message dumping function and a filter.

The `BInvoker` lets you customise the message content and target, so you should perform a

`SetTarget(this);`

in the view's `AttachedToWindow` hook.

Components!

Archiving an object is a powerful feature that isn't limited to replicants, although that's where it is more visible.

Interface generators are available that let developers generate the user-interface in a visual way and then simply unarchive it from their application's resource.

To create end-user applications through the composition of replicants, another element is required: the scripting architecture. Scripting should be completed and documented by the time you read this, and is necessary for replicants to learn about each other's features and connect with high level semantics.

The simplicity of the archiving process and of the dragger/shelf interface will hopefully encourage all developers to write replicant savvy applications, where the integration will make a component based application much more powerful than the sum of its parts. ■

Duncan Wilcox is a freelance consultant and programmer. You can contact him at duncan@mclink.it.

Zinc 5.0
Now Shipping
For Free Trial Browse
WWW.Zinc.com

Portability everyone can agree on

"Zinc's portability is transparent. And only Zinc has a full set of UI objects and real extensibility to develop commercial-grade applications. Full source code is a big plus, too."

DEVELOPMENT MANAGER

"We're now shipping our application on every major platform and in every major world market. Our sales force loves it! With Zinc we're beating the competition."

MARKETING MANAGER

"Zinc cost us very little up front and saved us years of development expense. Revenue is up, thanks to the new markets [Zinc enabled us to enter]. I'm very satisfied."

FINANCE MANAGER

When was the last time Development, Marketing, and Finance all agreed on anything? Now's your chance to make it happen. With Zinc you'll build better applications, on more platforms, in less time, and with less money—and that's a promise.

Only Zinc offers complete portability.

Since Zinc Application Framework is the only cross-platform tool that delivers 100% portability, you'll have your applications on other platforms as fast as you can recompile. And no one supports as many platforms as Zinc does. It's all part of what makes Zinc the most productive—and affordable—tool you can own.

Productivity that leads to opportunity.

Zinc zips through tedious tasks with C++ object orientation and a unique visual development tool. And, enabling your application for international markets is already done—just translate your text. Plus, Zinc is the only tool that supplies 100% of the source code.

It all adds up to productivity. Which means more profitability. Which means everybody's happy—especially you.

PCWEEKLABS
ANALYST'S CHOICE
JAN. 30, 1995

"Developers seeking easy delivery of GUI applications... will find Zinc their best option by far."

"This product is absolutely the best development environment I have personally seen for the international engineer."

Multilingual Computing

EDITOR'S CHOICE '95
MLC

"Best Portability"
INFOWORLD
February 6, 1995

"Zinc came closest of all the products we tested to our ideal of portability... In short, Zinc did a great job."

For free demonstration software and an information packet, please call:

+44 (0) 181 855 9918

USA: +1 801 785 8900 or fax +1 801 785 8996
Europe: +44 (0) 181 855 9918 or fax +44 (0) 181 316 2211
Asia: +81 (052) 733 4301 or fax +81 (052) 733 4328
Electronically: Info@zinc.com or GO ZINC on CompuServe.
Web: http://www.zinc.com/ Ftp: ftp://ftp.zinc.com/

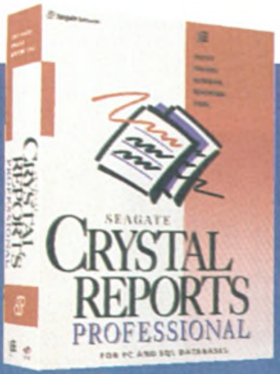
z i n c

NO LIMITS

ENQUIRY NO. 027

Giving You the Reporting Power to Publish to the Web

Call for Our Special Upgrade Price Now



Crystal Reports is the world's most widely used reporting tool with over 2 million copies in use and the preferred reporting engine of more than 135 leading software vendors including Microsoft, Borland, Oracle and Netscape.

Crystal Reports now supports HTML so you can publish to the Web and give others access to your presentation quality reports using any web browser.

Create amazing presentation quality reports such as subreports to form style reports, conditional reports, multiple detail section reports, web-ready reports, multiple summary cross-tab reports, BackOffice® reports and more.

- Low cost upgrade for VisualBasic, Visual C++, Delphi and all major developer tools.
- 16 and 32-bit version with ActiveX, VBX and VCL controls for rapid application development.
- Over 80 customisable graphing styles.
- Over 160 data analysis functions and operators.
- Over 20 file formats for exporting data.
- Data access from any PC, SQL, ODBC-compliant, groupware or Microsoft BackOffice™ datasource.
- Automatically vary reports based on conditional test on data.

ORDER TODAY

CONTEMPORARY
software

Talking Your Language

Tel: 01344 873434

Fax: 01344 872228

ENQUIRY NO. 028

Email: sales@contemporary.co.uk

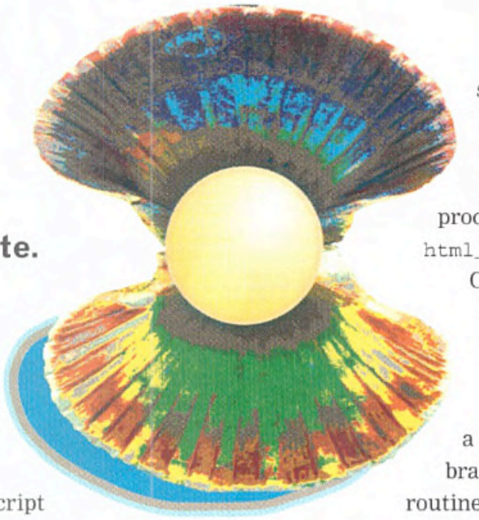
Website: www.contemporary.co.uk

Seagate Software
Information Management Group
Formerly CRYSTAL a Seagate Software Company

© Contemporary Software Ltd. Crystal Reports is a trademark of Seagate Software, Inc. BackOffice and ActiveX are trademarks of Microsoft Corporation.

(Really) enough Perl to get by

Peter Collinson concludes his exploration of Perl 5 with a complete CGI script to handle email notifications on a Web site.



Last month, I started working on a Perl CGI script that could be used to service a Visitors book application for the web. The idea is to take some text from the user, send the text by email to someone, and respond to the Web user with a page that says 'OK, I've done that'. I didn't get too far with the script, but did manage to take the information that the user had typed into the Web page and load it into a Perl associative array in a decoded state so that it could be used in the remainder of the script.

The other aim of last month's article was to explore the basics of Perl and I looked at variables, string constants, lists, associative arrays, two of the fundamental string matching and manipulation primitives, and two of the basic control statements of the language, all this in eleven lines of Perl.

I did lay out some ground rules in last months article. First, I'm not using any of the public libraries largely because I feel that I need to understand the code that I am writing. Second, this article talks about Perl 5 which is the latest release of the language.

Subroutines

There's a great temptation to write Perl CGI scripts as one monolithic chunk where the program enters at the top of the script and steams through the code falling out of the bottom. I have found that a basic knowledge of Perl subroutines can help to make the code somewhat more comprehensible and also you will tend to reuse those routines in later scripts.

For example, any CGI script will want to return different pages to the viewer. At a minimum, you'll want a page saying 'yes that was OK' and one saying 'no, sorry, it failed'. The job of a CGI script is to send some HTML (or some graphics) to its standard output, providing a page for the user to see. It's an obvious saving in code and will also clarify the source if we put some of the program into subroutines and use calls to the routines in several places in the source code.

For a start, let's think about two primitive routines: `html_hdr` will send the start of the HTML page and `html_foot` will send the bottom of the page. By using these routines, we can make the active code just concentrate on the output of the page content.

Subroutines in Perl are defined by:

```
sub html_hdr {
 ...
}
```

The `sub` statement defines a global name of a subroutine. You'll notice that even though I will be giving this subroutine a parameter or two, you don't define local parameter names like C or other procedural languages. You can call the procedure using regular function syntax:

```
html_hdr("Error!");
```

Older versions of Perl insisted that you marked a procedure call by using an ampersand before the name:

```
&html_hdr("Error!");
```

The ampersand is optional in Perl 5. Perl is a little ambivalent about the need for the round brackets around the parameter list. If the `html_hdr` routine is defined before its use, then you can say:

```
html_hdr "Error";
```

If you prefer this style of working, but want to put your subroutines at the end of the script (I do), then you can give the Perl compiler notice that you will be defining a procedure later by including a prototype at the top of your code:

```
sub html_hdr;
 ...
 ... loadsa code
 ...
sub html_hdr {
 ...
}
```

If a procedure has no parameters, then you may want to write

```
noparams;
```

to call it. However, Perl must know that the word `noparams` is a routine. Using a prototype is one way of achieving this, alternatives are to call the routine using an ampersand before the name:

```
&noparams;
```

or by supplying an empty parameter list:

```
noparams();
```

Parameters (and results) are passed as lists. A subroutine can access its list of parameters using the magic list called underscore: `@_`, you may recall that when we talk about lists as a whole we precede the name by the `@` character. So the first parameter to the routine will be `$_[0]`, the `$` character tells Perl that we are expecting a scalar value, the `[0]` is an array index returning the first item of the array. We can use the direct value `$_[0]` in the routine whenever we want to access the parameter, but this isn't exactly user friendly. We'd much prefer to give the parameter a name. One option is to say:

```
sub html_hdr {
  $title = $_[0];
  ...
}
```

assigning the parameter to a variable. An alternative is to use a list assignment:

```
sub html_hdr {
  ($title) = @_;
  ...
}
```

Sorry Sir, – that title is protected with a DESkey

You don't need to travel to find cheap software; international pirates now offer mail order, millions of pounds worth of commercial software is duplicated on 'Gold' CD-ROM's in the UK and Europe. Such is the disregard for software licence agreements by the business community, that emulators for lesser dongles are advertised in the European Computer press.

DESkey security is built into the hardware. ASICs and microprocessors running proprietary algorithms provide real protection. A comprehensive range of drivers and our software protection utility, DESlock, work to bring the highest level of security with the minimum of effort.

Call today for product information, demonstration units and technical advice.

Don't just Dongle it – DESkey it

PC MAC PC Card UNIX etc

Designed for Microsoft Windows 95 · PC CARD · Member of FAST Promoting the legal use of software · QUALITY ASSURED FIRM ISO 9001 SGS · ACS ASIC · Mac OS · Ready for OS/2 WARP

DESkey

ENQUIRY NO. 029

Data Encryption Systems Limited *Silver Street House, Silver Street, Taunton, Somerset TA1 3DL*
Telephone 01823 352357 Fax 01823 352358 BBS 01823 352259 E:mail sales@des.co.uk www.des.co.uk

References & where to get Perl

The book *Programming Perl (2nd Edition)* by Larry Wall, Tom Christiansen and Randal L. Schwartz is published by O'Reilly & Associates and is ISBN 1-56592-149-6. Just in case you didn't know, Larry Wall is the originator of Perl.

Learning Perl is by R.L. Schwartz, again from O'Reilly & Associates. It's ISBN 1-56592-042-2.

The Perl script I am describing is lifted, stolen, and plagiarised from the excellent book *Managing Internet Information Sources* by Cricket Liu, Jerry Peek, Russ Jones, Bryan Buus and Adrian Nye. It's published by, you guessed it, O'Reilly & Associates and is ISBN 1-56592-051-1.

You can get Perl source from the Web: you are looking for the *Comprehensive Perl Network* or CPAN, the main URL for this is *http://www.perl.com*.

If you are a Microsoft user, then you are not left out. Perl is distributed by Microsoft for Windows NT 4.0, it's part of its resource kit.

Finally, if you want to see the code that I've been talking about in action then check out *http://www.hillside.co.uk/visitor.html*. You'll also find a link to the complete source of the CGI script on the *EXE* section of my Web site at *http://www.hillside.co.uk/articles/exe.html*.

Note that we are now dealing with the whole list on the right hand side. By putting brackets around a list of variables on the left-hand side, we are indicating that we want a list assignment, placing a single element from the parameter list into the `$title` variable. Single item lists are somewhat of a special case, list assignment is perhaps more comprehensible if we have two parameters:

```
  ($title, $msg) = @_;
```

We'd call the routine with two parameters, and these would be assigned in one operation to the variables `$title` and `$msg`.

By default, all variables in Perl are global. Ideally, we really want variables in procedures to be local to that procedure, so there is less chance of one procedure interfering with another. We can declare that names are local to the current block by using the keyword `my`, so the `html_hdr` header procedure is better defined as:

```
sub html_hdr {
  my ($title) = @_;
  ...
}
```

which tells Perl that the `$title` variable is local to the procedure block. It behaves exactly like a C local declaration, only statements in this procedure can see the variable.

Generating an HTML header

Having talked about how to define procedures and how to call them, we can now write the complete code to generate the HTML header.

```
sub html_hdr {
  my ($title) = @_;

  print "Content-type: text/html\n\n";
  print "<HTML><HEAD>\n";
  print "<TITLE>$title</TITLE>\n";
  print "</HEAD><BODY>\n";
  print "<h1>$title</h1>\n;
  print "<p>\n";
}
```

The output is a standard HTML page header preceded by a MIME specification giving the browser the type of document that we are sending, don't miss that there's a blank line after the MIME header. I've used Perl's ability to interpolate variable values into strings to print values that contain the title of the page. Perl does have a `printf` routine but this is seldom used, because string concatenation and replacement will do the job for us most of the time.

If you want to output many lines of HTML, it can be tedious to use separate `print` statements. It's quite common to use Perl's *here* documents. Here's the footer routine using a here document:

```
sub html_foot {
  print <<'END_OF_TEXT';
</BODY>
</HTML>
END_OF_TEXT
}
```

The notion of the here document is lifted from the Bourne shell. Here documents are a great way of getting chunks of HTML into a CGI script, and writing that HTML as if it were on a page. The here document allows you to insert inline text that will be printed verbatim from the first line after the << up to (but not including) the end-of-text marker. As you can see, the object after the << character pair is taken as the end-of-text marker. In the example, I've quoted the end marker using single quotes, which tells Perl that the text should not expand any variables that it finds. If I had used double-quotes or omitted any quotes then Perl would scan the text and expand any variable values.

Note the semicolon at the end of the `print` statement. This feels a little strange at first, somehow you feel the statement should be terminated after the end of the here document. Another gotcha is to put a space between the << and the end-of-text string. Perl will take the space as the end-of-text marker, it will treat it as a null value and will terminate the here document on the first blank line. This has somewhat unpredictable results.

Sending mail

Having created the basics for replying to the user, let's look at how to send email. Perl can easily start new programs and send information to them. I'm assuming here that you are running on a Unix system that supports `sendmail`. However, the routine should be easily changeable for other systems and other mail sending programs. The routine will take four parameters: the destination for the mail, the name that the person typed onto the web page, the email information that the person supplied to the web page and the information that they typed.

```
sub mail_send {
  my ($to, $name, $from, $msg) = @_;

  open MSG,"| /usr/lib/sendmail -t";

  print MSG "From: The Web Server <www\@site>\n";
  print MSG "To: $to\n";
  print MSG "Subject: Message from the Web\n";
  if ($from ne "") {
      print MSG "Reply-To: $from\n";
  }
  print MSG "\n";
  print MSG "A person called $name";
  if ($from ne "") {
      print MSG " (email $from) ";
  }
  print MSG "said:\n$msg\n";
  print MSG "--- ends ---\n";
  close MSG;
}
```

Of course, if you plan to use this, you'll need to do some tailoring. I am using the ability of `sendmail` to read all the information that it needs to send a message from its standard input. The `open` statement is used everywhere in Perl to create a file handle that is then used to pass streamed data out of (or into) the Perl program. Here, I am using it to create a pipe to a running command. I am intending to write data down that pipe, this is shown by the vertical bar in the second parameter of the `open` statement. Once I have the program open and running, I can simply throw data down the pipe using `print` statements. Of course, I should check that the open has succeeded.

The contents of the mail message are straightforward. When getting email from the Web, I tend to add a `Reply-To:` field in the mail header so that when the mail ends up in my mail reader I can simply reply to it, should the originator have managed to type in a correct email address. Quite often people don't do this, so beware. It's a good idea to pick up the name of the originating machine from the CGI script environment and pass that in the mail message, this can sometimes tell you what the correct email address should be.

When I have finished creating the mail message, I can close the file handle and `sendmail` will deliver the message. The ability to create crafted data like the mail message and send it out using a variety of other programs is one of the reasons why Perl is such a useful language and tool.

Putting this together

We can now put things together to generate the calling CGI script that provides support for the Visitors book:

```
#!/usr/bin/perl5
$webmaster = 'www@site.uk';
$mailto = 'person@site.uk';

sub html_hdr;
sub html_foot;
sub mail_send;

if ($ENV{'REQUEST_METHOD'} eq 'POST') {

  read(STDIN, $buf, $ENV{'CONTENT_LENGTH'});
  @pairs = split(/&/, $buf);
  foreach $p (@pairs) {
    ($name, $val) = split(/=/, $p);
    $val =~ tr/+/ /;
    $val =~ s/%([a-f0-9][a-f0-9])/
                pack('C', hex($1))/gie;
    $FORM{$name} = $val;
  }

  mail_send($mailto, $FORM{vis_name},
            $FORM{vis_email}, $FORM{vis_comments});

  html_hdr("Thanks");
  print "Your message has been";
  print "sent to $webmaster\n";
  html_foot;

} else {
  html_hdr("Comment Form Error");
  print "Form not processed. ";
  print "Please contact $webmaster.";
  html_foot;
}
```

The script starts by establishing some constants, the local webmaster name and the destination mail address. I've written some scripts that have taken the destination address from a hidden input field on the Web page. I now think that this is unwise. You don't want to create the ability for someone to take a local copy of your HTML page, change the hidden field and then use your CGI script to send mail to some random person.

If you are worried about this kind of misuse, you should include something in your CGI script that verifies that the script has been called from a page that you have control over. My Apache server places the name of the referring page into the script environment using the name `HTTP_REFERER` so you can check the value is what you would expect and bomb out if it's incorrect. Adding an explicit match to the test for POST would do the trick:

```
if ($ENV{'REQUEST_METHOD'} eq 'POST' &&
    $ENV{'HTTP_REFERER'} =~ m,^yoururl.*$,) {
```

The additional test binds (`=~`) a regular expression to be matched against the appropriate environment variable. Usually, we use the slash character to delimit match strings but *yoururl* is likely to contain a slash, so I've used a comma to mark the start and end of the regular expression used to test the environment variable. The regular expression matches the start of the string (`^`), the text of *yoururl*, followed by any character repeated zero or more times (`.*`) up to the end of the string (`$`).

Having defined the constants, I've placed prototypes for the three routines I've developed. You can insert the routines inline here, or place them at the end of the code.

