The International Journal of OMNIS™ Developers
A Publication of The DLA Group Pty Ltd

Omni 7 Directional

Issue 7
September / December 1996

Web Enabler

The OMNIS Web Enabler

provides a full- featured set of extensions for applications created using the OMNIS 4GL command language. Applications built with these extensions can access any web site and extract information from those sites. OMNIS Web Enabler allows OMNIS developers access to other Internet services such as File and Mail services.

OMNIS Web Enabler includes translators to quickly convert any Report or list structure written in OMNIS to an HTML tagged report. Web Enabler also extends the built-in OMNIS graphic support to popular Internet formats such as GIF and JPEG. OMNIS Web Enabler adds easy-to-use commands to the OMNIS 4GL coding language, shortening the learning curve of successful web application development. OMNIS Web Enabler provides both high-level and low-level protocol support concerning extensions and service protocols.

The OMNIS application development tool suite allows the developer total cross-platform application development. Whether your choice is Windows, Macintosh, Power Macintosh or OS/2, the OMNIS solution allows you to develop once and deploy anywhere including the Internet.

Protocol Extensions

OMNIS Web Enabler provides low-level protocol support allowing a developer to build a connection to any network service. OMNIS Web Enabler utilizes the following protocol extensions to accomplish this.

TCP/IP Sockets

TCP/IP is the major communications protocol for connectivity in a network or Internet environment. TCP Sockets extensions provide low level access that permits OMNIS Web Enabler developers to directly access network programming from within OMNIS.

Socket Threading

OMNIS Web Enabler allows the developer the ability to wait for a response from the remote host when establishing a connection, by providing both blocking and non-blocking sockets. Socket threading allows a developer to simulate a multi-threaded OMNIS environment, even though OMNIS is a single threaded application.

Multiple Sockets

OMNIS Web Enabler provides support for both multiple clients and multiple functions simultaneously.

Platform Support

OMNIS Web Enabler provides TCP support on Windows 3.1, Windows NT, Windows 95 and OS/2 through the Winsock support libraries. On Macintosh and Power Macintosh, the support is provided by MacTCP libraries.

Service Protocols

OMNIS Web Enabler also provides developers with high-level protocol support for specific Internet services including File, Mail, and the Web.

File Services

Continued page 8

DLA OMNIS Asia Pacific Conference!

See page 3 for details.

In This Issue...

OMNIS Web Enabler	1
Blyth Announcements	2
Conference Report	3
Success Story - FedEx	4
Client/Server Migration	5
DLA Snapshot	7
New look DLA Web Site	8
OMNIS 7³ Training	10
Notation	11
Prometheus DAMs	13
Notation Tips	14
OMNIS Videos	15
Code	16
Tech Tips *Resource Editing*	17
Tech Tips *Porting OMNIS Externals to Win95 & WinNT*	19
Internet Connectivity	20
OMTOOLS	22
List Model	23

BLYTH SOFTWARE ANNOUNCES TIM NEGRIS TO BECOME EXECUTIVE VICE PRESIDENT

Marketing Veteran in Key Role as Company Readies 'Net Development Tools

FOSTER CITY, CA (November 12, 1996) — Blyth Holdings, Inc. (NASDAQ: "BLYH") announced today that Tim Negris has joined the company in the newly-created role of Executive Vice President, Marketing and Development.

Mr. Negris joins Blyth with over 15 years of experience in software marketing. Most recently as Vice President, Sales and Marketing for IBM's multi-billion dollar Software Solutions Division, he spearheaded marketing for their award-winning database, workflow, digital media and application development products. Mr. Negris was also instrumental in establishing joint development and marketing programs with emerging Silicon Valley new media and electronic commerce ventures, as well as in affirming IBM's presence as a software provider in emerging markets such as China, Eastern Europe and Latin America.

Prior to IBM, as Vice President, Server Product Marketing at Oracle Corporation, Mr. Negris was one of the primary architects of Oracle 7's highly successful market launch. Previously he served as an executive at Sybase and a number of other software companies.

Mr. Negris joins Blyth at a strategic time, as the company readies its new product line for entry into the hot arena of electronic commerce, while completing a major revamping of its well-established OMNIS application development tools.

"Tim's experience with introducing new technologies will be a key factor to Blyth's success in the new generation of web-enabled client/server tools," said Michael J. Minor, Chairman and Chief Executive Officer at Blyth Software.

"Tim Negris has always had an outstanding vision of the software industry. He understands customer's needs and has a proven track record in the highly competitive field of software marketing. Blyth's excellent product line has found a worthy spokesperson," stated Peter Kastner, Vice President at Aberdeen Group.

OMNIDirectional is produced by DLA Technology as a service to users of Blyth Software's OMNIS™

D L A

TECHNOLOGY MANAGEMENT CONSULTANTS

A Bi-monthly publication of:
The DLA Group Pty Limited
A.C.N. 003 329 039
Level 3, 35 Clarence Street
Sydney NSW 2000 Australia
Tel: (+61 2) 9262 2255
Fax: (+61 2) 9262 2290

DLA Technology
- Change Management
- Database Design
- Consulting Services
- Software Development
- Internet Advertising
- OMNIS Software distribution and development

DLA Software
- Solicitor's Companion practice management software for solicitors
- Counsel's Companion practice management software for barristers
- Litigation support development and services
- Databases for marketing, human resources and library management

Email: info@dlagroup.com.au
URL: http://www.dlagroup.com.au
Ftp: ftp://ftp.dlagroup.com.au

Editor: Stephen Miller
stephen_miller@dlagroup.com.au

Managing Director: David P. Lewis
david_lewis@dlagroup.com.au

Layout: Phil Kingsmill
phillip_kingsmill@dlagroup.com.au

OMNIS and OMNIS 7 are Trademarks of Blyth Software Inc. All other product or service names mentioned herein are trademarks of their respective owners.

Copyright (c) The DLA Group Pty Ltd 1996. All rights reserved.

BLYTH SOFTWARE ANNOUNCES STRATEGIC PARTNERSHIP WITH INFORMIX SOFTWARE

FOSTER CITY, CA October 22, 1996 — Blyth Software, Inc. (NASDAQ: "BLYH") announced today a new strategic partnership with Informix Software that includes product integration and joint marketing programs.

The engineering projects will initially focus on integrating Blyth's client/server application development environment, OMNIS®, and its application management tools, Change Management and Version Control Systems, with Informix's Dynamic Scalable Architecture(TM) database server products. The second phase of engineering projects will integrate Blyth's new suite of Intranet and Internet products with Informix's DSA products. Global co-marketing programs of the new Informix-Blyth products will commence in the fourth quarter of 1996 and focus initially in Latin America, where both companies have large enterprise success. Informix and Blyth have recently been selected as standards for Banco Internacional, S.A. (BITAL), one of the largest financial services institutions in Latin America.

"We are pleased to announce this strategic partnership with Blyth," stated Jeff Hudson, Vice President for Business Development at Informix. "Blyth's development and management tools have a proven track record for creating successful complex electronic commerce applications in large corporate accounts. Our joint success at BITAL is a good indicator of the power of our combined technologies and represents a joint success in the financial services industry."

"Informix has become the worldwide leader in database technology, and the integration of Illustra's object relational technology will extend their competitive advantage," stated Michael J. Minor, Chief Executive Officer and Chairman of Blyth Software. "The integration of our client/server, Intranet and Internet tools with Informix's database products should create a suite of very strong offerings for the rapidly changing electronic commerce marketplace. Our recently formed partnership with BITAL continues Blyth's success in the financial services market, represented by such strategic customers as JP Morgan, World Bank, Citibank, Harris Bank, AON Insurance, St. Paul Companies, American Express, Chicago Board of Trade, Federal Reserve of Minnesota and Barclay's."

Alejandro Rojas, Chief Information Officer at BITAL, commented, "The partnership between Informix and Blyth brings together two leaders in technology. We are making significant future commitments to their joint products. Our entire electronic banking strategy is based upon the success of this partnership."

Peter Kastner, Vice President at Aberdeen Group, stated, "The partnership between Informix and Blyth is a win-win, combining great database technology with superb development tools. Blyth's web tools integrated into the INFORMIX-ONLINE DYNAMIC Server(TM) should provide state of the art Intranet and Internet products. Informix's marketing programs should also enhance Blyth's visibility in the corporate arena."

DLA OMNIS Asia Pacific Conference

November 25, 1996

The DLA OMNIS Asia Pacific Conference held on November 25 was an enormous success. With sponsorship by Apple Computer Australia and Genitech Pty Limited, presentations by our US OMNIS experts and hands-on demonstrations of Prometheus Beta 1, the 60 delegates were well rewarded.

Jerry Lipscomb

Jerry Lipscomb, President of The Dynabyte Corporation in Denver Colorado <jlipscomb@dynabyte.com> was in great demand as he demonstrated the new OMNIS products.

WebEnabler™ enables Blyth's client/server applications to easily access 'Net services such as HTML page creation, electronic mail, news groups and TCP/IP facilities. Importantly, WebEnabler can connect an application to any 'Net site and extract information from the site.

Web Data Server™ offers true server implementation as a gateway to OMNIS or other popular relational databases. It will support Oracle, Sybase, Informix and ODBC as well as OMNIS and exhibits developer APIs for application plug-ins. Web Data Server is a high end product to manage a corporation's distribution of data via the 'Net.

Jerry co-hosted the OMNIS Underground Conference in Denver this October and he has made available all Videos of the critical sessions. So those of you who missed the DLA Conference can still catch up on the latest OMNIS Tips and Tricks.

Dewey Gaedcke from CDS in Austin Texas <dgaedcke@compdata.com> provided an excellent insight in the SQL world as well as an expose on Object Oriented OMNIS programming.

Next year DLA hopes to offer a 2 day conference in light of the many 'complaints' received from delegates that they just wanted more time and more topics!!

Dewey Gaedcke

Steve Larbig

I look forward to seeing even more of you at our 1997 DLA OMNIS Asia Pacific Conference.

David P. Lewis
Managing Director
<david_lewis@dlagroup.com.au>

At-A-Glance

Strategic Objectives:

- Implement client/server technology that supports transaction-based electronic commerce applications worldwide.
- Lower administrative business costs associated with aircraft maintenance and quality control.
- Replace paper-based workflows with electronic exchange of information.
- Create more flexible work processes, system control, and data management through reengineering.