The next chunk of code was developed last month. We check that the CGI script has been called using the `POST` method of forms processing, and if so we read the data that the user has typed in. The data will be several *name=value* pairs separated ampersand characters. First the data is split into a set of pairs and then each pair is processed to create an associative array `FORM` where each key in the name of an input field from the CGI form and its value is the text that the user supplied.

Having decoded the data, we send mail and then create a response page for the user telling them what has happened.

If the page has been called using some other method than POST from a form, then we send an error message. This means that should someone type the URL of the script they will get a sensible reply page. It's also a way of testing that the script compiles and some of the routines work. You can throw the raw script into Perl and you should get some HTML from it.

Well, the object of these two articles has been to give you enough Perl to get by, and I believe that many quite complex web services can be created using the simple tools I've given you. You should worry about the input from users of your web pages and wonder whether that input can be used to subvert your system or your web site. It is usually sufficient to check that the data that the user has supplied makes some sort of sense. ■

Peter Collinson is a freelance consultant specialising in UNIX. He can be reached electronically as pc@hillside.co.uk, by phone on 01227 761824 or on the Web at http://www.hillside.co.uk.

ENQUIRY NO. 030

ENQUIRY NO. 031

ENQUIRY NO. 032

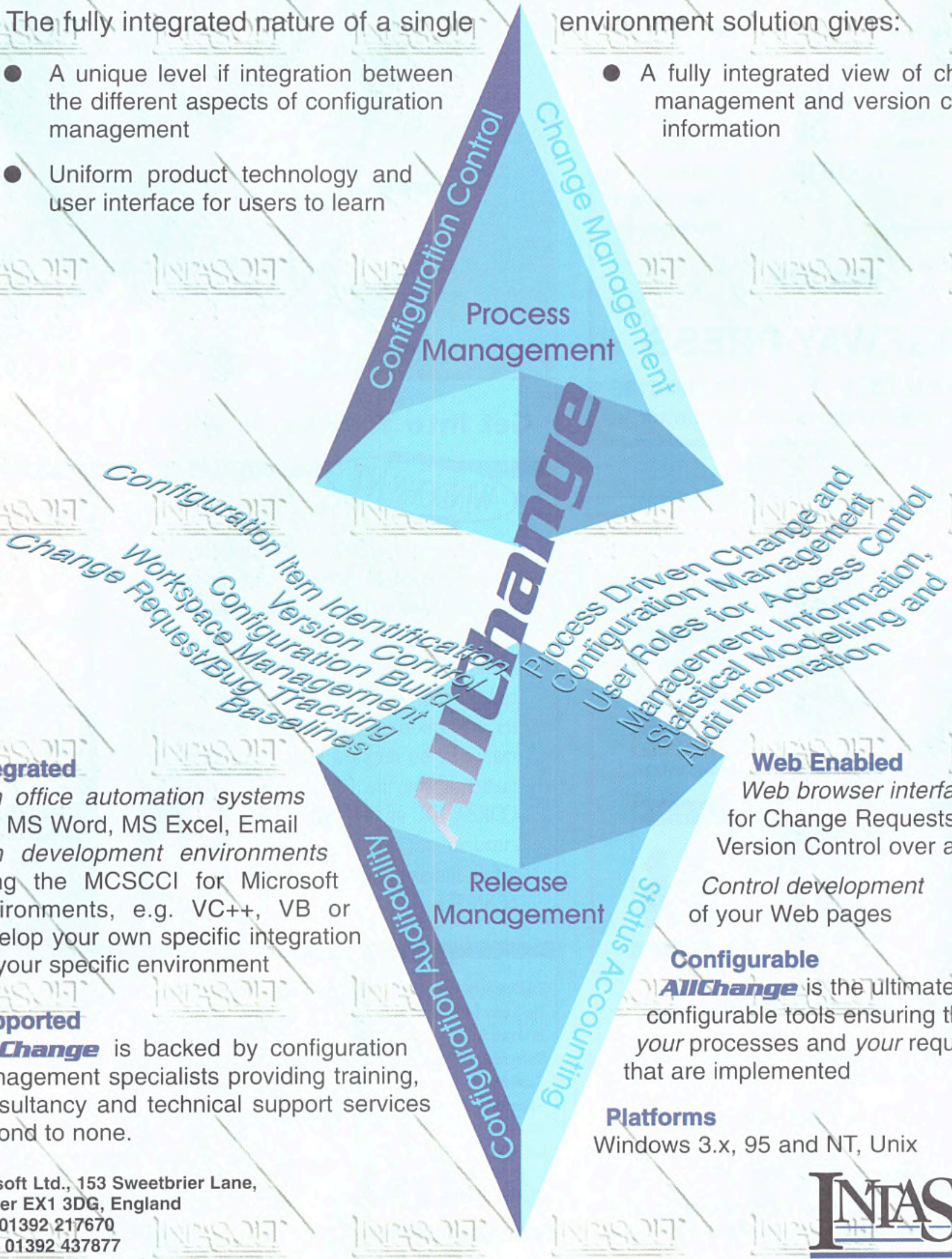
AllChange®

The *Single Integrated CM Solution*

At last you can implement and manage the business processes that you require to support the control of your development and the changes your business imperatives and pressures subject you to in one CM solution – **AllChange**.

The fully integrated nature of a single environment solution gives:

- A unique level if integration between the different aspects of configuration management
- Uniform product technology and user interface for users to learn
- A fully integrated view of change management and version control information



Integrated
with office automation systems e.g. MS Word, MS Excel, Email *with development environments* using the MCSCCI for Microsoft environments, e.g. VC++, VB or develop your own specific integration for your specific environment

Supported
AllChange is backed by configuration management specialists providing training, consultancy and technical support services second to none.

Web Enabled
Web browser interface for Change Requests and Version Control over a net

Control development of your Web pages

Configurable
AllChange is the ultimate in configurable tools ensuring that it is *your* processes and *your* requirements that are implemented

Platforms
Windows 3.x, 95 and NT, Unix

Intasoft Ltd., 153 Sweetbrier Lane,
Exeter EX1 3DG, England
Tel: 01392 217670
Fax: 01392 437877
Email: sales@intasoft.co.uk
Web: http://www.intasoft.co.uk/intasoft/

ENQUIRY NO. O33

INTASOFT

Configuration Management tools for professional software developers

AllChange is a registered trade mark of Intasoft Limited. All trademarks acknowledged.

Mixing constituents

By mixing Basic Constituents you can create new ActiveX controls. Dave Jewell puts on his chef's hat and tries to invent a new recipe.

Photodisc - photodisc.com

Basic Constituents is a recently released collection of ActiveX controls which provide enhanced functionality to the Visual Basic developer. The collection is aimed at Visual Basic 5 users and – according to maker BeCubed – is intended 'specifically for use in ActiveX control creation'. This might seem bizarre until you remember that VB 5 allows you to create a new ActiveX control by aggregating existing controls into a new, enclosing control. Effectively BeCubed is suggesting that you use its controls as building blocks to create controls of your own. If I understand the BeCubed licence agreement correctly, you're entitled to create new controls which you can then market as your own, providing you distribute only the run-time parts – the OCX files – of the BeCubed software to your customers. It's because BeCubed expects you to use its software as *constituent* components of a larger control that the product is called Basic Constituents.

Let's get the bad news out of the way first; unlike many similar products, Basic Constituents ships with printed documentation. Why is this bad news? Well, the printed manual is of mediocre quality with an index which is nothing short of pathetic – less than a page and a half of material for a 482-page book. Things would be pretty desperate if this were all you got, but fortunately Basic Constituents also includes on-line help which is available from the Visual Basic IDE in the normal way. It's essentially the same material as in the printed manual, but since it's in WinHelp form, you're not stuffed by having a virtually non-existent index.

Ingredients

A primary aim of Basic Constituents is to expose the base functionality inherent in many Windows controls, making it available to the application programmer or the control developer. Any API-level developer will tell you that the built-in Windows controls (those contained in `USER.DLL`) and the so-called 'Common Controls' all contain a lot more functionality than is exposed through the properties, methods and events which Visual Basic makes available to you.

To take a simple case, consider the humble Windows push-button. It's not obvious to the end-user, and (unless he's dabbled with the API) it probably isn't even obvious to the average Visual Basic programmer, but at the API level, checkboxes, radio buttons, icons, group

Time to solve the year 2000 problem

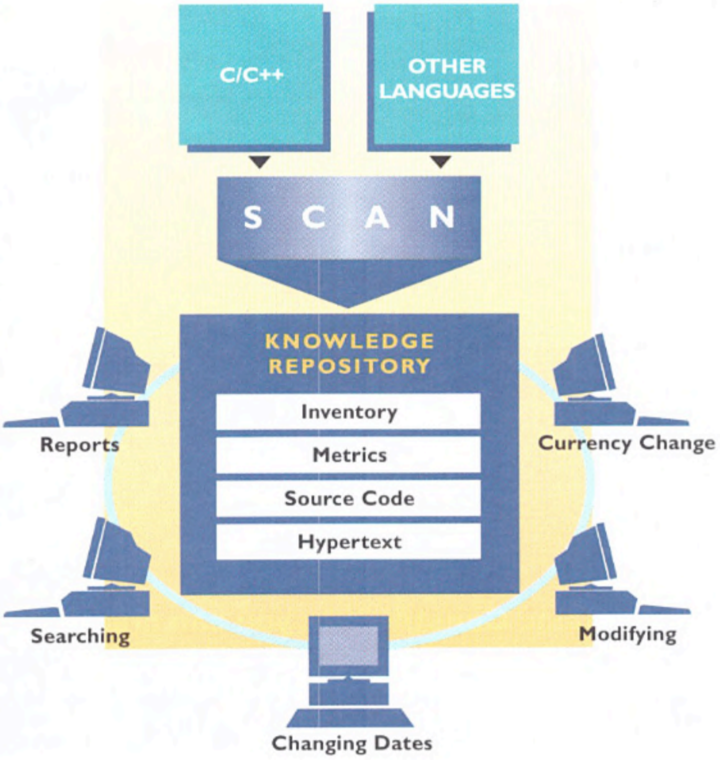
LEGACY SYSTEMS WORKBENCH (LSW) WILL CONVERT YOUR YEAR 2000 CRISIS INTO A MANAGEABLE MAINTENANCE PROJECT.

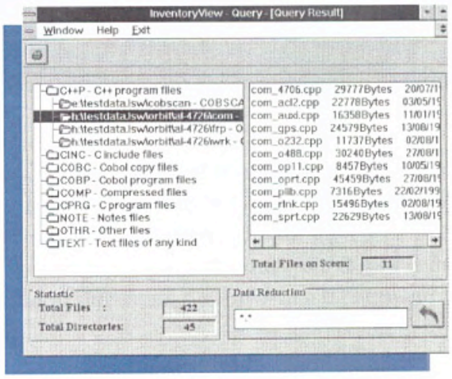
LSW is already widely used in the States by major companies, AT&T, Comsys, Amdahl, the US Ministry of Defence and Government Audit Office. This innovative product is revolutionising and standardising computer software maintenance.

How? LSW reads C/C++ and many other languages, FAST! What might in the past have taken you months can now be achieved in hours.

LSW scans source code and creates a total Knowledge Repository which incorporates a complete System Inventory, Metrics, Source Code and links between code objects, in Hypertext and graphic representations - all in a client/server environment.

Applying this to your Y2K problem will enable you to make the best use of your resources as experienced programmers will be free to instigate date changes and test resultant changes.




InventoryView - Query - [Query Result]

Window Help Exit

```
-C+P - C++ program files                         com_4706.cpp   29777Bytes  20/07/1
  -e:\testdata.lsw\cobscan - COBSCA              com_acl2.cpp   22778Bytes  03/05/19
  -h:\testdata.lsw\orbit\al-4726\com -           com_aud.cpp    16358Bytes  11/01/19
  -h:\testdata.lsw\orbit\al-4726\frp - O         com_gps.cpp    24579Bytes  13/08/19
  -h:\testdata.lsw\orbit\al-4726\wrk - (         com_o232.cpp   11737Bytes  02/08/1
-CINC - C include files                          com_o488.cpp   30240Bytes  27/08/1
-COBC - Cobol copy files                         com_op11.cpp   8457Bytes   10/05/19
-COBP - Cobol program files                      com_oprt.cpp   45459Bytes  27/08/19
-COMP - Compressed files                         com_pilb.cpp   7316Bytes   22/02/199
-CPRG - C program files                          com_rlnk.cpp   15496Bytes  02/08/19
-NOTE - Notes files                              com_sprt.cpp   22629Bytes  13/08/19
-OTHR - Other files
-TEXT - Text files of any kind
```

Total Files on Screen: 11

Statistic
Total Files : 422
Total Directories: 45

Data Reduction
.

Recognising the 'Millennium Bug' as a real problem is your first step. Applying LSW to your code will ensure the next steps are swift, successful and cost-effective.

Each passing day takes you closer to the Year 2000 Problem, LSW takes you closer to the solution.

Evaluate FREE!

Call
+44 (0)171 328 0300

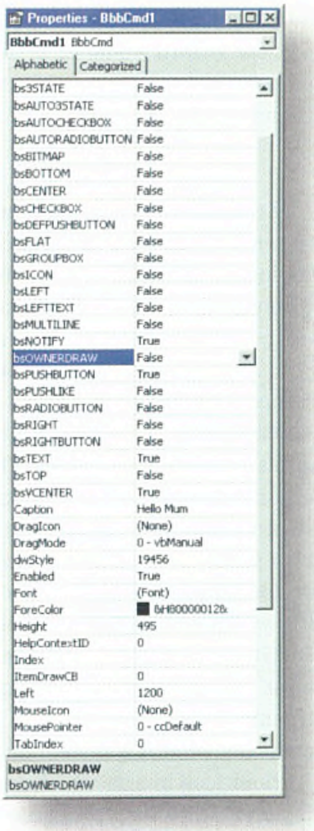
"The trick with maintenance is understanding the system BEFORE you change the code."

QA

Quality Awareness UK Limited

50 Regent Street London W1R 6LP
Tel: +44(0)171 328 0300 Fax: +44(0)171 624 4060
Email: QAUK@QUALITYAWARE.COM

ENQUIRY NO. 034



Properties - BbbCmd1

BbbCmd1 BbbCmd

bs3STATE	False
bsAUTO3STATE	False
bsAUTOCHECKBOX	False
bsAUTORADIOBUTTON	False
bsBITMAP	False
bsBOTTOM	False
bsCENTER	False
bsCHECKBOX	False
bsDEFPUSHBUTTON	False
bsFLAT	False
bsGROUPBOX	False
bsICON	False
bsLEFT	False
bsLEFTTEXT	False
bsMULTILINE	False
bsNOTIFY	True
bsOWNERDRAW	False
bsPUSHBUTTON	True
bsPUSHLIKE	False
bsRADIOBUTTON	False
bsRIGHT	False
bsRIGHTBUTTON	False
bsTEXT	True
bsTOP	False
bsVCENTER	True
Caption	Hello Mum
DragIcon	(None)
DragMode	0 - vbManual
dwStyle	19456
Enabled	True
Font	(Font)
ForeColor	■ &H80000012&
Height	495
HelpContextID	0
Index	
ItemDrawCB	0
Left	1200
MouseIcon	(None)
MousePointer	0 - ccDefault
TabIndex	0

bsOWNERDRAW
bsOWNERDRAW

Figure 1 – The Visual Basic Properties window with a Button Constituent control selected. You can clearly see how BeCubed has aliased a boolean property with each of the possible style bits associated with the BUTTON window class. While flexible, this approach is sure to cause confusion to programmers who aren't familiar with the Windows API.

boxes and push buttons are all implemented through a single window class called `BUTTON`. The different types of control are implemented as distinct styles of the `BUTTON` class. Thus, to get a push button, an API programmer might create a `BUTTON` window using the `BS_PUSHBUTTON` style. To get a radio button, he'd use the `BS_RADIOBUTTON` style, and so on.

The inherent multipurpose nature of the `BUTTON` class becomes much more obvious when we look at the BeCubed implementation of the button control interface. This is provided courtesy of `BBBCMD.OCX`, resulting in a control of type `BbbCmd`. From the partial view of the Properties window (Figure 1), you can see that the component sports a large number of Boolean properties with names such as `bsCheckBox`, `bsPushButton`, and so forth. Many of these properties are mutually exclusive (a `BUTTON` can't be a checkbox and a push button at the same time!) but other style bits can be logically OR'd with other styles to give options such as `WS_OWNERDRAW`. In addition, the control provides a callback hook for setting up a routine to handle owner-draw buttons, and another two hooks which allow you to implement custom message processing (one hook being fired before the call to `DefWindowProc`, and the other after).

Wonderful as all this is, I suspect that it will be somewhat mystifying to the average Visual Basic developer. To get the best out of Basic Constituents, I reckon that you really need a solid grounding in the API. I feel that the decision to expose the full `BUTTON` functionality in the one control was a bad one. It's great to have access to all the available API style bits, together with owner-draw and messaging hooks, but it would have been more sensible to implement one control for radio buttons, another for push buttons and so forth.

This raises another issue – code size (as regular readers will know, code size is one of my favourite bugbears). In the BeCubed manual, it states that using the standard `ListView` component requires the presence of Microsoft's `COMCTL32.OCX`, which weighs in at nearly 580 KB. However, we're confidently told that by using the equivalent BeCubed Constituent, we get far more functionality and only need a relatively small OCX of 50-60 KB. It sounds great, but the comparison is nonsense for at least two reasons. Firstly, the equivalent BeCubed OCX control only provides functionality for the `ListView` control, whereas `COMCTL32.OCX` provides the Visual Basic interface for the *entire* set of Common Controls implemented in `COMCTL32.DLL`. More significantly, the comparison neglects to tell you that the BeCubed OCXs are built using Microsoft Visual C++ and therefore require the presence of `MFC42.DLL`, a 942 KB DLL containing the dynamically linked MFC 4.2 code. Needless to say, this library isn't required by `COMCTL32.OCX`.

So what else does Basic Constituents give you? Aside from the previously mentioned Button Constituent, you get OCX wrappers for the Windows combo box control, edit control, extended combo box, header, image list, listbox, listview, panel, progress bar, rebar (also known as the Coolbar), status bar, toolbar, tooltip and other built-in and common controls. As with the Button constituent, various hooks are provided for exploiting API-level functionality. The combo-box and listbox controls, for example, include hooks for application-supplied callbacks which measure and draw individual list items (corresponding to `WM_MEASUREITEM` and `WM_DRAWITEM` at the API level). There's even a hook for providing your own non-standard comparison routine in sorted listboxes and combo-boxes. Again, this corresponds to the `WM_COMPAREITEM` API message.