The Solution:

An OMNIS-based client/server solution that tracks aircraft component and safety information was implemented by the Air Operations Division. Over 150 employees use the system to schedule repair and maintenance, track vendor specialisation and legal requirements, manage inventory, process accounting transactions, and execute financial forecasts. OMNIS' electronic commerce architecture will support over 800 users worldwide via the Internet and Intranet.

Benefits:

The reengineered applications have lowered the cost of business through a reduction in processing errors, more efficient accounts payable processes, and faster access to information. Flexible data management reduces islands of information and allows the Engineering, Quality Control, Purchasing, Accounting, and Maintenance departments to interface seamlessly.

FedEx Federal Express

REENGINEERING AIR OPERATIONS MANAGEMENT WITH OMNIS' ELECTRONIC COMMERCE TECHNOLOGY

An OMNIS Success Story in Transportation

Leaders in the Business

Federal Express, based in Memphis Tennessee is a world-known express package delivery service with global operations, over 100,000 employees, and annual revenues in excess of $10 billion. Federal Express has a fleet of over 500 aircraft that must be tracked according to Federal regulations for maintenance and safety at the component part level. To accomplish this, the Air Operations Division must interface with multiple departments within Federal Express as well as external vendors and regulatory agencies.

"With OMNIS there was no limit to the solutions we could build to improve our work processes and implement electronic exchange of information. We expect to Net-enable these solutions and see endless possibilities for new electronic commerce applications with OMNIS technology in the future."

Bill Meyers
Program Administrator, Federal Express

As the IT group examined its existing operations and mainframe-based technology, it was clear that nothing short of a "paradigm shift" would be sufficient to position the company for the future. Operationally, each department had its own stand alone work process, making training and system maintenance expensive. Paper-based transaction processing was time consuming. A system was needed that could maintain, process, and route transaction information electronically and automatically. Federal Express began the search for a client/server product designed to support global, enterprise operations.

OMNIS: Client/Server Technology Architected for Complex Electronic Commerce

As a strategic leader, Federal Express deploys technology to provide excellent products and service. For example, customers can access information regarding package delivery status from the FedEx World Wide Web home page. This lessens the boundaries imposed by time and geography. Federal Express adopts products that allow them to push these boundaries farther and faster than their competition. As Program Administrator Bill Myers summarised, "we wanted a product that would meet our immediate need to reengineer mainframe to client/server solutions, but it had to be a product designed to support seamless extension into the Internet/Intranet model of electronic commerce — the model that will be our future." After a six month selection process, Bill and his team chose OMNIS because it was cross-platform and would

Continued page 6

The twists and turns of client/server migration

The move from mainframes to client/server is never an easy task, and sometimes a workgroup project can balloon into enterprisewide significance. Just ask Banco Internacional, one of Mexico's largest banks.

Jorge Sosa can't pinpoint exactly the moment when his search for a new client/server application development environment spilled over the boundaries of his workgroup and became a matter of enterprisewide importance. But once the project passed that turning point, every step took on a new significance. Suddenly, a big chunk of the competitive edge of Banco Internacional was hanging in the balance.

From a distant No. 8 among Mexican banks three years ago, Bital (as Banco Internacional is known in Mexico) has rocketed to the top echelons. Bital's rapid rise has come in part through its aggressive use of technology. At a time when banks across Mexico routinely close for the day in midafternoon to allow the books to be balanced by 7:00 P.M., Bital has centralised its processing and brought every branch on line so that transactions can be reconciled in real time.

That's the good news. The less-good news is that the technology infrastructure that brought Bital to the dance is about ready to sit one out. The IBM 9021, Bital's biggest mainframe, was configured to carry the load for about five years. But after only two years, the bank's rapid growth has nearly maxed the machine out.

According to Sosa, an IT implementation manager with Bital, one problem is the growing number of 3270 sessions. Each of the 10,000 PCs at Grupo Financiero, the financial services corporation that includes Banco Internacional, has the ability to crank up a session, as do more than 2,500 corporate customers that dial in to use the bank's cash management system. The mainframe can't support all of this activity at once. This puts a lid on the bank's ability to meet internal information needs as well as to deploy more copies of its popular cash management system.

Another challenge is the inevitable mainframe app dev backlog. One part of Bital's development team of more than 150 programmers works in Computer Associates' IDEAL accessing CA-Datacom files. The other part programs in COBOL and assembler against a data engine from Hogan Systems. Nightly batch runs replicate specific data from the Hogan banking software to the CA-Datacom files, making the corporate data resource more readily accessible. Nevertheless, a serious backlog of app dev requests has stacked up.

At-A-Glance
Grupo Financiero Bital

The company: Grupo Financiero Bital, Maxico City, is made up of eight financial services companies. The largest, employing 8,000 of the group's 12,000 employees, is Banco Internacional, which is one of Mexico's biggest banks.
The problem: The bank's IT department, which is in charge of technology for the entire group, has to keep its overburdened mainframes from throttlink the bank's aggressive growth.
The solutions: IT is moving to dramatically reduce the number of 3270 sessions by switching key apps from screen scrapers to client/server. It is also decreasing the backlog of mainframe-programming requests-and, at the same time, increasing the flexibility and responsiveness of IT-by shunting many new projects to C/S development teams.
The IT infrastructure: Bital has an IBM 9121 mainframe and a 9021 mainframe (IBM's largest in Latin America) running MVS, CICS, Computer Associates' CA-Datacom database and IDEAL 4GL, and Hogan Systems' banking software based on VSAM files. On the desktops are 8,000 PCs and 2,000 Macs, all with 3270 capabilities. The mainframe communications must also handle 2,500 corporate customers who dial in to 3270 sessions.

Federal Express (cont)

support integration with existing mainframe systems. Bill added, "Blyth's Internet strategy is based on a proven track record with scalable, enterprise enterprisesolutions. We are confident that OMNIS architecture is a solid foundation from which to extend the solutions we build to the Net."

Solutions for Efficient Air Operations Management and Engineering Authoring
Federal Express reengineered their applications with OMNIS. The system fully automates transactions requiring extraction and manipulation of mainframe-based data. Today's production system supports 150 users, and after a full-scale roll out, is expected to support 800-1,000 users worldwide and will be Net-enabled. The modules of the system are fully integrated and use an OMNIS front-end and Sybase back-end.

The solution automates transaction processing for maintenance and replacement of aircraft components. It helps ensure that vendors are in compliance with Federal regulations. It tracks scheduling of work and automates preprocessing authorisation for service and repair orders for components. Information on the status and availability of the aircraft components is maintained along with the specialisation of vendors and their certification to perform specific work on components. Invoice processing is now automated. Engineering uses the system for authoring, managing and delivering documentation. Document exceptions and anomalies are flagged by the system as well. Electronic project tracking supports real time process management.

Reducing Costs and Improving Work Processes
Federal Express now has a powerful, flexible and scalable system. Instead of taking *days* to process an engine-related work order, it now takes *minutes*. Federal Express can take better advantage of vendor discounts given to customers who pay invoices quickly. Estimating total aircraft maintenance costs is easy because item-level costs for the smallest hardware components are easy to retrieve. Such estimates facilitate accurate budgeting and financial forecasting on the cost of business operations. The ability to share information across departments has allowed Federal Express to reengineer and improve internal communication, project tracking, and information management.

The twists and turns of Client/Server migration (cont)

A workgroup problem
Taken together, these all-too-familiar mainframe problems are beginning to threaten Bital's continued success in a highly competitive marketplace. But although this enterprise technology challenge has been setting off alarms at some levels in the bank, that's not where our story begins. Instead, we turn to a small group and its small problems.

It's February 1995. Sosa has been with the bank for about three months. He heads a small group of front-end developers in IT's Special Projects area. They work in CA's Clipper, Microsoft's FoxPro, and ClientSoft's ClientBuilder, a screen-scraper app dev tool. Some of the group's most important "customers" are bank executives who don't want to open up 3270 sessions on their Macs. ClientBuilder gives Sosa's team a way to connect the execs to the Hogan and CA-Datacom data without forcing them to learn how to run the 3270 apps.

But then, as now, a taste of technology only gave the execs anappetite for more. And when that happened, Sosa found himself in a bind. His users kept wanting one more new feature and another little twist, but there was only so much his team could do with a screen scraper and two dBase descendants. And he couldn't turn to the mainframe programmers and get them to modify the screens he was scraping. That would have made it easier for him to accommodate the execs, but the mainframe team was way too busy to bother with little requests like his.

So Sosa began looking for a new tool, a strong, flexible GUI builder. It had to give him direct access to legacy datastores. And it had to work on both PCs and Macs. With a tool like that, Sosa knew he could increase his team's productivity and be much more responsive to the needs of his users.

After extensive reading on client/server and rapid application development tools, Sosa started to take a serious look at PowerBuilder from Sybase's Powersoft. At this point, Alejandro Rojas, Bital's IT corporate director, suggested that Sosa also take a look at Blyth Software's Omnis, which was being considered by another Bital Special Projects team. Sosa was ready to move forward, so he got the nod to take over the Omnis evaluation.

The scope expands
Meanwhile, IT's Credits and Loans group was looking at PowerBuilder and Microsoft's Visual Basic. Juan Carlos Aburto, the head of the section, was pushing for a Windows C/S tool, but the burden of a big project meant he couldn't spare that much time and attention to the search. Rojas stepped in, asking Sosa to work with Aburto's team to see to it that the Special Projects evaluation select a product Credits and Loans would be happy with, as well.

Opening two-way communications with the other group was a chance for Sosa to add technical breadth and depth to his evaluation team. But even more important, it was an opportunity to include a broader swathe of IT in the evaluation and to begin some serious consensus building.

Another major result of the collaboration was the elimination of Visual Basic. Whereas Credits and Loans had been looking for a Windows-only tool, Sosa was charged with finding a cross-platform solution, and Visual Basic clearly was not that.