Using these various Constituents, it's possible to achieve effects that would otherwise require a certain amount of complex API programming, window sub-classing and so forth. As I pointed out in the July issue, using Visual Basic 5.0's `AddressOf` operator alone can cause all sorts of unpleasantness when it comes to debugging an application – or indeed – whenever a sub-classing application is stopped within the IDE. Much as I dislike the idea of shipping an application with assorted OCX files in tow, there doesn't seem to be any realistic way of avoiding it if you want full access to the API.

Aside from the aforementioned controls, a number of more interesting components provide some higher level functionality. For example, a Task Bar Icon Constituent can be used to add an icon to the system tray area of the Windows 95/NT taskbar. It's easy to use and has relatively few properties and methods; to use it, you just need to set the `Icon` and `Tip` properties to the wanted icon image and tip text (the tip text appears as the mouse is moved over the icon in the tray area) and then call the `AddIcon` method to add the icon to the taskbar. To remove the icon, call `RemoveIcon`, and to modify it, just alter the tip text and icon properties before calling `ChangeIcon` to update the taskbar. Easy!

A pinch of this

Unfortunately, even with a control as simple as this, BeCubed seems to have got things badly wrong: in order to receive events from the taskbar icon, you need to set up an event handler for the control. This handler might, for example, be called BbbTray1_MouseDown, BbbTray1_MouseMove, or whatever, depending on the name of the component and the type of event that you want to receive. For consistency with Visual Basic's standard event types, these handlers take four parameters as shown in the code snippet below:

```
Private Sub BbbTray1_MouseDown(Button As Integer,
        Shift As Integer, x As Single, y As Single)
   Debug.Print "MouseDown" & Button & " " & Shift
        & " " & x & " " & y
End Sub
```

So far so good. However, when you come to use this event handler in anger, you'll quickly discover that irrespective of the mouse button pressed, the Button parameter will always be set to 1! Not only that but the Shift, x, and y parameters are always set to zero! BeCubed seemed to think it was enough to merely signal that the event had taken place, rather than reporting the details of the event. In the present case, it's important to discriminate between left-button and right-button clicks because tray icons typically do different things according to which is pressed (try left and right-clicking the Microsoft volume control applet on the system tray and see the different behaviour that results). Worse, it's important for the application to establish where the mouse click occurred because it will almost always want to display a popup menu in response to a right-click. If we don't know where the tray-click occurred, how do we know where to display the menu? It should be obvious that these considerations render the Task Bar Icon Constituent useless for real-world applications. BeCubed just didn't bother to think things through.

I also encountered numerous problems with many of the supplied sample projects. When installing Basic Constituents, I didn't accept the suggested destination pathname, but instead opted to install into a directory of my own choosing. Silly me. It turned out that many of the samples had project files containing hard-wired pathnames and – not unnaturally – Visual Basic refused to properly load these projects when asked to do so. Surely all this sort of stuff should have been handled by the install program?

Another deficiency I encountered concerned the BeCubed Constituent Control Wizard (let's call it BCCW for short). The BCCW is a Visual Basic add-in that's intended to simplify the creation of ActiveX controls which use BeCubed controls as constituents of the aggregated control. The manual devotes an entire chapter to the BCCW, showing a series of tantalising screen shots with examples of how to add subsidiary components to an ActiveX control, which properties, methods and events of subsidiary components should be exposed by the new control, and so forth. The Wizard includes the useful ability to map properties, methods and events to more than one subsidiary control so that (for example) if the user were to change the background colour or font of the aggregated control, then the property change could transparently be received by all subsidiary components. Once you've set up the various mappings as you want, BCCW will generate all the necessary source code for you.

So where exactly is it? Nowhere. An accompanying readme file confesses that the Wizard 'is not complete as of the shipping date' (7th May 97). I couldn't help wondering why they didn't defer the shipping date until the product was complete. The aforementioned readme offered a ray of hope by stating that the Wizard would be made available on the BeCubed Web site. With eager, trembling fingers I fired up Internet Explorer and found – nothing. According to the information posted on the Web site, the Wizard would be available 'within a few weeks', but at the time of writing it's almost four months since the shipping date and still no wizard. Ho hum.

A pinch of that

Now the good news. There are some genuinely useful components in the Basic Constituents package. Feast your eyes on Figure 2 where you'll see what looks very much like the Windows Explorer. This sample application is made from a single, compound ActiveX control of type xExplorer in conjunction with a very few lines of code. In turn, the xExplorer control is comprised of a number of lower level constituents. Along the top, there's a toolbar control while to the left you can see a Tree View control. On its right, there's a List View control. The aggregate also contains a panel control, a tips control and a number of system image list controls. All the glue which links these controls into a coherent whole is hidden inside the aggregated control but naturally you get the source to all this code which effectively forms part of the sample. (But of course, you don't get the source code to the constituents themselves.)

Maybe you're wondering what is a system image list control? It makes available the various icon images which are used by Windows itself (notably the Explorer) when displaying files of a specific type. This applies not only to files but also to namespace objects such as the printer, the waste paper bin, and so forth. On the subject of namespaces, a shell namespace constituent can be used to give Visual Basic developers access to the Windows shell namespace including Control Panel, Network Neighbourhood, etc.

One potentially interesting control in the Constituents arsenal is the snappily named bbbBrowser component. Using this constituent, it's possible to programmatically read the type library information from a DLL or OCX control where it can be enumerated using installed callback routines. Although this is a pretty specialised com-

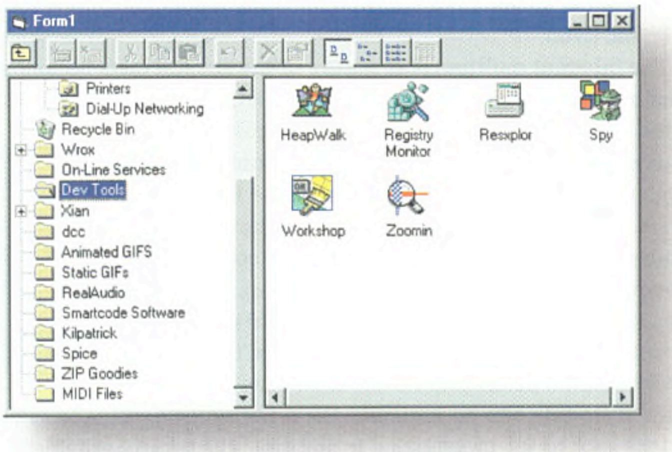
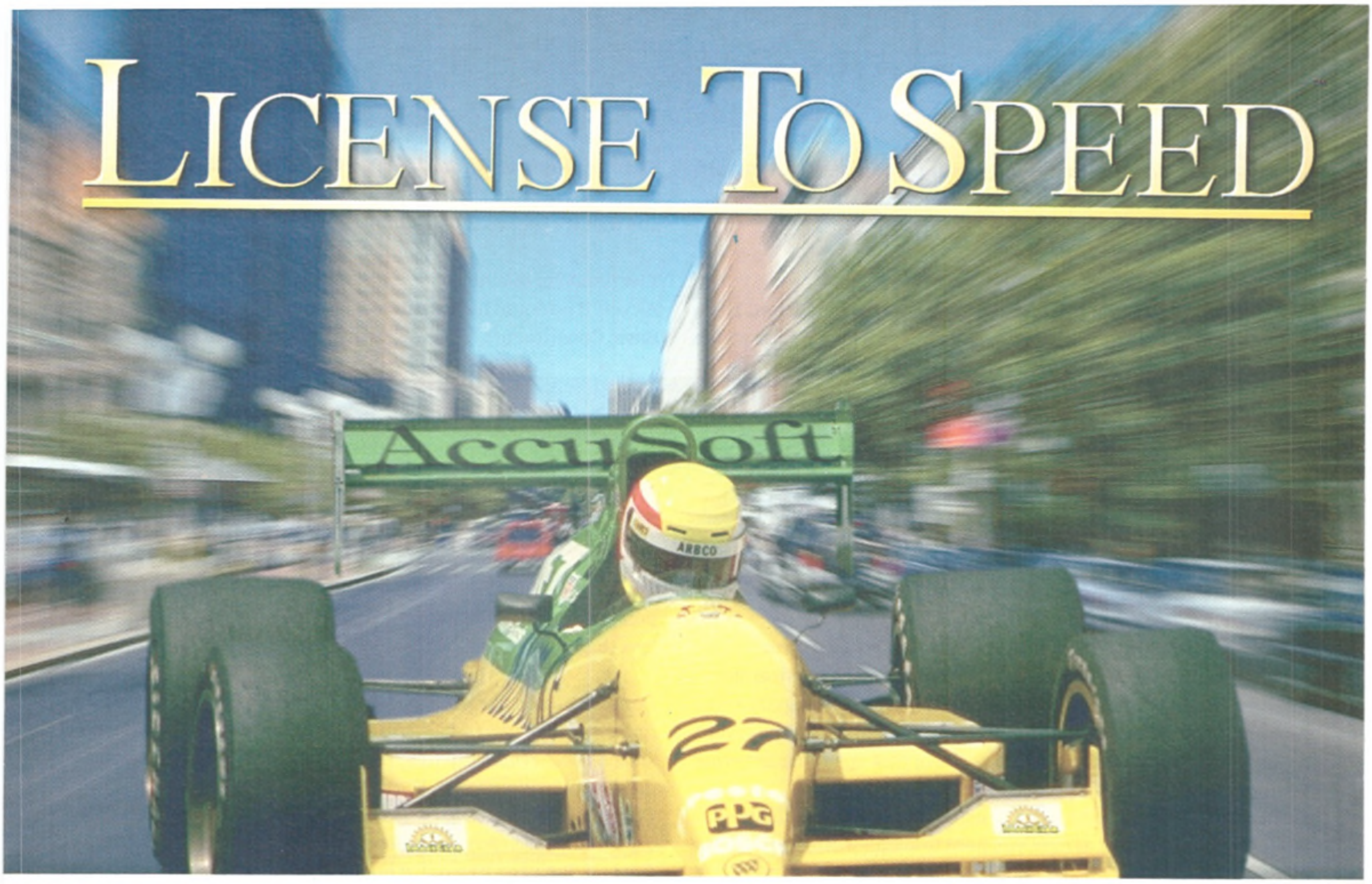


Figure 2 – This Explorer-style demo program illustrates how to put a series of Constituents together to create a single, aggregate control. Here, the entire content area of the window corresponds to a single ActiveX control which has been created from a number of Constituents. See the accompanying text for a description of what controls were used.



ImageGear: High Performance by Design™

Fastest Imaging Toolkit

Give your applications a License to Speed with ImageGear® Pro Gold - the fastest image loading, displaying, image processing, compression and image handling available. No one wants to wait for images to display, and fast is never fast enough, so AccuSoft is committed to continuous optimization of algorithms and code to keep you ahead of your competition.

Over 50 raster formats

ImageGear now supports over 50 formats - more than any other product. With the unique AccuSoft Image Guarantee, it offers the most advanced support for TIFF, JPEG, GIF, MO:DCA, CALS, Photo CD, PNG, AVI and all the rest (see website for complete list). ***Your image is important, so choose the only toolkit that guarantees it!***

Hundreds of features

ImageGear has the most comprehensive set of imaging features available: high speed display, printing, scanning, image processing, format conversion, compression, plus advanced features like scale-to-gray, auto-deskew, noise removal, and all at unmatched speed. You get both high-level and low-level API for maximum ease of use and flexibility plus more of the features you need.

## ImageGear Extensions

ImageGear is designed for easy expansion with optional Extensions that keep you ahead of the game. This allows you access to the latest technologies without waiting for new versions of the toolkit. You also get to choose only those components that you need. We offer industry-leading extensions for: annotations, FlashPix, DICOM, ABIC and other special formats, plus more on the way. The list will keep growing, so check the web site regularly.

Choose the Leader

We make your selection of an imaging toolkit easy by supporting the most formats, and by providing advanced features, faster performance, more options, and the best cross-platform support available. All of this has made AccuSoft the industry leader and your best source for imaging technology.

## ImageGear Extensions

ART 2.0 Annotation Extension

The easiest way to add industry-leading annotations to your applications.

- Full support for the Wang spec for storing annotations in TIFF
- Available as DLL or ActiveX
- Read/write into TIFF or elsewhere
- Over 25 different objects
- Full hyper-linking
- Advanced redaction
- Easy internationalization & security
- High performance display
- Built-in toolbar & dialogs
- Fully automated user interaction
- Fully programmable user interaction

FlashPix Extension

The official and most complete FlashPix support.

- Supports complete set of advanced FlashPix features
- Includes over 30 special API functions
- Supports all FlashPix transformations
- Enables full integration between FlashPix and ImageGear
- Converts FlashPix to/from any other ImageGear format

DICOM Medical Imaging Extension

Advanced support for 8-16 bit grayscale medical imaging.

- Full read & write DICOM 3.0 support
- Easy and full access to all non-image data
- Advanced 16 bit image processing
- Full fidelity of images maintained
- High quality display including real-time Window Leveling
- Conversion of DICOM images to/from any other ImageGear format
- Full access to all ImageGear features
- Transfer syntax auto-detection

- ***No cost Evals available for all Extensions***
- ***All toolkits available as DLL or ActiveX***

ImageGear™
for Document Imaging

- ***Only $2995! Save $2000!***
- ***ImageGear Pro Gold***
- ***ART 2.0** - Annotation Extension*
- ***ImageClean** - Automatic Image Improvement Extension*

AccuSoft®
High Performance Imaging™

Order Today
0181 316 5001

Visit our Website
www.highlander.co.uk
Special Offers, Bundles, Demos and more!

ENQUIRY NO.035

HIGHLANDER
PROFESSIONAL SOFTWARE DISTRIBUTION
SOFTWARE LTD

ponent, it can be quite fun to play with (hint: check out my old favourite, the MSO97.DLL which implements much of the user interface for Microsoft Office 97).

With a cherry on top

The EllTech ZIP component is worthy of special mention. It's a stripped-down version of a product called Compression Plus which is available separately from EllTech and has been bundled into Basic Constituents to provide some added value. The documentation accompanying the ZIP component is the most comprehensive and professional in the manual, extending to almost 40 pages. As implied earlier, the documentation you get relates to the full version of the product, but those parts of the manual which aren't applicable to the stripped-down version have been crossed out.

The stripped-down component can add files to a ZIP archive and delete or extract them, but the ability to freshen or update archives has been removed and it's not possible to perform integrity tests. Automatic creation of stored directory names, handling of password-protected entries and the ability to perform disk spanning are other features which have been disabled in this version.

Despite its limitations, the ZIP component ranks as the best in Basic Constituents. But as ever, you need to keep a weather eye on the subsidiary files required. The ZIP control is implemented via a 68K OCX file which, as ever, requires `MFC42.DLL`. This OCX file is essentially a wrapper which interfaces Visual Basic to the 'real' ZIP engine that's implemented in an EllTech-supplied DLL called `COMPPL32.DLL` (237 KB). I suspect that this is the full version of Compression Plus and that it's the OCX which limits the calls that can be made.

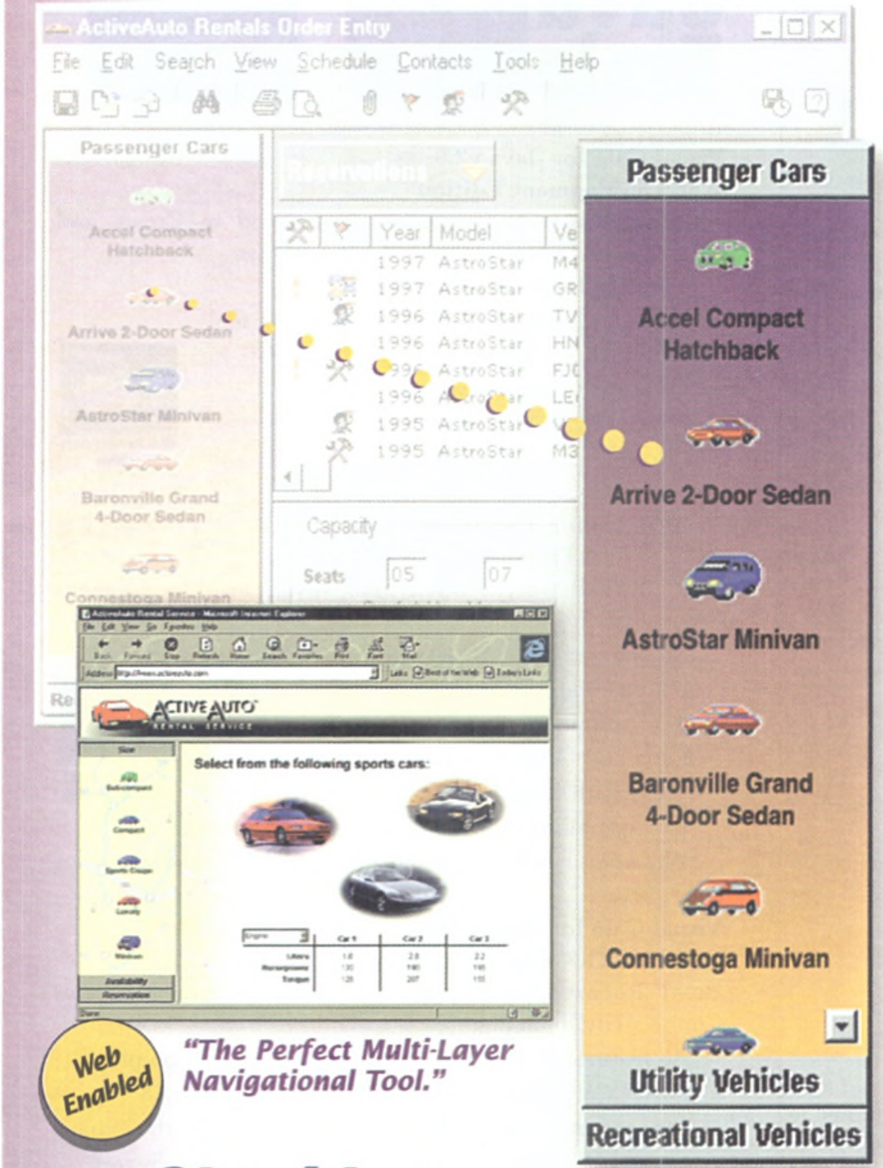
The finishing touch

As you'll no doubt have gathered, I wasn't wildly enthusiastic about Basic Constituents. The overall quality isn't up to the standard that one would expect, the software is clearly still in development, and the whole thing smacks more of a beta than a finished product. I was particularly irritated by the missing add-in Wizard, the poor documentation and the lack of dependency information. OK, it's only version 1.0, but things should really be better than this. My advice would be to wait for the next version. And if you really need the ZIP capabilities provided by the EllTech component, you can purchase the full version of Compression Plus for less than the cost of Basic Constituents. ■

Dave Jewell is a freelance consultant, programmer and technical author specialising in low-level systems programming, development systems and compiler design. He is the author of 'Instant Delphi' published by Wrox Press. You can contact Dave as Dave@HexManiac.com.