As Sosa's evaluation gathered steam, it also gathered weight. He was no longer picking a solution just for his workgroup: His choice would clearly send ripples through IT as a

whole. Discussions about how to lighten the load on the 9021 and how to ease the mainframe-programming backlog took on added significance. "I began to see how this evaluation fit with everything else," recalls Sosa. "I understood better the problems that IT faced. Now, I was looking for a solution to more than just my problems."

Contemplating the enterprise perspective, Sosa realized that aserious plunge into client/server would require more than a good front-end tool. It would take a strong middleware product to create the kind of flexible, transparent, distributed data access scheme that would reduce the number of 3270 sessions and break the bottleneck that the mainframe had become. So Sosa launched a middleware evaluation in tandem with the ongoing app dev tool evaluation. Information Builders' EDA/SQL and Sybase's OmniConnect emerged as the leading middleware contenders. Either one of the two could work with either of the app dev tools Sosa was considering.

An intensive evaluation consisted of building an application with three buttons. The first handled a SQL query to the CA-Datacom database on the mainframe. The second ran a remote procedure call to a mainframe program written in COBOL under CICS, kicking off a Hogan transaction that would pull an account balance. The third combined the information from CA-Datacom and Hogan into a local database and presenting the result in GUI form. In the end, Sosa and his team settled on the Omnis and EDA/SQL combination. Thus, in the last days of December 1995, nearly eleven months after Sosa began his search for a better tool for his little group, Bital signed corporate licenses with Blyth and Information Builders.

The first project

Today, Bital has trained about 70 programmers in Omnis. The first major project using the tool has been to revamp the bank's high-visibility, mission-critical cash management system.

About two years ago, Bital had developed the system on a mainframe to let commercial customers transfer money from one of their accounts to another. About a year ago, the bank had added a very popular feature: direct payroll deposit. It had been implemented using the file transfer features of the 3270 emulation program, which customers ran on their PCs or Macs. The files were transferred from a customer's desktop to a TSO region on the mainframe for processing against the Hogan database.

The feature was an enormous success, and that has been the system's downfall in many ways: Doing file transfers in a 3270 emulation is tricky for users. It's handled differently on the Mac than it is on the PC. The telephone lines are very bad in Mexico. TSO sessions are very costly on a crowded mainframe. And everybody in Mexico gets paid on the 15th and on the 30th. So when more than 2,500 Bital customers all jumped on the same shaky infrastructure on the same day, there were bound to be problems.

In January of 1995, IT's Electronic Banking division had promised to deliver a new cash management system—client/server, GUI-based—in a year. But the perpetual crush of problems with the old system had kept the division from even getting the project started. In July, the project had been given to another development group. But three months later, they hadn't made any headway either.

At the end of September, IT gave the project to Sosa, who built a separate help desk for the old cash management system and started designing the new system. He put together a 15-member development team, 10 of whom were devoted entirely to the new system.

Scrapping the prototypes

At the beginning of 1996, the team started working with Information Builders to implement EDA/SQL. A team from Blyth arrived, took a look at the front-end and back-end prototypes Bital had built, and, according to Sosa, said, "No, no, no. That's not the way to do it." The prototypes were scrapped, and the Bital team started to learn the new approach to programming with middleware. Finally, after months of installing, configuring, and fine-tuning EDA/SQL on MVS, the middleware was working, and Sosa's team did its first integration test in May.

Two months later, the new cash management system was rolled out to 10 users for three weeks of beta testing. Sosa plans to do some careful capacity planning based on his experience with those initial users. The tentative plan is to continue rolling out the new system to about 50 more users per month, upgrading resources, equipment, telephone lines, and so on as the project proceeds.

If all goes according to plan, Bital's customers will find the new system a huge improvement over what they had, and the IT department will have taken a significant step toward keeping the bank on its fast track to the top. That's a big return on letting the manager of a small group try to find a better way of working.

DLA Snapshot

So who is Amanda?

Wonder no more.
Amanda loves listening to hold music and playing silly sounds on her Macintosh.
In her free time you can find her working out at the gym and training her flock of pure—bred racing pigeons.
Give Amanda a call and ask her about current pricing on Omnis products, or better still, drop her an email and strike up a fair old chat!
amanda_turner@dlagroup.com.au

Web Enabler (cont.)

Through the use of File Transport Protocol commands, OMNIS Web Enabler allows OMNIS applications to send, receive, and perform remote file operations.

Mail Services
The OMNIS Web Enabler Mail extensions allow an OMNIS application to send and receive E-mail through the use of SMTP and POP3 protocols.

- POP3 — Post Office Protocol level 3 interrogates a user's mailbox, advises on message status, and retrieves mail.

- SMTP — Simple Mail Transport Protocol provides the ability to post messages to a mail server.

Web Servers
The OMNIS Web Enabler Hypertext Transport Protocol (HTTP) extensions allow OMNIS to serve as either a web client or a web data server. HTTP Client support allows OMNIS applications on the desktop to access sites on the Web, gather data, and utilise that data in the application.

A companion product, the OMNIS Web Data Server allows OMNIS applications to work in tandem with standard web servers, providing rich application logic and data manipulation capabilities. Universal clients — NetScape and Microsoft Explorer — can utilise the OMNIS Web Data Server to retrieve and process data from numerous databases.