To 'try before you buy', download trial versions of Basic Constituents from the BeCubed Web site at http://www.becubed.com. Basic Constituents costs £135 and can be obtained from Contemporary Software on 01344 873434 (email: sales@comtemporary.co.uk).

With ActiveListBar™ it's easy to develop the right outlook for today's Active Office.



"The Perfect Multi-Layer Navigational Tool."

ActiveListBar is an ActiveX™ component that helps give your applications the "look and feel" of Microsoft® Office '97. ActiveListBar helps build programs that users find friendly and familiar.

- **Sliding Groups** - Multiple sliding panels that emulate the look and feel of Office '97's Outlook bar. Groups slide vertically or horizontally.
- **Asynchronous Graphics Download** - When used on a Web page, graphics download asynchronously, allowing the rest of the application to be active before all of the graphics have been downloaded.
- **PNG Compression** - PNG compression files make downloads from the Web noticeably faster.
- **Framework Control** - A lightweight 32-bit control (that doesn't require bulky MFC dll's) minimizes run-time distribution size.
- **Custom Property Pages** - Point and click Custom Property Pages allow users to configure groups, list items, and images in a snap.
- **Customizable Group Backgrounds** - Each panel can have its own picture or colored background. Backgrounds can be tiled, centered or stretched.
- **OLE DragDrop** - Allows quick and easy placement of objects via drag and drop. Users can easily drag OLE objects into the **ActiveListBar**.
- **ActiveBorder Icon Button** - Automatically highlights the border of the icon as the mouse passes over the button, as in the toolbar in Internet Explorer.
- **Sound Effects** - Adds sound to your applications. Sound files are downloaded asynchronously when on a web page.
- **ActiveListBar** is compatible with any fully-compliant ActiveX host environment such as Visual Basic®, Visual C++®, and Internet Explorer™ 3.02 or greater.

To order call: 01344 873434

Sheridan™
The Developer's Edge

CONTEMPORARY
software
Talking your Language

Tel: 01344 873434 Fax: 01344 872228
E-Mail: sales@contemporary.co.uk
The Mews, Kings Ride Court, Kings Ride, Ascot, Berkshire SL5 7JR

Order ActiveListBar™ NOW
£99

ENQUIRY NO. 037

Prices quoted exclude VAT and delivery.

http://www.contemporary.co.uk

www.SILICONRIVER.co.uk

GREENLEAF SOFTWARE

Delivering Programming Solutions For Over 10 years

Greenleaf CommLib v5.2 £259

The industry standard asynchronous comms library for C, C++, Delphi and Visual Basic. This feature-rich product supports file transfers under XModem, YModem, ZModem, Kermit, Compuserve B+ and ASCII. Also includes numerous other features including handshaking, port status and monitoring. Supports all PC platforms and comes with full source.

Greenleaf Comm++ v3.0 £229

Written in C++ and assembly language, provides extensive functionality similar to CommLib, but designed from the ground up as an object oriented alternative. Also supports all PC platforms and comes with full source.

Greenleaf ViewComm v1.1 for WIN 95/NT £289

ViewComm for Windows offers the ability to see what's on the line when you need to work on protocols, comm software development or debug an asynchronous communications link. ViewComm comes complete with additional cables. (DOS version also available.)

These and other Greenleaf products are available from stock.

Video Training for C and C++

Over 5,000 Silicon River video training courses have been sold to companies, universities and individuals wishing to learn or improve their C/C++ skills. Why not join them in getting the most cost effective, high quality C/C++ training available?

C Video Course Personal Edition (6 videos with over 13 hours of training, 365 page workbook and source disk)	£199.95
C++ Foundation Course Personal Edition (6 videos with over 12 hours of training, workbook and source disk)	£199.95
C++ Advanced Module - STL	£49.95
C++ Advanced Module - Exception Handling	£49.95

Call for details of Company and Educational Packs.

SYMANTEC INTERNET TOOLS

ALL NEW v2.0

RELEASED SEPT 15, 1997 WITH

JavaBeans™ JDK 1.1

Visual Café for Java v2.0
Web Development Edition £79

The Solution with Everything You Need to Create and Electrify your Web-Site with Java. Includes:

- @ Visual Café for Java IDE
- @ Visual Page wysiwyg HTML authoring tool
- @ Full JDK 1.1 support
- @ Netscape Communicator
- @ Over 100+ JavaBeans
- @ Introduction to Java Book

Visual Café for Java v2.0
Professional Development Edition £199

The Most Powerful Java and JavaBean Development Environment. Includes:

- @ All features of Web Development Edition plus:
- @ Incremental Debugging
- @ Java source to x86 native compilation
- @ Drag and Drop JavaBeans and Java applets
- @ Work with visual objects or in file view mode
- @ Fast compilers - bytecode or native executables
- @ Wizards, class libraries and JavaBean components

Visual Café for Java v2.0
Database Development Edition £349

Build Robust Java Applications with Instant Database Connectivity. Includes:

- @ All features of Professional Development Edition plus
- @ dbANYWHERE Server - which handles up to 25 connections. Scaleable to virtually any size with simple upgrade. Native drivers for Oracle, Informix, Sybase and MS SQL Server plus ODBC support to 30+ other databases.
- @ Netscape FastTrack Web Server
- @ Sybase SQL Anywhere Relational Database
- @ Object Database and Version Control software

CALL FOR UPGRADE PRICING FROM ANY SYMANTEC CAFE/C++ PRODUCT OR MS J++

CALL FOR SPECIAL EDUCATIONAL PRICING

Silicon River
(0181) 317 7777

Silicon River Video Training Courses, Greenleaf Software Products and Symantec Internet Tools are available direct from Silicon River or from most leading development tools specialists.
Silicon River Ltd, 106-108 Powis Street, London, SE18 6LU. TEL: (0181) 317 7777, FAX: (0181) 316 7778, email: sales@siliconriver.co.uk www.siliconriver.co.uk
All trademarks are recognised. Prices exclude delivery and VAT.

ENQUIRY NO. O38

Java for Windows

It might make Microsoft deeply unpopular with the rest of the Java community, but the days of the Windows-only Java application have arrived in the form of J/Direct. Tom Guinther wonders what makes it tick.

For some time, Microsoft has been working to integrate Java more tightly with the Windows platform. A major part of this effort has materialised in the form of J/Direct, which provides a simple and transparent mechanism for calling Win32 API functions directly from Java code.

In this column we'll cover the basic usage of J/Direct and discuss data type translation, known to COM types as marshalling.

So what is it?

J/Direct is a component of the Microsoft Java SDK 2.0. It allows you, with little or no pain, to call Win32 API functions, such as `Shell_NotifyIcon` or `CreateWindow`, as well as functions in your own custom DLLs. Although the same thing can be accomplished with RNI (Raw Native Interface), J/Direct is designed to work with code that is not specifically Java-aware, while RNI functions must be written to cooperate with the VM.

J/Direct automatically handles cooperation with the VM (much of the guts of J/Direct are implemented in the VM) and most importantly, it automatically marshals Java data types to their C/C++ equivalents. If you are not familiar with the term, think of it as translating from one data type to a different but semantically equivalent one. Marshalling is also used to describe moving a data type from one process or machine to another.

So what does it mean?

The only arguments the developer is required to provide are the function name, the location of a function, the parameters and their types, as well as the return type of the function. In the case of user-defined data types (such as C/C++ structures) the developer must also provide a description of the type that is adequate to make data conversion possible.

In the case of Win32 functions and data types Microsoft has done all the hard work by providing a package with all the common Win32 API functions and types. You can create a simple application that uses Win32 APIs quite easily as the following code shows:

```
public class SimpleTest
{
  public static void main(String args[])
  {
    try {
      MessageBox(0,
        "Calling User32.MessageBox() from Java.",
        "J/Direct Test", 0);
    } catch (Exception e) { System.err.println
        ("Unexpected Exception: " + e); }
  }
  /** @dll.import("USER32") */
  static native int MessageBox(int hwndOwner,
    String text, String title, int style);
}
```

In this case, we are directly declaring a prototype for the `MessageBox` function which resides in `USER32.DLL`. The `MessageBox` function prototype is a standard Java declaration of a native function. What *is* unusual is the documentation comment that precedes the function declaration. It contains a special directive, `@dll.import("USER32")`, that informs the Java compiler that the function should be dynamically linked from `USER32.DLL`. Because this is a 'SimpleTest' the method is declared inline. You probably don't want to do this in every class where you need to call the `MessageBox` API, so use the `User32` class provided in the `com.ms.Win32` package of the Microsoft Java SDK. This class provides the prototypes for the most common `USER32` functions. There are also corresponding classes for `KERNEL32`, `GDI32`, `SHELL32` and a few of the other common system DLLs.

So how does it work?

J/Direct works by using directives embedded in the standard Java Documentation comment `/**`. The compiler parses the directive and adds a special attribute to each J/Direct method or field in the class file. The VM then uses the information encoded in the attribute to determine how to execute the method or marshal data.

The primary directive, `@dll.import`, is used to declare that a native function will use the J/Direct protocol rather than the RNI protocol, and is placed directly above the method declaration. Optionally, the `@dll.import` directive can be placed above the class declaration, which indicates that all native methods should use J/Direct.

The standard syntax for the `@dll.import` directive is:

```
/**@dll.import("LibraryName",<Modifier>)*/
```

where `LibraryName` is the name of the DLL that exports the function. Table 1 lists the range of available modifiers and their meanings.

A second directive, `@dll.struct`, is used to describe a C/C++ structure as a Java class. The `@dll.struct` directive should be placed above the class declaration.

The standard syntax for the `@dll.struct` directive is:

```
/**@dll.struct(<LinkTypeModifier>,<pack=n>)*/
```

where the optional parameter `LinkTypeModifier` specifies the representation of `String` and `char` types and can be `ansi`, `unicode`, or `auto` (in which case the VM will detect the appropriate type at runtime). The optional parameter `pack` specifies the alignment of the structure members and can be either 1, 2, 4 or 8 (the default is 8).

The final directive, `@dll.structmap`, is used to handle declarations of fixed-size array elements within structures. This includes declaring fixed-sized strings, which is necessary because Java treats all `String` data types as having a dynamic size. The `@dll.structmap` directive should precede the string or array declaration in the Java class.

The standard syntax for declaring fixed-size strings using the `@dll.structmap` directive is:

```
/**@dll.structmap([type=TCHAR[size]])*/
```

where `size` indicates the number of characters in the string, including a space for the NULL terminator.

The standard syntax for declaring fixed-size strings using the `@dll.structmap` directive is:

```
/**@dll.structmap([type=FIXEDARRAY,size=n])*/
```

where the parameter `n` indicates the number of elements in the array.

So what's the catch?

The first step in marshalling data is to find an equivalent physical representation. The Java `long` type and the C/C++ `long` type are name equivalent but do not have a compatible physical representation. The Java `long` is 64-bits while the C/C++ `long` is typically 32-bits. Under Win32 operating systems, marshalling a Java `long`, requires the use of the `__int64` data type.

The second step is to ensure that semantic equivalence, or meaning, is maintained. For example, the Java `boolean` type can have one of two values, `true` or `false`. Until recently C/C++ did not have a direct equivalent and code written prior to introduction of the C/C++ `bool` type typically uses an `int` type to represent a boolean value. For legacy code, the Java `boolean` type must be marshalled to a C/C++ `int` type, with the value `false` translating to 0. The two most logical choices for translating the `true` value are 1 or -1. Microsoft chose to translate `true` to 1, which is reasonable but might cause problems with languages that specifically expect or require -1.

The Java `char` type is a 16-bit unsigned character so you might expect a straight physical translation to a C/C++ `unsigned short`. Fortunately, this is not the case, and for good reason. By default, a Java `char` type is translated to an 8-bit `signed char`; the type most commonly used to represent characters in C/C++. Ultimately, the `char` type can be translated according to your specification using the `ansi`, `unicode`, or `auto` directives described previously.

Beyond the simple scalar types, marshalling arrays, the Java `String` type (which is a special form of an array), and C/C++ structures are probably the most essential things to understand. In C/C++, arrays and pointers are equivalent. Thus if you marshal a Java `int[]`, its corresponding C/C++ type will be `int *`.

In the case of the `String` type (or `char[]`) the corresponding marshalled type will be `const char *`. It is `const char *` because a Java `String` type is read-only. This means that you cannot directly pass a Java `String` type to a C/C++ API that intends to modify it. In order to pass a `String` type to a C/C++ API, you must create an instance of the `StringBuffer` class, and set its buffer capacity (`StringBuffer.setCapacity(int)`) to a value which is appropriate for the API you are calling. After the API call you can use the `StringBuffer.toString` method to extract the `String`.

Finally, C/C++ uses many user-defined structures. These are very similar to a Java or C++ class with variations on member access (the default is `public`). Java classes have one property that is not shared by C/C++; the order and offset of the members within the class are not specified. These characteristics of the class are determined by the VM at run-time. When the `@dll.struct` directive is specified the VM treats the Java class as if it was a C++ structure. Whenever an instance of the translated Java class is created, it will be allocated from memory that will not move during garbage collection. The alignment of the structure members will be controlled by the alignment (`pack=`*n*) specified in the `@dll.struct` directive. Also, any `String` type that appears in the Java class will be marshalled according to the `ansi/unicode/auto` specification of the `@dll.struct` directive.

`Ole` – Tells the VM to use special semantics required by OLE APIs

`Ansi, unicode, auto` – Specifies a specific form of a function (`ansi, unicode`) or the most efficient form (`auto`) to be detected at run-time.

`SetLastError` – Indicates the function uses `SetLastError` to set important error information. Use this modifier to prevent the VM from clobbering the API result.

`Entrypoint="DLLFunctionName"` – Used to create an alias for a function name.

`Entrypoint="#ordinal"` – Used to link to an entry-point that is only exported by ordinal (ie it is not named.)

Table 1 – Modifiers for @dll.import.

So where's the example?

To get a feel for how marshalling works we'll take a look at how to define a Java function prototype for the Win32 `RegisterClass` API, and declare a Java class for its `WNDCLASS` data structure. `RegisterClass` API takes as its only parameter a pointer to a `WNDCLASS` structure that describes the window class that is to be registered. We don't have pointers in Java, so we'll pass a `WNDCLASS` reference to the native function. The VM will manage translation of the reference to a pointer as necessary.

The first thing required is to declare a Java equivalent of the C/C++ `WNDCLASS` structure. By declaring a Java class `WNDCLASS` and defining its members, in structure order, with the appropriate marshalled Java to C/C++ data types, and using the `@dll.struct` directive we have essentially created a 'Java struct'.

```
/** @dll.struct(auto) */
public class WNDCLASS {
      public int style;
      public int lpfnWndProc;
      public int cbClsExtra;
      public int cbWndExtra;
      public int hInstance;
      public int hIcon;
      public int hCursor;
      public int hbrBackground;
      public String lpszMenuName;
      public String lpszClassName;
}
```

For the function prototype we use the `@dll.import` directive to specify that the `RegisterClass` API should be linked from `USER32.DLL`. We also want the system to dynamically determine which form of the function (`ansi` or `unicode`) should be used.

```
/** @dll.import("USER32",auto) */
public native static short RegisterClass
   (WNDCLASS lpWndClass);
```

So what's the point?

J/Direct is obviously a powerful way to bridge the gap between new Java technology and the large base of C/C++ legacy code. Next month we'll examine using J/Direct with OLE, and interfacing to custom DLLs. ■

Tom Guinther is a Software Architect for NuMega Technologies in Nashua NH, USA. He can be reached via e-mail at tomg@numega.com. The Microsoft Java SDK 2.0 is in public beta and can be found at http://www.microsoft.com/msdn. The SDK contains the complete documentation for using J/Direct and marshalling data types.

If you do it with C++ – Do it with Sybase Power++!

Real C++ for the C++ Pro:

Raw C++ Speed

Power++ delivers fast, tight code using the industry-leading Watcom C/C++ 11.0 compiler technology. Praised for its outstanding performance and ability to respond to even the most rigorous application demands, Watcom C/C++ technology not only adds speed, it creates a proven, reliable foundation for Power++ and the applications built with it.

Power++ 2.0 adds support for Intel MMX technology and the Pentium Pro processor, to take full advantage of high-performance hardware where available. The chart below provides a comprehensive list of code generator optimizations employed by Power++ to deliver fast, small executables. In addition, precompiled header support, incremental linking, and fast background builds allow Power++ to minimize build times during development. Strong code generator optimizations, including Pentium Pro optimizations and the ability to exploit register-based parameter passing, allow Power++ to generate fast, tight code. You can individually select the optimizations you need to fine-tune your code's performance.

- Loop unrolling
- Constant propagation
- Arithmetic simplification
- Redundant compare elimination
- Machine-specific instructions
- Enregistering of variables
- Jump-chain compression
- Mathematical function optimization
- Instruction scheduling
- Copy propagation
- Common subexpression elimination
- Redundant load/store elimination
- Dead code elimination
- Inline function support
- Tail recursion elimination
- and many more...

Support for New C++ Language Features

Power++ 2.0 now supports run-time type information (RTTI) and namespaces, from the latest ANSI draft. RTTI allows the type of an object to be determined during program execution. A namespace is a declarative region that attaches an additional identifier to any names declared in the object, helping you eliminate conflicts with names declared elsewhere in the program.

Power++ makes C++ approachable

With traditional C++ tools, learning C++ can mean a formidable learning curve. You need to become an expert in Windows application development, the underlying Windows framework class library of the development system, object-oriented programming, and the C++ language. Only after learning all of this can you build applications using C++.

Power++ makes C++ application development approachable because its component-based environment allows you to get underway immediately and learn while you build applications. To get started with Power++, you visually create your GUI, then drag and drop objects into the code editor window to program your event code. A Reference Card presents all the actions you can perform on an object and Parameter Wizards walk you through the coding process with explanations and examples of the parameters involved.

Migrate to Sybase Power++ risk free*

Offers for owners of a Windows C++ Compiler, Migrate to...