Utility
Common Gateway Interface, UU Encoding, and Error Handling are OMNIS Web Enabler's three utility extended commands that simplify coding Internet applications.

- Common Gateway Interface — allows an application to encode or decode data from HTML forms. The fields from CGI forms are decoded and immediately available for use within OMNIS.

- UU Encoding — a format that ensures content integrity as data routes through various internet services.

- Error Handling — provides the specification for an OMNIS procedure to be called when an error occurs in HTTP, E-mail, and Sockets extended commands.

Extended Features
OMNIS Web Enabler includes two features that allows designers to develop applications in less time with less effort. These features are Common Code and SourceR.

Common Code
A short cut that reduces the work involved in writing applications. Maintenance of this method is extremely easy because the application code and the Common Code interface do not change. As a result, if the internal code should change, the application would need no modification.

SourcereR
SourcereR makes coding an application faster and more efficient, by providing cut-and-paste reusable code segments that can be embedded in exsiting application.

New look DLA Web Site

Visit the new look DLA web pages located at:
http://www.dlagroup.com.au

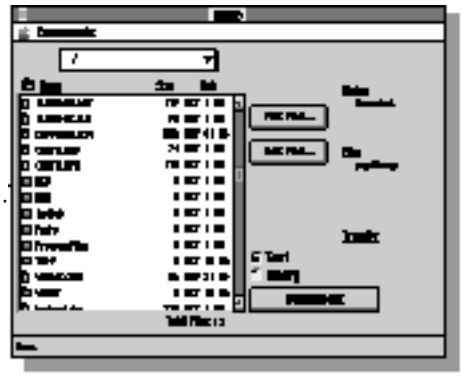
You'll find the latest news about Omnis and other products offered by DLA.
And don't forget about the new ftp site at:
ftp://ftp.dlagroup.com.au

All you need to bring is Netscape or Internet Explorer versions 2 or higher.

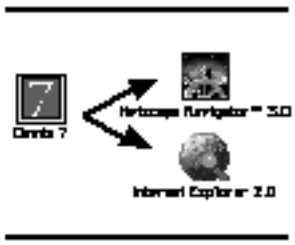
OMNIS® Web Enabler

This exciting new tool is available to Australasian OMNIS developers for a very short time at A$499.
This price is significantly under the US RRP. Send all orders to Amanda Turner <amanda_turner@dlagroup.com.au> or fax her on +612-9262-2290.

ONLY A$499



• *Macintosh FTP application with files*



• *OMNIS can perform as a Web Server*

• *Windows screen shot of Email application*

Name:_____

Company:_____

Tel:_____ Fax:_____

I enclose Cheque/Money order for:_____

Please Specify: Mac ❒ or Windows ❒

Credit Cards cannot be accepted on this special deal.
Offer ends December 31, 1996.

E-Mail:_____

Please return completed form to DLA Technology,
Level 3, 35 Clarence Street, Sydney NSW 2000 Tel: (+61 2) 9262-2255 Fax: (+61 2) 9262-2290

OMNIS 7³ Training

OMNIS 7v3 Essentials

Teaches OMNIS Developers the foundations of application development and the procedural language of this 4GL Development Environment. Upon completion of OMNIS Essentials, developers will be able to create OMNIS 7 applications consisting of windows, menus, reports, procedures and data structures.

OMNIS Essentials Course Topics

• Client/Server and OMNIS Environments • Relational Database Management Systems • Windows design and creation • Menu design and creation • Report design and creation • Sophisticated procedures and parameters • The Debugger • Embedded SQL in OMNIS • Working with Parent/Child Relationships • Lists of kinds (choice, storage, file etc) • Toolgroups • Changing Properties of Objects (Notation) • Introduction to Graphing • Introduction to Object Oriented Development • Validating Data and Controlling Access • Working with a re-useable object library

OMNIS®

OMNIS Essentials Course

March 17-21, 1997

Cost: A$1,950

(10% discount Level 1 Support, 20% discount Level 2 Support.)

OMNIS Essentials course prerequisites

- **Familiarity with database processing concepts**
- **Solid working experience with Macintosh and/or Windows**
- **3.x.3GL and/or 4GL programming experience**
- **Completion of the OMNIS 7³ tutorial**

Cancellations after March 7 will receive a 50% refund. Substitutions are permitted, or students may transfer to another course.

The OMNIS Essentials course includes 5 days of hands on instructions. Course materials, catered lunch, your own Mac or PC workstation for the class duration, and a certificate of completion are provided. Course materials will be despatched a week before all courses to allow students to familiarise themselves with the issues covered.

Name:_____

Company:_____

Tel:_____ Fax:_____

I enclose Cheque/Money order for:_____

Please Specify: Mac ❒ or Windows ❒

Or please charge my:

Bankcard❒ Visa❒ Mastercard❒ Amex❒

Card No:_____

Expiry Date: / /

Cardholders Signature:_____

Please return completed form to DLA Technology,
Level 3, 35 Clarence Street, Sydney NSW 2000 Tel: (+61 2) 9262-2255 Fax: (+61 2) 9262-2290

Notation: The Concepts

Leon Venter

Leon Venter, the "demigod" of notation gave a great talk about notation at the Omnis Underground Conference in Denver. Here is a summary of the talk.