● **Power++**	for only £132.00	SYB070
● **Power++ Professional**	for only £332.00	SYB071
● **Power++ Enterprise**	for only £686.00	SYB072

* *60 day money back guarantee*

Prices valid until 31/10/97. Prices do not include VAT, or delivery E&OE

Power++ – makes and uses components

Power++ makes it easy to build custom business solutions by leveraging prebuilt, reusable ActiveX components including ActiveX controls and server components. ActiveX components that you build or purchase from a commercial vendor are automatically integrated so you can immediately tap into their capabilities with drag-and-drop programming, dynamically generated wizards, and online reference information. This not only translates into instant application functionality, but instant expertise for your development team, since you get full component knowledge and assistance at your fingertips. And for maximum performance, Power++ automatically exploits the dual-interface standard of ActiveX components.

The Fastest Way to Build Components

Power++ provides a simple way to create Power++ and ActiveX Server components so you can share your development efforts throughout the team. Power++ components are based on true C++ so they can exploit object orientation, multi-reading support, and inheritance. The product includes component library source code which lets you subclass existing components and add your own properties and events. To enhance productivity, your components are automatically added to the Reference Card so you can tap into the drag-and-drop programming features and wizards.

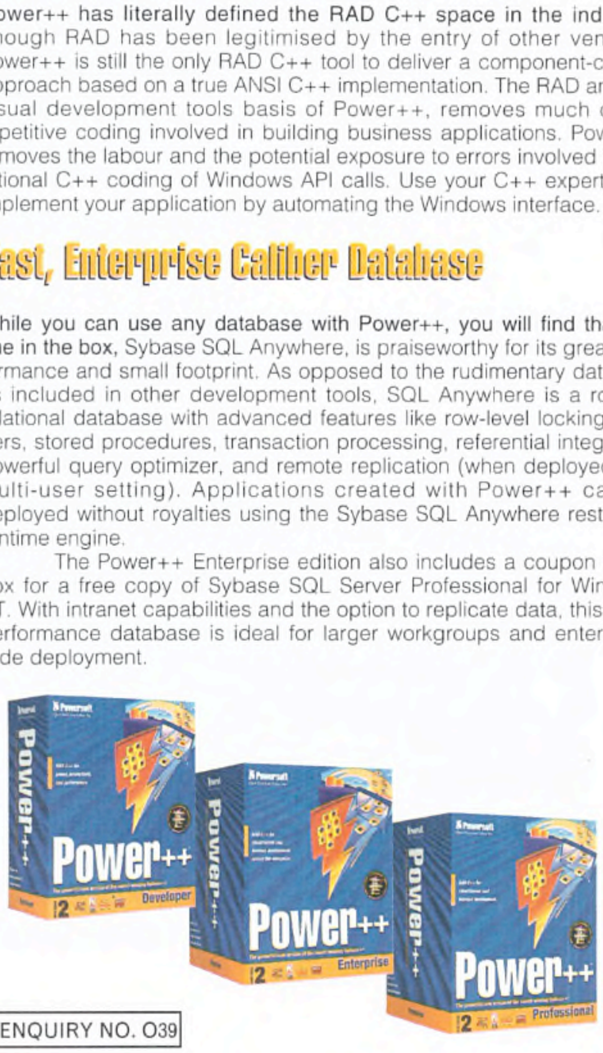
Power++ and Rapid Application Development (RAD)

Power++ has literally defined the RAD C++ space in the industry. Though RAD has been legitimised by the entry of other vendors, Power++ is still the only RAD C++ tool to deliver a component-centric approach based on a true ANSI C++ implementation. The RAD and the visual development tools basis of Power++, removes much of the repetitive coding involved in building business applications. Power++ removes the labour and the potential exposure to errors involved in traditional C++ coding of Windows API calls. Use your C++ expertise to implement your application by automating the Windows interface.

Fast, Enterprise Caliber Database

While you can use any database with Power++, you will find that the one in the box, Sybase SQL Anywhere, is praiseworthy for its great performance and small footprint. As opposed to the rudimentary databases included in other development tools, SQL Anywhere is a robust, relational database with advanced features like row-level locking, triggers, stored procedures, transaction processing, referential integrity, a powerful query optimizer, and remote replication (when deployed in a multi-user setting). Applications created with Power++ can be deployed without royalties using the Sybase SQL Anywhere restricted runtime engine.

The Power++ Enterprise edition also includes a coupon in the box for a free copy of Sybase SQL Server Professional for Windows NT. With intranet capabilities and the option to replicate data, this high-performance database is ideal for larger workgroups and enterprise-wide deployment.



ENQUIRY NO. O39

0171 833 1022

SYSTEM SCIENCE

System Science, 1 Bradley's Close, White Lion St. London N1 9PN

WebDev.SoftDev 97
Olympia 2, London. November 12/13

100% Focused

At last, an event designed exclusively for the UK's leading web & software development teams.

Where else can you:

- Experience the very latest development tools for yourself
- Save time & money
- Gain competitive advantage

Special features at this year's WebDev.SoftDev

- **Java Beans vs. ActiveX** – live debate with Microsoft, Netscape & Lotus, chaired by The Java Forum.
- **Developers' career path** – independent advice about the best skills to acquire and the right moves to make.
- **The DevZone** – hand-on tutorials of the most innovative new products.
- **Visions of the Web** – free keynotes from some of the biggest names in web development:
  Danny Shader, Head of Java Development at Netscape Inc. presents Visual JavaScript for the first time in the UK.
  Sharad Gandhi, Multimedia & Internet Technology Director at Intel presents their for e-commerce over the internetin the new millenium.

Some of the particpating companies include:

Acucobol
AFD
Bits Per Second
Hitachi Software
Informix
LDRA
Liant Software
Lotus
Magic Software
Net Dynamics
Netscape
Novell
Popkin Software & Systems
Powersoft
Rainbow Technologies
Select Software Tools
Sun
Sybase
Symantec
TNG Software
...and many, many more

All this and more – completely free of charge when you visit WebDev.SoftDev 97 at Olympia 2, London, November 12/13 1997. To get further information:

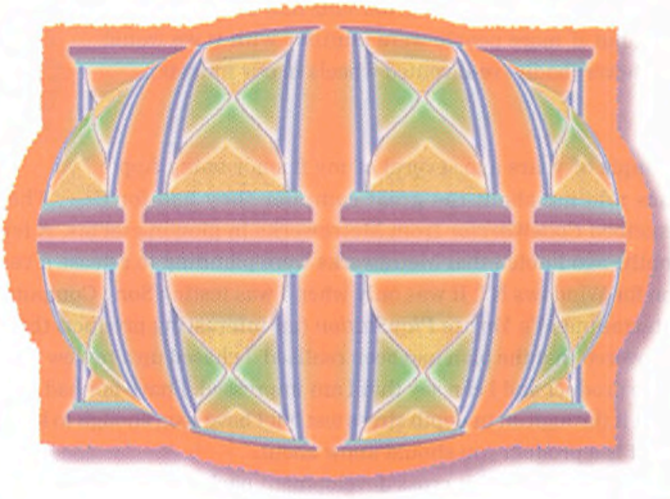
visit our website: http://www.softinfo.com/ev07_ev.htm
or email: softworld@softinfo.com t: 0181 541 5040 f: 0181 974 5188

In association with
COMPUTER WEEKLY

Supported by
EXE
FT
VISUAL BASIC USER GROUP
JAVA FORUM

ENQUIRY NO. O40

Plus ça change



Stroustrup's C++ reference evolves to reflect the major changes in the language while Knuth's legendary The Art of Computer Programming has remained almost unaffected over the years. Here are Francis Glassborow's first impressions.

In my last column I reported very briefly on the publication of *The C++ Programming Language, 3rd edition* (ISBN 0 201 88954 4) and exhorted you to read it. Let me expand a little further.

The three editions of this book represent three key stages in the development of C++. In the first edition (closely modelled on K&R) the C++ is substantially C with a number of interesting extensions. The book clearly addresses C programmers who are looking for support for a higher level of abstraction. This phase of the development of C++ is the one typified by the description 'a better C'. I do not want to get into an argument about that description because whether you find C++ better or worse than C largely depends on what you are using it for.

The second edition represents a stage when C++ was metamorphosing from 'almost C with extras' to a full fledged free standing language. At that stage (1991) C++ was rapidly becoming popular (not, thank goodness, as explosively as Java in 1996) but it was still clearly regarded as a dialect of C. The natural route to C++ proficiency lay through a C style apprenticeship. A major problem with this was that the newcomer was faced with all the complexity of C (doing good work with minimalist tools requires skill and understanding) and then all the added complexity of C++ under development. To make matters worse, many writing about C++ had only half digested the implications of both the actual and proposed extensions. While I am sure that Bjarne Stroustrup had a clear vision of where he was going, much of the middle ground was still only partly formed from primordial chaos.

In the intervening six years a great deal has happened. C++ has matured into an independent entity which should be learnt as a language in its own right. New and powerful idioms have been developed. The experts have refined their view of what is appropriate. We now have an excellent vehicle for a wide range of people from students taking their first tentative steps at programming to distributed teams developing large scale products.

The third edition of *The C++ Programming Language* presents a much cleaner introduction to modern C++. I think that many would be less keen to jump into the cauldron of Java if their first introduction to C++ had been to the language Stroustrup presents here. This book is the product of many years spent listening to the ideas and fears of others by someone who is willing to change. As a single example consider the problem of whether a parameter should be by-value, a pointer type or by-reference. Bjarne Stroustrup is now quite clear and explicit. Small objects can usually be passed by value, larger ones by `const &` (ie the semantics of value but without the overhead of copying). He recommends that you use a pointer rather than a reference for a parameter where the state of the original argument is subject to change by the function. Note that he is writing about declared parameters not the implicit `this` parameter used by a member function, which has always been a pointer.

The second edition of Bertrand Meyer's *Object Oriented Software Construction* (0 13 629155 4) landed on my desk a few weeks earlier. I have always recommended the first edition of this book even to people who had no interest in Eiffel because it was an excellent introduction to OO. Like Stroustrup's book above, *Object Oriented Software Construction* is the product of much thought and a maturing insight into the reader's needs. It's also effectively an entirely new book built on the skeleton of the first edition. You should read this book if you want to understand the subject rather than just use development tools provided by your employer.

Knuth is back

A few days ago the third edition of volume 1 (*Fundamental Algorithms*, 0 201 89683 4) of Donald Knuth's seminal work *The Art of Computer Programming* arrived. I was still doing my Maths degree when Knuth started planning this book in 1962. By the time of first publication, it had grown from a 12-chapter book to a planned 12-chapter work in seven volumes. The second edition of volume 1 was published in 1973 along with the first edition of volume 3. The second edition of volume 2 (*Seminumerical Algorithms*) came out in 1980. Since then, *The Art of Computer Programming* has remained one of the few constants in our world of programming, a half-finished classic.

If you expect an entire rewrite twenty-nine years after the first edition then you have probably not read the original. Few things have changed in the area of fundamental algorithms during that time. 'Proving Fermat's Last Theorem' as an end of chapter exercise has been downgraded in difficulty from 50 (research subject) to 45 (very hard term paper). A slightly cleaner implementation of Euclid's Algorithm appears in the solutions section. The changes are all of this kind, adding polish and updating an already outstanding work. You can't quite use the index of the second edition with the third but you won't be many pages out.

As well as new editions of each of the first three volumes (I expect more change in volume 3 *Sorting and Searching* as a lot of work has been done in this area) we are promised volume 4, *Combinatorial Algorithms*. I am awaiting this with interest, not least because while it only covers two of the original chapters it is being published in three parts. Is this the first time a volume of a multi-volume work is itself a multi-volume publication? And a chapter takes more than a volume?

The Art of Computer Programming may be an academic work, but if more people were familiar with it we would have fewer (badly) re-invented wheels in our industry.

Another compiler

A couple of years ago several of my friends using Apple Macs told me about the great compiler they had for C++. They described *CodeWarrior* from Metrowerks in glowing terms. A few months ago I noted that Metrowerks were planning to release a version for Windows NT. It was only when I was testing Sony Computer Entertainment's *Yaroze PlayStation* (an interesting product) that I discovered that the plan had been realised. I chased up a review copy of the product and I can say that I am impressed. I have not had time to put it through an in-depth stress test but I have seen enough to know that it is a product you should know about.

Briefly, it is a 32-bit cross development package that handles not only C and C++ but also Java and Object Pascal (it can even manage much Delphi source code). In general, the compilers compile to an intermediate language (similar to the excellent JPI TopSpeed compilers) which is then used for chip-specific code generation. In the current version, Java is excluded from this mechanism but I understand that this is going to change some time in the future.

One interesting result of providing compilers for several languages in a single package is that mixed language development becomes standard. Another is that the intermediate language must support the needs of all the high-level languages. This means that array bounds checking is available to C and C++ (you need to do some work, but the compiler can support your efforts). It also implies that future releases may support garbage collection in the intermediate language (required by Java).

There is a special single-platform version for non-professionals (though I believe the licence allows its use for shareware as well as PD development) called *CodeWarrior Discovery Edition* at £59.50. For details and supply of this and the professional versions contact Full Moon Software Distribution Ltd (tel 01628 660242, *http://www.fullmoon.com/*)

Last month's problem

What is wrong with the following code that results in broken code if an exception is thrown from within the try block?

```
void my_string::concat(const my_string & a,
                       const my_string & b){
  len = a.len + b.len;
  try {
    char * temp = new char[len + 1];
    strcpy(temp, a.s);
    strcat(temp, b.s);
    delete [] s, s = temp;
  }
  catch(bad_alloc ba) {
    cout << "Out of memory in string::concat, no
                                    changes made"
         << endl;
    throw ba; // rethrow for user handler
  };
}
```

This is one of those coding problems that you either spot straight away or miss altogether. What makes it worse is that the problem will only surface when an exception has been thrown. That is about the worst scenario for debugging that I can imagine. At the very least test harnesses need to generate fake exceptions, that means that you need a toolkit of special purpose functions. For example, you need a way of getting `new` to throw `bad_alloc` even when there is plenty of memory. You cannot simply over-ride operator `new()` with a version that always throws `bad_alloc()` because that would be triggered too early. I wonder if anyone knows of debugging tools that allow you to force the throw of a specific exception at your chosen point. If so please do not keep it to yourself.

Now have a quick look at my code above and see if you can spot the problem. It is there in the very first line where I change the value of `len`. If an exception does come out of the try block, the `my_string` instance will have the wrong value of `len` for its current array of `char`. So `len` must not be changed until after the call to `new`.

To the best of my knowledge, deleting an array of `char` cannot generate an exception, but in general programmers also need to be very careful of destructors. Any destructor that leaks an exception is very bad news. Consider:

```
Mytype * mt = new Mytype[10];
  // other code
delete [] mt;
```

What happens if one of the calls of `~Mytype()` during the deletion of the array throws an exception? Almost certainly your program is in a mess and needs to use whatever emergency exit is available. This exit should be provided by a call to `unexpected()`. By default, this will call `abort()`. You may not be enthusiastic about this but at least the program will not continue to thrash around doing untold possible damage.

How do we ensure that we get this behaviour? By requiring that all destructors in the code we write have an exception specification of `throw()`. In other words the prototype for `~Mytype` should be:

```
~Mytype() throw();
```

It is time that we started using exception specifications, both to get sane behaviour from application code and to sensitise ourselves to the need to think about what we are going to pass on to others to deal with.

This month's problem

Many of us are aware of the need for using casts when manipulating addresses through `void *` in C++. Unlike C, C++ requires a cast to convert a `void*` to a `T*`. We are also aware that it is good practice to use the new style C++ casts. In that context, can you see any problem with the following code?

```
int comp(void const* vp1, void const* vp2) {
  T const* tp1=reinterpret_cast<T const *>(vp1);
  T const* tp2=reinterpret_cast<T const *>(vp2);
  return tp1->compared_with(*tp2)
}
int main(){
  T array_of_T[10];
  // do something to initialise members of
  // array_of_T
  qsort(array_of_T, 10, sizeof(T), comp)
  // do whatever
  return 0;
}
```

Assume that `compared_with` is a suitable `const` member function of class `T`. ■

Associations of C/C++ Users subscriptions: individual £15, student £7.50, corporate £80, Overload & C++ SIG £30 (including ACCU membership). For further information and application forms write to Francis Glassborow, 64 Southfield Road, Oxford, OX4 1PA, call 01865 246490 or email francis@robinton.demon.co.uk

Microsoft
REQUIRES WINDOWS 95 OR WINDOWS NT WORKSTATION
Development System for Windows and the Internet
The right tools for any job
Visual Basic
Visual C++
Visual J++
Visual FoxPro
Visual InterDev
Microsoft Visual Studio 97 Professional Edition
DEVPARTNER STUDIO
NUMEGA
DEVPARTNER™
The SmartDebugging™ Companion for Microsoft Visual Studio™ 97
STUDIO
BoundsChecker™
SoftICE™
CodeReview™
new! SmartCheck™
FailSafe™
NuMega
NEW!

IF YOU KNOW WHERE YOU WANT TO GO TODAY...

WE CAN HELP YOU GET THERE FASTER.

Announcing DevPartner™ Studio, the SmartDebugging™ productivity companion for Visual Studio.

DevPartner Studio accelerates your team development of multi-language components and applications for the enterprise and the internet. Its full complement of SmartDebugging tools automatically detect, diagnose, and help resolve errors and performance problems. Plus, with our money saving SmartSubscription program, your team's toolbox grows as ours grows. And the introductory price of $869 is a great deal, indeed. Call or visit for the full story.

MONEY-BACK 30 DAYS GUARANTEE

Programmer's Paradise, Inc.

NuMega Technologies
Making Software Work

Programmer's Paradise, UK
+44 0500 284 177
www. programmers.co.uk

Programmer's Paradise, France
+33 01-49.04.11.11
www.iway.fr/ppfrance

Programmer's Paradise, Deutschland
+49 08121/982-130
www.pparadise.de

Programmer's Paradise, Italia
+39 02/9670.0409
www.pparadise.it

NuMega Technologies
+44 (0)118 9253 253
www.numega.com

SmartDebugging™ for Faster Development | Visual Basic® | Visual C++® | Visual J++™

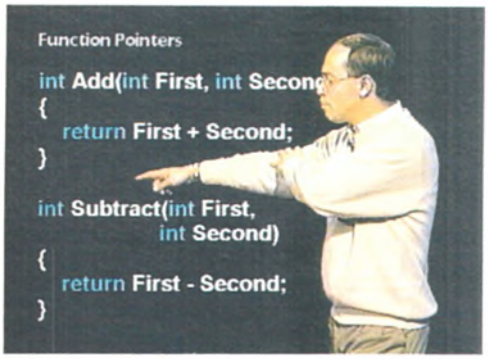
Microsoft Enterprise Development Partner

EXE0997

Numega Technologies, Inc., its logo, Making Software Work, DevPartner, and SmartDebugging are trademarks of Numega Technologies, Inc. All others are the property of their respective owners.

ENQUIRY NO. O41

Silicon River VIDEO TRAINING FOR C & C++



```
Function Pointers

int Add(int First, int Secon
{
   return First + Second;
}

int Subtract(int First,
             int Second)
{
   return First - Second;
}
```

The C Video Course

This comprehensive training course, takes you from beginner to proficient C programmer in the shortest time possible. Designed specifically for Video, the course features advanced display techniques to produce a visually stimulating presentation. By keeping your interest level high, the video ensures that your retention level is also high. The entire C language is covered in 13 hours of video.
Including the time spent working through exercises, you should be able to complete the course in approximately 60 hours. Even though you may not know anything about C when you start the course, you should be a proficient C programmer by the time you finish. You can use the course with any standard C or C++ compiler.
In addition to teaching C, this course is also designed to build the foundation you need for learning C++.

THE C VIDEO COURSE TAPE 1 UNITS 1-6
THE C VIDEO COURSE TAPE 2 UNITS 7-12
THE C VIDEO COURSE TAPE 3 UNITS 13-17
THE C VIDEO COURSE TAPE 4 UNITS 18-23
THE C VIDEO COURSE TAPE 5 UNITS 24-27
THE C VIDEO COURSE TAPE 6 UNITS 28-31

Video Training Benefits

Remarkable value

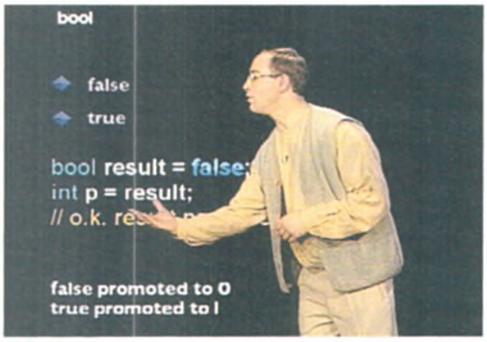
One week's training course	£1295
Personal Edition Video Course	£ 200
You Save	£1095
On Site Training (For Ten)	£6500
Company Edition Video Course	£ 500
You Save	£6000

Convenience
Learn when and where it suits you and at your own pace.

Permanently available
Forget something ? Have a free repeat lesson should you ever need it.

The C++ Foundation Video Course

This comprehensive training course enables C programmers to be proficient in C++ and object oriented techniques in rapid time. Designed specifically for video, the course uses advanced display techniques and powerful illustrations drawn from familiar everyday situations. This approach impresses meaningful pictures on your mind, and stimulates your thinking process, making it easier to grasp C++ concepts first time, and for good !
After watching each of the 12 subject video presentations, completing the corresponding exercises in the accompanying workbook drives home the concepts in the video.



```
bool

  false
  true

bool result = false;
int p = result;
// o.k. re

false promoted to 0
true promoted to l
```

Advanced C++ Video Modules

These modules are ideal for programmers who have completed the C++ Foundation Course, or who already have a solid knowledge of C++.
They cover advanced topics, such as Exception Handling, Templates, STL and Iostreams. These modules examine the syntax of advanced features and spell out the implications for your code. For example one module looks at the benefits of exception handling and the challenges involved in using it to build more robust applications.

Why Silicon River Video Training

"Both products show video can be an excellent medium for training"
PCW magazine
"Recommended" EXE magazine

Over 5,000 video training courses have been sold.
Previous customers include:
Abbey National, Alliance & Leicester, Cable & Wireless, CERN Institute, Glaxo/Wellcome, Logica, Greenwich University, Lucas R&D, Oxford University, Philips Research, Reuters, plus many more company's, colleges and thousands of individuals.

C Video Course

Personal Edition 6 video's with over 13 hours of training, plus approx..50 hours of practical exercises. 365 page workbook and source disc.	£199.95
Five Pack As Personal Edition but with 4 extra workbooks.	£299.95
Company Pack As Personal Edition, but with an extra set of video tapes, 9 extra workbooks, plus exclusive course leaders notes.	£499.95

C++ Foundation Video Course

Personal Edition 5 Video's with over 12 hours of training, plus approx. 45 hours of practical exercises. Extensive workbook and source disc.	£199.95
Five Pack As Personal Edition but with 4 extra workbooks.	£299.95
Company Pack As Personal Edition plus extra set of video tapes, 9 extra workbooks, & exclusive video interviews on applying C++ and OOP.	£499.95

Advanced C++ Video Modules

Personal Edition-Currently Available	
Exception Handling	£ 49.95
Standard Template Library	£ 49.95
Available Summer 96.	
Templates (Personal Edition)	£ 49.95
Iostreams (Personal Edition)	£ 49.95
Combination Set (all 4 modules)	£149.95
Company Packs	£ T.B.A.

ON LINE (Phone/Fax/email) Support/Consulting Packages Available for C and C++. £Call

How To Order

CALL 0181 317-7777

FAX 0181 316-7778

email sales@siliconriver.co.uk

MAIL
SILICON RIVER Ltd.
106-108 POWIS STREET
LONDON
SE18 6LU

VISA

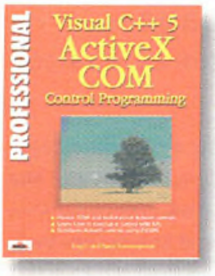
Please add £10 delivery and VAT to all orders

SEE http://www.siliconriver.co.uk

Silicon River Limited, 106-108 Powis St, London SE18 6LU. 0181 317 7777, fax 0181 316 7778.

ENQUIRY NO. 042

Professional Visual C++ 5 ActiveX COM Control Programming reviewed by Gavin Smyth



First of all, if you haven't heard of ATL, have a look at Dave Jewell's *Glad you could COM* in June's EXE: the Active Template Library is Microsoft's latest library for creating slick ActiveX objects. I found that my attempts to use the ATL were frustrated by the inadequacy of introductory material supplied with the library, and that is where this book comes in, as it covers ATL as well as other approaches to writing ActiveX controls in C++.

Overall the book is up to the customary high standards of Wrox's Professional series: there are very few typos and the authors do not waste paper with complete code listings. There is no accompanying disc, but complete source code can be found on the Wrox Web site (*http://www.wrox.co.uk*). As well as control development, deployment and testing of controls in C++, Visual Basic and Internet Explorer is covered extensively. The book is very pro-Microsoft; some of the sections read like marketing blurb. Although the book says that Visual C++5 is required, you can get away with 4.2b plus the ATL download from Microsoft's Web site and still gain a lot from it.

There is a very good introduction to COM concepts, starting sharing binary objects, through to interfaces and object factories. Having taken a theoretical look at ActiveX, the authors develop a simple control from scratch showing that, although the code is not difficult, it is tedious to get right. After this they dive into ATL, rewriting and extending the previous control illustrating how much simpler the task is. Knowledge gained from the earlier section proves to be valuable in demystifying the contents of the myriad of files generated by the ATL wizard. The ATL chapters cover control activation, drawing, properties and property pages, methods and events. Surprisingly the authors omitted scripting and initialisation safety (how to prevent warnings when a control is used within an Internet browser). Fortunately, that is one area covered well in Microsoft's ATL documentation. After ATL derived controls the book moves on to an explanation of bulkier OCXs using sophisticated MFC support.

All of this is put into practice in a larger application – a distributed enterprise-wide event calendar. The controls used range from front-end user interfaces to worker objects, and are written using ATL or MFC, though the authors do not explain why they chose one and not the other for any particular control. My favourite quotation from the book is 'DCOM is just COM with a longer wire'; the ease with which a set of objects working together on one machine can be expanded to the same objects on a number of platforms illustrates the truth of that statement. The book closes with a discussion of control security and licensing, and the deployment of controls via the Internet.

If, like me, you find the Microsoft ATL documentation heavy going, reading this book clarifies much about ActiveX programming.

✓ ***Verdict:*** *Recommended.*

Title:	*Professional Visual C++ 5 ActiveX COM Control Programming*
Author:	*Sing Li & Panos Economopoulos*
Publisher:	*Wrox Press Ltd*
ISBN:	*1-861000-37-5*
Price:	*£36.99*
Pages:	*480*

Framing Software Reuse: Lessons from the Real World reviewed by Mary Hope

Do not be misled by the upbeat title; this is a book in a time warp. The foreword explains how Ed Yourdon and the author mooted this book ten years ago. A long genesis is not in itself a bad thing but this book manages to misrepresent reuse in object-oriented programming, ignore the work of reuse through patterns and the reuse of binary components such as VBX/OCX/ActiveX, and attack yesterday's demons such as the Waterfall method. Bassett proposes a reuse maturity model with no apparent awareness of the various similar theories and models.

The warnings are present right from the start in those awful 'Real World' words in the title. I have always had difficulty with the concept that some worlds are more real than others. Different yes but equal in their existence. A title that better reflects the contents might be 'Reuse with pseudo objects and Cobol' but no publisher would go with that one.

This book could be useful to an unreconstructed Cobol programmer. If the reader has managed to be either ignorant or unenthusiastic about reuse this book has some convincing (but not very original arguments) for its use. It also acknowledges that reuse is not just a technical issue but a cultural and organisational one.

The opening shot is a hard sell about the productivity gains of using the 'frame' approach in software development. When teaching I have often lamented the dearth of hard evidence about the promised benefits of reuse. Superficially, the first few chapters of this book remedy this deficit. However there is a spuriousness about the evidence. It reminded me of pyramid selling meetings designed to inspire where rounds of applause greet the latest staggering success stories. The oft repeated phrase used to describe frames is 'the same as, except'. One waits with bated breath to find out exactly how to use the 'frame approach'.

Well what a disappointment that is. It seems to boil down to writing a Cobol source file that you anticipate might be reusable if parts of it could be changed. This is a frame. You modify the frame, ie reuse it, with a small selection of commands. These commands are implemented by putting the frame through a 'frame processor'. To my way of thinking, this is not much more than a souped up word processor. Possibly useful, but it is not going to replace sliced bread.

Tucked away on page 119 is a small paragraph headed 'language richness' explaining that some languages, such as C++, cannot be used with this method. The main reason being that, in these languages, there are too many ways to do the same thing. Bring back Cobol, all is forgiven.

This book is a useful reminder that we have not yet cracked the reuse problem in software development.

✗ ***Verdict:*** *Only suitable for unreconstructed Cobol Programmers.*

Title:	*Framing Software Reuse: Lessons from the Real World*
Author:	*Paul Bassett*
Publisher:	*Yourdon Press Computing Series*
ISBN:	*0-13-327859-X*
Price:	*£26.95*
Pages:	*364*

SUBSCRIBERS CLUB

Discounted books – exclusively to you

SAVE 20 PER CENT

UML Distilled

Martin Fowler

Publisher's price - £23.95 **Club Price £19.16**

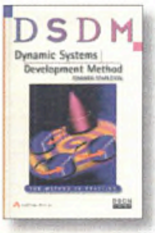
Written for those who are already grounded in object-oriented analysis and design, this concise overview introduces you to UML, highlighting key elements of its notation, semantics and processes. Included is a brief history of UML's history, development, and rationale, as well as discussions on how UML can be integrated into the object-oriented development process. The book profiles various modelling techniques associated with UML along with concise descriptions of notation and semantics and numerous insightful tips for effective use.

SAVE 20 PER CENT

DSDM – Dynamic Systems Development Method

Jennifer Stapleton

Publisher's price - £27.95 **Club price £22.36**



The Dynamic Systems Development Method provides a framework of controls and best practice for rapid application development. It explores the day-to-day realities of implementing the Method. It is a practitioner's guide, dealing with issues such as getting people from different disciplines to work as a team. It has proved extremely effective in delivering maintainable systems which match the needs of the business better than those produced using traditional life cycles.

SAVE 20 PER CENT

The C++ Programming Language – Third Edition

Bjarne Stroustrup

Publisher's price - £27.95 **Club price £22.36**

This is a complete rewrite of the most widely read and most trusted bookon C++. Based on the ANSI/ISO C++ final draft, this book covers the C++ language,its standard library, and key design techniques as an integrated whole.

The C++ Programming Language provides comprehensive coverage of C++language features and standard library components. For example:

- abstract classes as interfaces
- class hierarchies for object-oriented programming
- templates as the basis for type-safe generic software
- exceptions for regular error handling
- namespaces for modularity in large-scale software
- run-time type identification for loosely coupled systems
- the C subset of C++ for C compatibility and system-level work
- standard containers and algorithms
- standard strings, I/O streams, and numerics

With this third edition, Stroustrup makes C++ even more accessible to those new to the language while adding information and techniques that even expert C++ programmers will find invaluable.

ADDISON-WESLEY *Information supplied by the publishers.*

Selection	RRP	Club Price	
UML Distilled	£23.95	£19.16	Oct 97
DSDM – Dynamic Systems Development Method	£27.95	£22.36	Oct 97
The C++ Programming Language – Third Edition	£27.95	£22.36	Oct 97
Details of all books below can be found in EXE Magazine, Dec 96 - Sept 97			
Practical Object Oriented Develop. in C++ & Java	£27.50	£22.00	Sep 97
Client/Server Programming with Java & CORBA	£29.95	£23.97	Sep 97
Java Programming with CORBA	£24.95	£19.97	Sep 97
STL for C++ Programmers	£27.50	£22.00	Sep 97
Borland C++: The Complete Reference	£29.95	£23.95	July 97
Developing Software Applications	£33.95	£27.15	July 97
The Visual InterDev Handbook	£19.99	£15.99	July 97
The Year 2000 Problem Solver	£22.95	£18.35	July 97
Concurrent Programming in Java	£29.95	£23.95	June 97
C++ Distilled	£13.95	£11.15	June 97
Delphi Component Design	£30.95	£24.75	June 97
Managing Technical People	£17.95	£14.35	June 97
C++ Interactive Course	£46.95	£37.60	May 97
Teach Yourself C++ Programming	£23.00	£18.40	May 97
C++ - How to Program	£24.95	£19.95	May 97
Windows NT 4.0 Server Unleashed	£54.95	£43.95	May 97
Requirements Engineering	£24.99	£19.99	Apr 97
TCP/IP Unleashed	£49.95	£39.95	Mar 97
Netware Unleashed	£35.50	£28.40	Mar 97
ActiveX from the Ground Up	£22.95	£18.35	Feb 97
Visual J++ Handbook	£22.95	£18.35	Feb 97
The Cyberia Guide to Smart Web Publishing	£49.99	£39.99	Feb 97
The Java Language Specification	£30.95	£24.75	Jan 97
The Java Tutorial	£26.60	£21.25	Jan 97
The Java Class Libraries	£36.95	£29.55	Jan 97
Client/Server Unleashed	£41.50	£33.20	Dec 96
JavaScript Essentials	£26.95	£21.55	Dec 96
The Java Virtual Machine Specification	£30.95	£24.75	Dec 96
The Essential Client/Server Survival Guide	£24.95	£19.95	Dec 96

BOOK ORDERS

Title **QTY** **PRICE**

Shipping at £3.50 per order **TOTAL** ______

My Subscribers Club Number: ______

Payment Options Cheques or purchase orders only

❐ I enclose a cheque for ______
(payable to *EXE Magazine* and drawn on a UK bank)

❐ I enclose a company purchase order. Please send an invoice.

Simply fax to 0171 437 1350 (with purchase order) or post (with cheque) this form with your order to: ***EXE Book Page*, Centaur Communications Limited, Freepost 39 (WD 1414/29), St Giles House, 50 Poland Street, London W1E 6JZ**
Please allow 28 days for delivery.

Name ______

Address ______

Postcode ______

EXE DIRECTORY

Call Mark Parker on 0171 287 5000 for details

TRAINING

Computer Training & Education
34-36 Rose Street
North Lane
Edinburgh EH2 2PL
Tel. 01345 697611

Informix Software
Informix House
Littleton Road
Ashford
Middlesex TW15 1TZ
Tel. 0181 818 1010

Kingston University
Faculty of Technology
Penrhyn Road
Kingston Upon Thames
Surrey KT1 2EE
Contact Linda David
Tel. 0181 547 7054
Fax 0181 547 7971
Kingston University is a charity dedicated to the development of individuals and organisations through education.

Learning Tree
International Ltd
Mole Business Park
Leatherhead
Surrey KT22 7AD
Contact Jan Mott
Tel. 01372 364600
Fax 01372 364611
uksales@learningtree.com
WWW .learningtree.com

Merlin Training and Development Ltd
The Derwent Business Centre
Clarke Street
Derby DE1 2BU
Tel. 01332 201911
Fax 01332 201912
Specialist tailor made courses for groups of people on your premises, or public courses in Derby.

Network Consultants
7 West Bar
Banbury Oxon OX16 9SD
Tel. 01295 253689
Fax 01295 271218

Peritas Ltd
Burfield Road
Old Windsor
Berks SL4 2JP
Contact: Mari Shepherd
Tel. 01753 604138
Fax 01753 604160
http://www.peritas.com
Peritas is the largest UK IT Training Company. A Microsoft Authorised Technical Education Centre supporting the MCP Programme including Microsoft Certified Solution Developer Certification.

QA Training Ltd
Cecily Hill Castle
Cirencester
Gloucestershire GL7 2EF
Contact Amanda Whitehead
Tel. 01285 655888
Fax 01285 644828
www.qatraining.com
Leaders in IT technical training and consultancy.

Rhino Training
PO Box 1087
Bristol
BS12 2XY
Tel. 01454 417057
Fax 01454 417067
100525.2054@compuserve.com
WWW .rhino-software.co.uk
Providers of training in; Visual Basic, Visual C++, Windows NT, SQL Server, Delphi, Java, Internet/Intranet

Richfords Computer Services
South Bank Technopark
90 London Road
London SE1 6LN
Tel. 0171 922 8819
Fax 0171 922 8839

Skilladvance Training
707 High Road
Finchley
London N12 0BT
Contact Bill Cosgrave
Tel. 0181 446 6481
Fax 0181 446 9143
training@skilladvance.co.uk
WWW.skilladvance.com
High quality Training in Unix, Informix, Oracle, Windows NT, Access, Visual Basic, Internet and Networks.

SECURITY PRODUCTS

Aladdin Knowledge Systems UK Ltd
Tel. 01753 622266
Fax 01753 622262
sales@aldn.co.uk
http://www.aks.com
Aladdin is a leading supplier of advanced software security (HASP) and smart card development tools (ASE) for software developers.