What does notation do for me?

- Provides a programming interface to all the features of the Omnis design environment
- Provides the ability to modify the appearance or behavior of an application on the fly (3GL-like control using a 4GL)
- Provides more powerful and more convenient versions of existing commands
- Provides access to new, useful interface features
- Makes for more maintainable code

What does it look like and where do I use it?

- It's a normal expression, i.e. it can be used wherever expressions are allowed (read)
- It can be used as the destination field in a Calculate statement (write)
- It can be used in square-bracket notation to refer to library, format, datafile names, etc.

The important functions:

$makelist()
  Calculate MyList as $winds.$makelist($ref.$name,...)
  Set current list MyList
  Redefine list
$appendist()
  Set current list MyList
  Define list
  Calculate #F as $winds.$appendlist(MyList, $ref.$name,...)
$findname()
  If myref1.$findname('TheObjName')
  Set ref myref2 as myrefl.$findident(TheObjldent)
$assign()
  Calculate #F as $cwind.$title.$assign('New Window Title')

Some more examples

- Use $visible to show/hide fields
- Use $enabled to test if field is
- Use listfld.$line and listfld.$linecount to avoid 'Set current list'
- Using custom cursors over fields
- Getting the current selection in a text field (use of refs)
- Setting/clearing colours/styles for cells in a table
- Controlled resizing of windows

Common mistakes

- Missing $
- Misspelled items and attributes
- Missing parts of the tree
- $name/$desc/$fullname
- Names starting with numbers or containing periods - the rules for using / /
- Referring to $winds/$cwind when the window isn't open
- $windows.$head vs $winds
- Calculate ref as xxx.$ref
- Errors in the documentation

Debugging techniques & hints

- Use the Attribute Inspector to assemble your notation strings
- Verify correctness starting from the left
- Use $assign and check the flag
- Use an error handler

  Load error handler STARTUP/499 (kerrBadNotation)
  Local variable begin (Short integer (0 to 255)) =
    pos(chr(lO),#ERRTEXT)+1
  Local variable len (Short integer (O to 255))=
    pos(chr(10),mid(#ERRTEXT,begin,len(#ERRTEXT)))-1
  OK message (High position,Large size,Sound bell) {Bad
    notation:'[mid(#ERRTEXT,begin,len)]'}
  SEA continue execution

Notation Sample Code

Leon Venter

Setting a button's text:

```
Calculate  $root.$libs.MyLib.$windows.MyWindow.$objs.MyButton.$text
as  'NewText'
```

Hide/Show field:

```
Hide fields 5 to 6
```

VS

```
Calculate $cwind.$objs.FirstName.$visible as kFalse
Calculate $cwind.$objs.LastName.$visible as kFalse
```

Checking for field enabled:

```
Test for field enabled 1
If flag false
```

OR

```
If $winds.MyWindow.$objs.
 MyFieldObjName.$enabled = kFalse
```

Current list line and number of lines:

```
Set current list ListField
Calculate CurrLine as #L
Calculate NumLines as #LN
```

OR

```
Calculate CurrLine as
 lst(ListField, 0, #L)
Calculate NumLines as
 lst(ListField, 0, #LN)
```

OR

```
Calculate CurrLine as
 ListField.$line
Calculate NumLines as
 ListField.$linecount
```

Changing the cursor:

STARTUP or Window Proc 0:

```
Calculate $clib.$prefs.
 $mouseevents as kTrue
```

Picture field:

```
If #MOUSEENTER
 Calculate $modes.$fixedcursor as
   kTrue
 Calculate $modes.$ccursor as
   kcursNogo
Else If #MOUSELEAVE
 Calculate $modes.$fixedcursor as
 kFalse
End If
```

Getting selected text:

Window Proc 0:

```
; Set a reference to the text field on
the open window belonging to this proc
Set reference fRef to $cwind.$objs.['fText']
```

Add Keyword button:

```
If #CLICK
     ; Get the number of chars in the
     current selection in the text field
     Calculate %lendisabled
     If %len ;; If something was selected
      ; Get selected characters
      ($firstsel = position of the
          first selected char -1)
      Calculate fKeyword as
          mid(fText,fRef.
          $firstsel+1,%1en)
      Set current list fKeyword_List
      Add line to list ;; Append it to
      the list

      ; Set #L to the new keyword so that
       it will be visible when the list is
       redrawn
       Calculate #L as #LN
       Redraw lists  (Selection only)
     End If
End If
```

Colored lines in a table:

Colorise button:

```
If #CLICK
     Set current list fAcct_List
     For each line in list from 1 to
      #LN step 1
     If lst(fAcct_Balance)<0
          Calc $cwind.$objs.['fAccts'].
           $tabrobjs('fAcct_No',#L).
           $textcolor as kRed
          Calc $cwind.$objs.['fAccts'~.
           $tabrobjs('fAcct_Bal',#L).
           $textcolor as kRed
          Calc $cwind.$objs.['fAccts'].
           $tabrobjs('fAcct_Name',#L).
           $textcolor as kRed
     End If
     End For
End If
```

Reset button:

```
If #CLICK
     ; Clear all exceptions
     Calculate #F as
      $cwind.$objs.['fAccts'].
      $clearexceptions
End If
```