BL Computer Security Ltd
101 Hendon Lane
Finchley
London N3 3SH
Tel. 0181 343 0734
Fax 0181 346 2672
bl@blcs.co.uk
www.blcs.co.uk
We specialise in design and manufacture of computer security products. Anchol, Lure Booster, Deadlock(Dongles) and C.L.A.M.P Alarms

Data Encryption
Systems Ltd
Silver Street House
Silver Street
Taunton
Somerset
TA1 3DL
Contact Roy Davidson(Sales)
Tel. 01823 352357
Fax 01823 352358
www.des.co.uk
deskey@silver.cityscape.co.uk
DES manufactures software security products developed as a solution to software piracy and theft

Glyn Williams & Associates
Ladywood House
Ladywood
Near Droitwich Spa
Worcestershire WR9 0AJ
Tel. 01905 757700
Fax 01905 757800
gwa@gwassoc.demon.co.uk
Software/Hardware copy protection systems - The professional choice for security and features. Worldwide support.

Rainbow Technologies Ltd
4 The Forum
Hanworth Lane
Chertsey
Surrey KT16 9JX
Tel. 01932 579200
Fax 01932 570743
sales@uk.rnbo.com
Only Rainbow delivers leading edge technology and ISO certified quality for software protection and license management.

Softlok International Ltd
Softlok House
2 Bark Street East
Bolton
Lancashire BL1 2BQ
Tel. 01204 436000
Fax 01204 436025
sales@softlok.com
Protect your future, protect your software. Established in 1987, Softlok specialises in software piracy protection.

DEVELOPMENT TOOLS

Bits Per Second Ltd
14 Regent Hill
Brighton BN1 3ED
Tel. 01273 727119
Fax 01273 731925
rflowers@bitspersecond.co.uk
http://www.bitspersecond.co.uk
Graphing/GIS tools. Centura support of training. Client/Server design & consultancy. Network storage consultancy. Multi-platform device driver development.

Borland International Ltd
8 Pavilions
Ruscombe Business Park
Twyford
Berkshire RG10 9NN
Tel. 01734 320022

Citadel Software Ltd
Coombe
Trewen
Launceston
Cornwall PL15 8QF
Tel. 01566-86037
Fax 01566-86147

ComponentSource
27-37 Vachel Road
Reading
Berks RG1 1NY
Tel. +44 (0)118 958 1111
Fax +44 (0)118 958 9999
Request FREE CD or product
sales 101320.2624@-compuserve.com
www.componentsource.co.uk
TRY freely available demonstations of hundreds of software components and BUY and unlock full versions instantly, from a FREE regular CD.

Expos
Acorn House
Straight Bit
Flackwell Heath HP10 9LS
Tel. 01628 533143

Highlander Software
112 Powis Street
London SE18 6LU
Contact Justin Robinson
Tel. 0181 316 5001
Fax 0181 316 6001
sales@highlander.co.uk
Suppliers of high quality development tools for C, C++, Visual Basic, Delphi and Java.

IBM United Kingdom
Freepost ACG 5022
Wintermill
Milton Keynes
Tel. 0800 969045

Intasoft Ltd
Tresco House
153 Sweetbrier Lane
Exeter EX1 3DG
Tel. 01392 217670
Fax 01392 437877
sales@intasoft.co.uk

MKS UK Ltd
243-253 Lower Mortlake Road
Richmond, Surrey TW9 2LL
Tel. 0181 948 5166
Fax 0181 948 8948

Popkin Software & Systems
St Albans House
Portland St
Leamington
Warwickshire CV32 5EZ
Tel. 01926 450858
Fax 01926 422165
uk@mks.com

Powersoft Europe Ltd
Windsor Court
Kingsmead Business Park
High Wycombe
Bucks HP11 1JU
Tel. 01628 34500
Fax 01628 38660

Programming Research Ltd
Glenbrook House
1/11 Molesey Road
Hersham,
Surrey KT12 4RH
Tel. 01932 888080
Fax 01932 888081
Contact Mr John Heathcote

Pure Atria
Wyvlos Court
Swallowfield Nr Reading
Berks RG7 1PY
Tel. 01344 462500

QBS Software Ltd
11 Barley Mow Passage
Chiswick London W4 4PH
Tel. 0181 956 8000
Fax 0181 956 8010
orders@qbss.co.uk
www.qbss.com
Vast range of development products: Next Day delivery; 90 days free support; account customers welcome.

Quadron
209 East Victoria Street
Santa Barbara
CA 93101 USA
Tel. +1 805 966 7630

Quite Software
105 Ridley Road
Forestgate
London E7 OLX
Tel. 0181 522 1726
Fax 0181 2571044

Select Software Tools Ltd
West Moreland House
82-86 Bath Road
Chelenham GL53 7JT
Tel. 01242 229700
Fax 01242 229701
http://www.selectst.com
Select Software Tools provide OO modelling tools for large scale Client/Server development.

Silicon River
58-60 Beresford Street
London SE18 6BG
Tel. 0181 316 7777
Fax 0181 316 4138

System Science
1-6 Bradley's Close
White Lion Street
London N1 9PN
Tel. 0171 833 1022
Fax 0171 837 6411

Zinc Software UK Ltd
106-108 Powis Street
London SE18 6LU
Tel. 0181 855 9918
Fax 0181 316 7778

PUBLISHING

AP Professional
24/28 Oval Road
London NW1 7DX
Contact: Rachel Bridgman
Tel. 0171 482 2893
Fax. 0171 267 0362
app@apuk.co.uk
www.europe.apnet.com/approfessional
Book publisher on internet development,programming, PDAs, software agents and more. Specialists in Macintosh books.

IDG
61/63 Uxbridge Road
Ealing
London W5 5SA
Tel. 0181 579 2652

HELP SYSTEMS

Peterborough Technical Communication
8 Whitewater
Peterborough PE2 6FB
Tel. 01733 237037
Lo-*call* 0345 419470
Fax 01733 239933
petecom@bcs.org.uk
www.gold.net/petecom/
Creation of manuals and on-screen help systems. Please call or e-mail for our free information pack.

Please arrange______yr(s) entry into the EXE Directory for me/my company.

❑ YES I want a one year entry at £240 (+ VAT)   ❑ YES I want a two year entry at £400 (+ VAT) + an extra 5 words free*

Name: .... Job Title: .... Company Name ....

Address: .... Postcode ....

Tel: .... Signature: .... Date: ....

Directory Heading: .... Fax No: .... Contact Name: ....

Contact Tel: .... Extra words: ....

❑ I enclose a cheque for £_______made payable to Centaur Communications. (Please write 'EXE' on the back of all cheques)

❑ Please invoice my company

For extra words on a one year booking the cost is £10 per word

Borland International, Inc is recognised world-wide as a leading provider of solutions for software developers and has a superb range of Desktop, Client/Server, Internet/Intranet and now Enterprise software tools. Borland's mission is to provide superior technology and services that dramatically improve the software development process. Borland now has the following opportunities based at their UK Headquarters in Twyford.

PRODUCT MARKETING MANAGER*

The Product Marketing Manager is the key evangelist in the UK Marketing team, working with customers and partners to communicate the business benefits of Borland's development tools and the Infonet strategy. First-rate presentation skills are essential, as is a good understanding of development tools and the technology product life cycle.

EUROPEAN SUPPORT ENGINEER

Borland's European Developer Services group provides support services for European customers and partners. The UK team requires a European Support Engineer to cover the growing need from UK and European customers for high level support. Applicants must have hands-on experience and education in programming in C++ and/or Delphi. A minimum of two years in a support or programming function, as well as experience with PC and Unix database products.

ENTERPRISE SYSTEMS CONSULTANT*

Part of the pan-European team dealing with major Corporate customers and working on large scale distributed computing projects, responsibilities include project scoping and design, troubleshooting and product support. Experience with UNIX and C or C++ is essential , experience with Windows NT and visual development languages is useful. A premium will be paid for candidates with proven skills in distributed systems technologies such as OSF/DCE, OMG/CORBA and Microsoft DCOM. A challenging and rewarding opportunity that requires a high level of self-management and motivation.

Borland offers competitive salaries, plus bonus, pension and other large company benefits (*includes company car / allowance).

**Please send or fax your CV to Beverley Mortimer at
Borland International (UK) Ltd 8 The Pavilions,
Ruscombe Business Park, Twyford, Berkshire, RG10 9NN
(no agencies please) or Email to the address below.**

Borland

e-Mail: bmortime@corp.borland.com Internet: http://www.borland.com

the soft corporation

Specialists in Software Development Staff Recruitment

OOD/OOP, C, C++, VISUAL C++ — ALL LEVELS

As the market for Object Oriented skills gathers pace we have a number of clients designing systems in diverse application areas including: **Multi-media, Virtual Reality, natural language, DTP, Telephony, LANs, Electronic publishing, On-line information Feeds, Finance and Banking** in both Windows and NT Server.

Position available vary from traditional Programmer/Software Engineer and Analyst/Programmers to Designer/Senior Software Engineers in the overall strategic direction for end-user organisations.

£17-£35K + benefits **REF: SC/01/EXE**

WINDOWS OR X-WINDOWS/BANKING — ALL LEVELS

Three city clients require windows skills at any level. Other relevant skills are SQL server, Transact, SQL, UNIX, VMS or PS-DOS, C, C++, Open Client (DB and Net library), MFC, Open interface and APT. Exposure to analysis, developing user interfaces and rapid development techniques. Full training in Middle Office/Production and Front Office Systems including: Financial and Management Accounting, Treasury, Equity, Fixed Income and Derivatives.

C AND C++ PROGRAMMERS — ANALYST PROGRAMMERS

Excellent opportunities exist for bright graduates with one year + experience. Personal background requires a solid understanding of the project life cycle and a commitment to high quality coding. You will be trained in all aspects of Investment Banking, relational databases, 4GLs and Object Orientated Design. A good opportunity for a second career move.

£17-£25K + Banking benefits **REF: SC/03/EXE**

CAMBRIDGE - MANY, MANY EXCITING OPPORTUNITIES

A wide variety of specialist, leading edge IT companies in areas as diverse as: **ROBOTICS, TELECOMMUNICATIONS, MULTI-MEDIA, GIS, BUSINESS MODELLING, FINANCIAL/TREASURY, EMBEDDED SYSTEMS AND SOFTWARE RESEARCH/ MANUFACTURING** require high calibre software development staff at junior and senior levels. Technical skills required include. **C, C++, VISUAL C++, VISUAL BASIC, X-WINDOWS MOTIF, GUI's, NT, TCP/IP/X25/X4000, PROGRESS, SAP, Relational Databases, INTERNET CONNECTIONS** and **ATM** (Communications not ATM machines). **REF:/04/EXE**

INGRES/ORACLE/SYBASE/GUPTA/OOD AND OOP — ALL LEVELS

Additional experience of: SQL, Forms, C and C++ required. We currently have client companies including Management Consultancies, Systems Houses, Systems Vendors, Bank and Finance clients looking for candidates with: Relational Database design, Database tuning, Systems Administration, DBAs, Pre/Post Sales and solid programming knowledge and expertise. Please call to discuss your particular requirements.

£18-£40K + benefits **REF SC/05/EXE**

C/C++/VISUAL BASIC/UNIX/WINDOWS 95/NT SERVER — DEVELOPERS

Software House and End Users in Finance, Banking, Manufacturing, Commercial, Scientific and Government application environments require excellent C skills. Both Windows development skills W3.1 SDK, NT, X-Windows and Visual Basic or strong C, C++ solid operating systems and good application knowledge are again much in demand. Software development experience is the key, and being able to deliver high performance, high quality, well specified software in competitive time scales. Opportunities vary from small to large software companies involved in expert systems, GUIs, Image Processing, GIS, EIS, Communications, Networking and Object Orientated Databases. Graduates through to senior software engineers/team leaders are required. Please call to discuss.

£14-£35K + benefits **REF: SC/06/EXE**

UNIX/VMS/WINDOWS 3.1/95/NT MFC/C/C++ — ALL LEVELS

A degree in computer of natural science, two years solid C/C++ programming experience and a sound understanding of UNIX, VMS or MS-DOS are required to work on large scale programs with user interaction. You will need an intelligent problem solving approach to work and be a quick learner to programmer software in an X-Windows, Windows SDK or NT environment, port software to different systems and liaise with customers to drive through product improvements. Excellent career opportunities for the right candidates.

£16-£28K **REF: SC/07/EXE**

LONDON/HOME COUNTIES WINDOWS SDK/NT DEVELOPMENTS

Senior Development Engineers **Analyst Programmers**
To £30K + benefits **To £27K + benefits**

Strong programming skills in C or C++ and Windows NT are pre-requisites for these positions. Experience in some of the following areas is also required: Windows 3.1/95, Windows NT, Windows SDK, MS C 7.0, MFC, Visual Basic, Visual C++ and Microsoft NT. Also desirable are Windows XVT libraries or networking skills. **REF: SC/08/EXE**

SOFTWARE ENGINEERS-SENIOR SOFTWARE ENGINEERS

Various Client/End Users, Software Vendors and Software Houses dedicated to strategic implementations of leading edge technology and integration of applications across different hardware and operating systems platforms require candidates to degree level with a scientific/technical development bias and 1-3 years experience. There are two main options

TECHNICAL DEVELOPMENT: Continued use of UNIX, VMS, MS-DOS, Windows NT (SDK, NT or X-Windows and Toolkits), Networking and Communications with companies offering technology based careers and management responsibility.

COMMERCIAL DEVELOPMENT: Using technical based skills already developed, but offering opportunities to apply analysis and design skills rather than remain 'a technical guru' in various environments including finance. Please call to discuss your particular career, growth and potential.

£12-£25K + benefits **REF: SC/09/EXE**

VISUAL BASIC SKILLS MUCH IN DEMAND - PLEASE CALL TO DISCUSS REF: SC/10/EXE
HANTS/LONDON - VIRTUAL REALITY DEVELOPERS - MFC, C++ - to £35K REF: SC/11/EXE
LONDON COMMS SPEC X25, X400 £40-60k REF: SC/12/EXE
C, C++/MFC - Countrywide REF: SC/13/EXE

S **the soft corporation**
Third Floor, 7-15 Roseberry Ave, London EC1R 4RP
Tel: 0171 833 2772 Fax: 0171 833 2774
email: jmcb@softcorp.demon.co.uk

JOIN THE WORLD'S LEADING SOFTWARE TEAM

Some company's are just more exciting to work for than others.

This company exists for Innovation and Creativity in developing some of the most advanced Software and Hardware Solutions for today's and tomorrow's Computing Applications. They seek exceptional candidates with both excellent Qualifications and Experience to originate, design and develop New and exciting Solutions for World Class Computing Products.

The development environment is Windows NT. All software is written in C++ using the latest development tools, making full use of object oriented analysis and design.

You will be a degree qualified min. 2.1 with 2 yrs or more experience gained with any of the following: C++, Win-NT, OOD, Class Libraries, System Architectures, GUI's, OLE, ActiveX, 32-bit Windows, MFC, Comms, Web Browsers, Java, Networking, Operating Systems and Device Drivers.

The company offers unlimited scope for career progression to Technical Consultant and Management. Excellent rates of pay are on offer (Salary to £36K on joining) with structured and regular reviews.

To join this World leader you just need enthusiasm for technology, a hands-on approach to design combined with the ability to pick up New Technology and skills quickly.

Call James Hunt for more information and make a positive career step.

We also urgently seek two young Software Engineering graduates with C++ and Windows SDK experience for a Telecoms/Networking systems designer based in Newbury area. Salaries to £25K. Ideal starting position for bright graduates with Comms and Windows design interests.

ASH Associates
COMPUTER RECRUITMENT CONSULTANTS
First Floor, 39 to 41 High Street
Ringwood, Hants, BH24 1AD
Email: ashassoc@globalnet.co.uk
TEL:(01425) 475480 FAX:(01425) 480807

**Call James Hunt,
Ron Cook or
Kaye Chambers Now!
Telephone
01425 475480
Email:
ashassoc@globalnet.co.uk**

See our jobs on the Net. http://www.globalnet.co.uk/~ashassoc

In and Around West Yorkshire

We have clients currently seeking the following:

Project Leader - a hands-on development role leading a small team in Visual BASIC/SQL/Access development under NT...	**to £30,000 + bonus**
Visual BASIC/MS Access Programmers for retail & back office systems........................	**to £20,000 + bonus**
Junior Visual BASIC Programmer with approx. 1 year's experience...	**to £15,000**
Analyst Programmers with Visual BASIC or 'C' or 'C++' for financial & insurance systems	**to £25,000**
Visual BASIC/SQL Developers for Windows/UNIX server systems	**to £23,000**
Project Leader, Visual BASIC/SQL for Windows/UNIX systems..	**to £30,000**

For your next career move around West Yorkshire telephone Vincent Atherton on Leeds (0113) 250 4560 or write to:

Airedale Recruitment

**Realtex Hs., Micklefield Lane,
Rawdon, Leeds, LS19 6 AX**

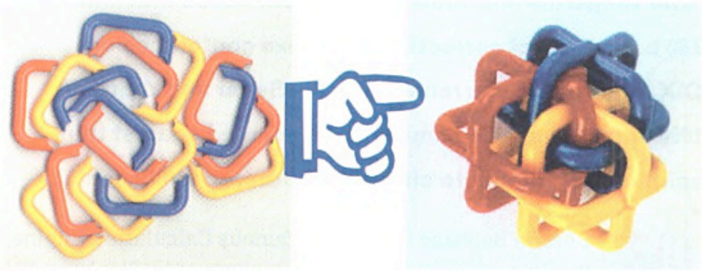
http://www.exe.co.uk
EXE
ONLINE

Twisted fingers

This month, Ctrl-Break received an anonymous parcel containing eighteen similarly shaped plastic objects, six red, six yellow and the rest blue. The only written text accompanying this suspect invention was a URL. Is that a sign that the Web is not powerful enough yet to convey physical objects – after XML, teleportation?

After having spent too much time working on cracking the damn puzzle, finished a roll of tape, and on a sure road to RSI, Ctrl-Break, on the verge of a nervous breakdown, eventually went to find the key to these Space Cubes on www.spacecubes.com.

Now, the cube is easy peasy, the link and the chain not much of a challenge but the double cube, the knot, the hypercube and the famous blob still remain far, far out of our reach at the time this story is written.



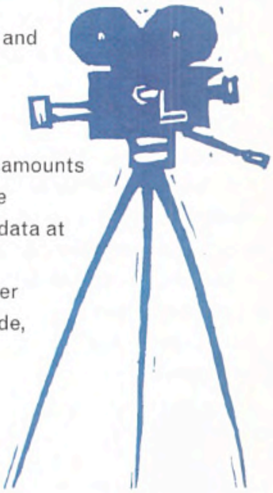
Life would be so much easier with six hands! ■

Writing software for Hollywood

A source from a well-known software tools company (one which writes compilers – hint), who will remain unnamed, got involved in the making of a spoof video. These are the rules he was forced to follow:

Twenty-four interesting things that you learn about computers in the movies...

1. Word processors never display a cursor.
2. You never have to use the space bar when typing long sentences.
3. All monitors display 2-inch high letters.
4. High-tech computers, such as those used by NASA, the CIA, or some such governmental institution, have easy-to-understand graphical interfaces.
5. Those that don't will have incredibly powerful text-based command shells that can correctly understand and execute commands typed in plain English.
6. Corollary: You can gain access to any information you want by simply typing 'ACCESS ALL OF THE SECRET FILES' on any keyboard.
7. Likewise, you can infect a computer with a destructive virus by simply typing 'UPLOAD VIRUS.' Viruses cause temperatures in computers, just like they do in humans. After a while, smoke billows out of disk drives and monitors.
8. All computers are connected. You can access the information on the villain's desktop computer, even if it's turned off.
9. Powerful computers beep whenever you press a key or whenever the screen changes. Some computers also slow down the output on the screen so that it doesn't go faster than you can read. The *really* advanced ones also emulate the sound of a dot-matrix printer as the characters come across the screen.
10. All computer panels have thousands of volts and flash pots just underneath the surface. Malfunctions are indicated by a bright flash, a puff of smoke, a shower of sparks, and an explosion that forces you backward. (see #7, above)
11. People typing away on a computer will turn it off without saving the data.
12. A hacker can get into the most sensitive computer in the world before intermission and guess the secret password in two tries.
13. Any PERMISSION DENIED has an OVERRIDE function.
14. Complex calculations and loading of huge amounts of data will be accomplished in under three seconds. In the movies, modems transmit data at two gigabytes per second.
15. When the power plant/missile site/whatever overheats, all the control panels will explode, as will the entire building.
16. If you display a file on the screen and someone deletes the file, it also disappears from the screen. There are no ways to copy a backup file – and there are no undelete utilities.



17. If a disk has got encrypted files, you are automatically asked for a password when you try to access it.
18. No matter what kind of computer disk it is, it'll be readable by any system you put it into. All application software is usable by all computer platforms.
19. The more high-tech the equipment, the more buttons it has. However, everyone must have been highly trained, because the buttons aren't labelled.
20. Most computers, no matter how small, have reality-defying three-dimensional, real-time, photo-realistic animated graphics capability.
21. Laptops, for some strange reason, always seem to have amazing real-time videophone capabilities and the performance of a Cray Y-MP.
22. Whenever a character looks at a monitor, the image is so bright that it projects itself onto his/her face.
23. Computers never crash during key, high-intensity activities. Humans operating computers never make mistakes under stress.
24. Programs are fiendishly perfect and never have bugs that slow down users. ■

8086 and all that

'The Tangerine Microtan 65 was a 6502 based machine, not Z80 based' – fact correction in the exe conference on CIX.'History as you remember it' – Sellar and Yeatman, 1066 and all that. (Or some such: it seemed against the spirit of the original to check the quote, so we haven't.)

1835 Charlie Babbage invents his famous Calculating Engine, the first ever computer, which is powered by Stephenson's Rocket, and consequently four foot eight and one half inches wide. Genius *père et fils* engineering double act Marc and Isambard Brunel offer to create a seven foot version but, as a result of an unfortunate error in the copying of the plans, instead build the South Devon Atmospheric Railway. Stung by disappointment, Babbage switches his efforts to a dual gauge Analytical Engine, and places the first ever advertisement for the first ever programmer – an advert which has been used ever since as a model by recruiting agencies ('Mr Charles Babbage of 147 Coprocessor Lane, Westminster Village seeks the assistance of a programmatic person, to aid him with his Government researches. On-the-job training will be given, but applicants with Visual C++, Oracle, Tuxedo, UNIX and NT4 will be preferred. Babbage Calculating Engines Ltd is an equal opportunities employer. Either sex may apply for this post, but breast feeding abilities considered an advantage').

Babbage is lucky enough to secure the services of Countess Ada Lovelace, at that time best known as the star of many early-Victorian pornographic lithographs. Although Babbage's working relationship with Lovelace is excellent, his interest in a practical engine for analysing coefficients of Taylor's theorem according to sundry measurements of external objects wanes, and his interest in going to bed in the afternoon waxes. Babbage's work is never completed; but this doesn't matter because nobody else does any work on it either. Thus England is Top Nation in computing for the next 100 years – a Good Thing.

1936 In Hitler's Nazi Germany, a beastly German person whose name is not important attempts to reinvent the computer 10 years too early, using the bakelite knobs that have fallen off Ferguson wireless sets. Unfortunately for Herr Not Important, Lawrence of Arabia and Lawrence of Olivier are soon both on the case, and turn the tables on him using a black-and-white John Buchan plot. Thus the Empire is preserved for people who pronounce the word hands as 'hends'.

Meanwhile, in England, mathematician and pipe smoker Alan Turing proposes his famous Turing Machine, a contraption which can display and interpret a sequence of odd-looking, arbitrary symbols. Predictably he fails to interest the British Government in his idea – with all resources committed to the Appeasement Effort, there is no money available to fritter away on infinitely long paper tapes – but it is taken up by the hotel and catering industry, who subsequently manufacture a slightly modified version in large quantities. This commercial version is named after the man who effected the modifications: Professor J R Fruit.

1942 At Bletchley Park, Turing invents Robbie the Robot, a fantastic automaton which can decrypt the secret Enigma codes of the U-boats, play chess, accurately forecast greyhound and horse racing results and make polite smalltalk at dinner parties. Prime Minister Winston Churchill wants to send Robbie on a special mission to kidnap Hitler and end the war; but he is foiled in this wish when it turns out that Robbie is unable to override his own Prime Directive: the robot cannot harm a human being no matter how evil, or how silly her moustache. Despondent at his failure, Robbie sets about a career as a Shakespearean actor, and eventually achieves fulfilment; his portrayal of Ariel in a 1950s production of *The Tempest* being particularly well received.

1950 The American company IBM launches UNIVAC, a dual vortex dataprocessor which beats as it sweeps as it cleans. Not to be outdone, British company Lyon's Maid launches Leo, a gargantuan hulk of machinery which turns out to be useless for office automation, but excellent for keeping ice lollies cold.

1954 At Manchester University, Alan Turing is fatally wounded in a laboratory accident, when the subject of his latest Turing Test turns out not to be, as he believed, a Mark II Electronic Brain running at half clock speed, but instead an irate builder's labourer, Mr Arthur Wit. Turing dies tragically on the operating table while surgeons battle to retrieve his own left upper canine from inside his kidneys.

Back at the Manchester University labs, Turing's former colleagues make what turns out to be a fatal mistake for British Computing: they decide to abandon his theoretical efforts on computer science, and instead concentrate on his pipe smoking work. Thus America becomes Top Dog in Computing, a Bad Thing.

1960s The UK continues to lag behind. In 1966, pressed by the Wilson Government 'yellow glimmer of know how' technology policy, Edinburgh University manages to build a noughts-and-crosses machine which, while not unbeatable, 'puts up a jolly good show'. It is opened by the Beatles, and in an exciting and tense match the machine loses five games to nil to Ringo (this was of course before mind-bending drugs took the edge off Ringo's noughts-and-crosses abilities). In 1968, English Home Counties Electric Valves produces Sir Ernie, the Premium Bond random number generator, a primitive forerunner of the exciting and leading edge technology which today drives the National Lottery.

1971 Edsger Dijkstra delivers his famous ACM paper dealing with naming conventions: 'Excessive consonants considered hard to spell'.

1979 At last a triumph for Great Britain! Sir Clive Sinclair's Z80-based Tangerine Microtan 65 launches. With its handsome (for its time) 'dead flesh' keyboard, its ability to drive a modified TV at a startling (for its time) rate of 405 lines per minute and its startling (for its time) sixteen byte memory, all for a knockdown price of £599.99, the machine is an immediate hit. For a while, it outsells even the former market leader the Apple Big Jobs, but eventually tragedy strikes when somebody else offers a much better computer for rather less money.

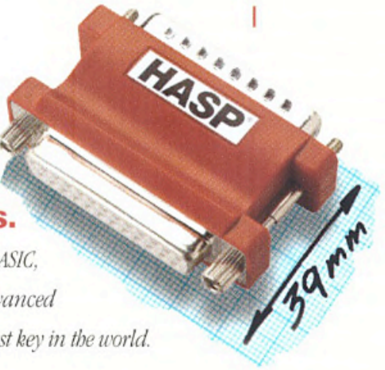
1980 IBM asks Gary Kildall to invent an operating system for the future PC, to be called OS/2. Kildall refuses, and then makes matters worse by flying around and around IBM headquarters in a biplane taunting the IBMers about 'their silly blue shirts'. IBM hires Bill Gates to blow Kildall out of the sky with an anti-aircraft gun, and as a token of gratitude for accomplishing this successfully hands over the rights to all computing technology forever.

And that is the end of computing history. ■

MORE DEVELOPERS PROTECT.

HASP® PROTECTS MORE.

All over the world, more developers are choosing to protect their software against piracy. They're protecting more products, on more platforms, with better protection – and selling more as a result. And more of these developers are protecting with HASP. Why? Because HASP offers more security, more reliability and more features than any other product on the market. HASP supports the most advanced platforms, including Win NT, Win95, Win32s, Win 3.x, OS/2, DOS, Mac OS, NEC, UNIX and LANs. To learn more about how you can protect better – and sell more – call now to order your HASP Developer's Kit.



yes It runs with NetWare · Microsoft Windows Compatible · Mac OS · Ready for OS/2 Warp

Rated No.1! NSTL

HASP Packs More Into Less.

Based on a full-custom ASIC, HASP packs the most advanced protection into the smallest key in the world.

01753 622266
www.aks.com

ALADDIN™
The Professional's Choice

United Kingdom	**Aladdin Knowledge Systems UK Ltd.** Tel: +44 1753-622266, Fax: +44 1753-622262, E-mail: sales@aldn.co.uk
North America	**Aladdin Knowledge Systems Inc.** Tel: (800) 223 4277, 212-564 5678, Fax: 212-564 3377, E-mail: hasp.sales@us.aks.com
Int'l Office	**Aladdin Knowledge Systems Ltd.** Tel: +972-3-636 2222, Fax: +972-3-537 5796, E-mail: hasp.sales@aks.com
Germany	**FAST Software Security AG** Tel: +49 89 89 42 21-37, Fax: +49 89 89 42 21-40, E-mail: info@fast-ag.de
Japan	**Aladdin Japan Co., Ltd.** Tel: +81 426-60 7191, Fax: +81 426-60 7194, E-mail: sales@aladdin.co.jp
Benelux	**Aladdin Software Security Benelux B.V.** Tel: +31 24-648 8444, Fax: +31 24-645 1981, E-mail: aladdin@worldaccess.nl
France	**Aladdin France SA** Tel: +33 1 41-37-70-30, Fax: +33 1 41-37-70-39, E-mail: 100622.1522@compuserve.com

■ **Aladdin Russia** 095 9230588 ■ **Australia** Conlab 03 98985685 ■ **China** (East) Shanghai LIRI 021 64377828 ■ **China** (North & Central) Feitian 010 62567389 ■ **Czech** Atlas 02 766085 ■ **Denmark** Berendsen 039 577316 ■ **Egypt** Zeineldein 02 3604632 ■ **Finland** ID-Systems 09 8703520 ■ **Greece** Unibrain 01 6756320 ■ **Hong Kong** Hastings 02 5484629 ■ **India** Solution 011 2148254 ■ **Italy** Partner Data 02 26147380 ■ **Korea** Dae-A 02 8484481 ■ **Mexico** SiSoft 091 80055283 ■ **Poland** Systherm 061 480273 ■ **Portugal** Futurmatica 01 4116269■ **Romania** Ro Interactive 064 140383 ■ **Singapore** ITR 065 5666788 ■ **South Africa** D Le Roux 011 8864704 ■ **Spain** PC Hardware 03 4493193 ■ **Sweden** Kordab 455 307 300 ■ **Switzerland** Opag 061 7169222 ■ **Taiwan** Teco 02 5559676 ■ **Turkey** Mikrobeta 0312 4670635 ■ **Yugoslavia** Asys 021 623920

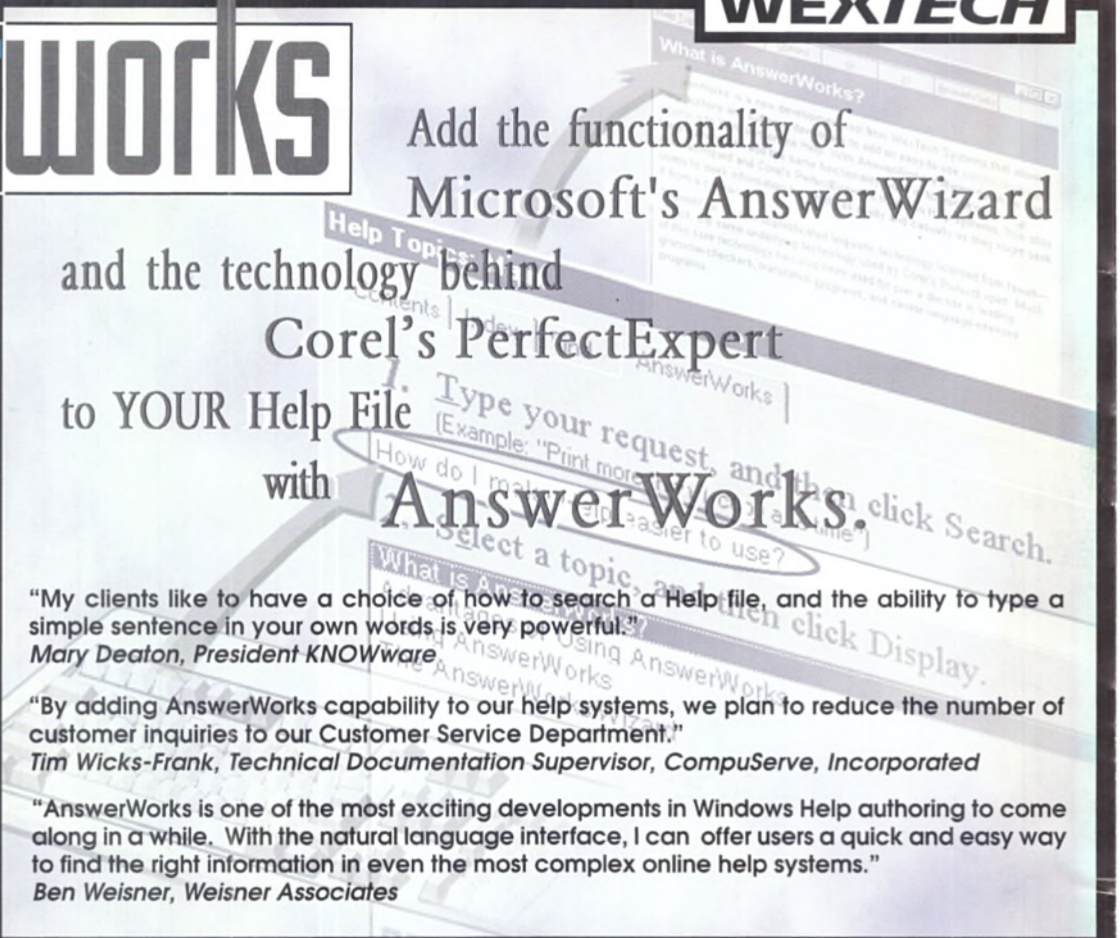
ENQUIRY NO. O43

"Intelligent Tools for Intelligent Documentation!"

WEXTECH

NEW

answerworks


What is AnswerWorks?
Help Topics
Contents
Index
AnswerWorks
1. Type your request, and then click Search.
(Example: "Print more ...
How do I make ... easier to use?
2. Select a topic, and then click Display.
What is AnswerWorks?
Using AnswerWorks
AnswerWorks

Add the functionality of
Microsoft's AnswerWizard
and the technology behind
Corel's PerfectExpert
to YOUR Help File
with AnswerWorks.

AnswerWorks brings state-of-the-art linguistic technology to YOUR application's Help. Online Help users will be able to get answers to their plain language questions. They simply type the questions in their own words and AnswerWorks will find and display the most pertinent topics. Reduce your support costs with AnswerWorks! Get faster, easier and more accurate access to Help.

"My clients like to have a choice of how to search a Help file, and the ability to type a simple sentence in your own words is very powerful."
Mary Deaton, President KNOWware

"By adding AnswerWorks capability to our help systems, we plan to reduce the number of customer inquiries to our Customer Service Department."
Tim Wicks-Frank, Technical Documentation Supervisor, CompuServe, Incorporated

"AnswerWorks is one of the most exciting developments in Windows Help authoring to come along in a while. With the natural language interface, I can offer users a quick and easy way to find the right information in even the most complex online help systems."
Ben Weisner, Weisner Associates

doctohelp Lets YOU:

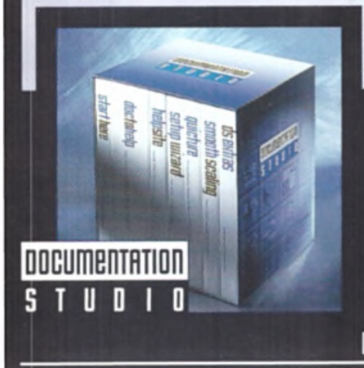
- Create professional documentation AND Windows 3.x, WindowsNT & '95, Help Files *simultaneously*.
- *Easily incorporate* hypertext links, audio, video and graphics into your Help File.
- Format your documentation and Help *independently* from a *single source*.
- Mark material for inclusion in the *"Documentation Only"* or *"Help Only"*.

Also available *Documentation Studio*: the *complete Doc-To-Help suite*. Special upgrade pricing for registered Doc-To-Help users.

Documentation Studio includes: Doc-To-Help, Helpsite, Quicture, Smooth Scaling, Setup Wizard, Lotus ScreenCam and Bookmaker ClickBook.

Create Printed Manuals and Windows Help from a Single Source!

doctohelp



DOCUMENTATION
STUDIO

AnswerWorks £399
Doc-To-Help £315

ENQUIRY NO. 044

For more information or to order call:
Soft/Export Ltd.
FreeFone: 0800 973 098
FreeFax: 0800 973 327
E-Mail: info@softexport.com

Designed for Microsoft Windows 95

Download a free evaluation copy of AnswerWorks! www.wextech-europe.com

All brands and product names are trademarks or registered trademarks of their respective companies.