Controlled Resizing:

Window control procedure:

```
If #RESIZE
     Calculate $cwind.$width as
      $cformat.$head.$width ;Fixed width
     Calculate $cwind.$height as
      max(80,  $cwind.$height)
End IF
```

Saving window positions:

Window control procedure:

```
If #CLOSE
     ; Copy open window's left and top
     values to the window's format
     Calculate $cformat.$head.$top
      as $cwind.$top
     Calculate $cformat.$head.$1eft
      as $cwind.$1eft
     Calculate $cformat.$head.$width
      as $cwind.$width ;;Must do this!
     Calculate $cformat.$head.$height
      as $cwind.$height
     Save format  {[$cwind().$name]}
End If
```

Databases and database versions supported by Prometheus DAMs

David McKay, Director, Product Marketing, Blyth Software Inc.

There seems to be some confusion as to what databases and database versions are supported by Prometheus DAMs. Until now I have not been able to communicate with you exactly what Prometheus would be supporting.

As with a number of other areas of the product, we looked at Prometheus as the point in time where we would begin removing support for older or less popular databases and database versions. For some of you, this may mean that Prometheus doesn't support your system. For example, support for older versions of Sybase, MS SQL Server and Oracle is removed from Prometheus 1.0. If you are running an older version that is not on the accompanying matrix, its not a fault, its just no longer supported.

The following are the testing suites for Prometheus DAMs. Other configurations or future database or client software releases may well work, but these are the supported configurations for Prometheus 1.0 direct connect DAMs:

DAM NAME DATABASE	PLATFORM	CLIENT SOFTWARE
dSYBASE (db_lib) Sybase System 10	Win 3.1	Open Client 11.0.1
	WinNT	Open Client 11.0.1
	Win95	Open Client 11.0.1
	Mac	Open Client 10.0.3
	PowerMac	Open Client 10.0.3
dSYBASE (db_lib) Sybase System 11	Win 3.1	Open Client 11.0.1
	WinNT	Open Client 11.0.1
	Win95	Open Client 11.0.1
	Mac	Open Client 10.0.3
	PowerMac	Open Client 10.0.3
dSQLSERV (ct_lib) MS SQL Server 6.5	Win 3.1	NetLibrary 6
	WinNT	NetLibrary 6
	Win95	NetLibrary 6
	Mac	* To be determined
	PowerMac	* To be determined
dSQLSERV (ct_lib) Sybase System 10	Win 3.1	Open Client 11.0.1
	WinNT	Open Client 11.0.1
	Win95	Open Client 11.0.1
	Mac	Open Client 10.0.3
	PowerMac	Open Client 10.0.3
dSQLSERV (ct_lib) Sybase System 11	Win 3.1	Open Client 11.0.1
	WinNT	Open Client 11.0.1
	Win95	Open Client 11.0.1
	Mac	Open Client 10.0.3
	PowerMac	Open Client 10.0.3

DAM NAME DATABASE	PLATFORM	CLIENT SOFTWARE
dORACLE Personal Oracle 7	Win95	SQL*Net 2.3.2.1.0
	PowerMac	SQL*Net 2.02
dORACLE Oracle 7.013	Win3.1	SQL*Net 2.3.2.1.3A
	WinNT	SQL*Net 2.3.2.1.0
	Win95	SQL*Net 2.3.2.1.0
	Mac	SQL*Net 2.0.1.5
	PowerMac	SQL*Net 2.0.2
dORACLE Oracle 7.016	Win3.1	SQL*Net 2.3.2.1.3A
	WinNT	SQL*Net 2.3.2.1.0
	Win95	SQL*Net 2.3.2.1.0
	Mac	SQL*Net 2.0.1.5
	PowerMac	SQL*Net 2.0.2
dORACLE Oracle 7.2	Win3.1	SQL*Net 2.3.2.1.3A
	WinNT	SQL*Net 2.3.2.1.0
	Win95	SQL*Net 2.3.2.1.0
	Mac	SQL*Net 2.0.1.5
	PowerMac	SQL*Net 2.0.2
dINFORMX Informix 7.1 Online	Win3.1	Informix*Net 5.01
	WinNT	Informix*Net 7.01
	Win95	Informix*Net 7.01
	Mac	Informix*Net 5.02
	PowerMac	Informix*Net 5.02

Assign Proctext Containing Local Variables

This tip is from Bjorn Kjolseth, AlphaBit, as a workaround for the problem of assigning $proctext to a procedure which contains local / forma**CodeCode**t variables.

1. Set up a reference to the procedure
   Set reference to REF as $menus.STARTUP.$procs.10

2. Get a copy of the original procedure
   Calculate #S1 as REF.$proctext

3. Modify procedure in #S1 the way you want, add or change local variables, whatever...

4. Calculate a string with the following temporary procedure
   Calculate #S2 as "Parameter NEW_PROCTEXT (Character 1000000)"
   Calculate #S2 as con(#S2,chr(13),"Calculate $cproc.$proctext as NEW_PROCTEXT")

5. Assign this temporary procedure to the original $proctext (Original $proctext will be deleted)
   Calculate #F as REF.$proctext.$assign(#S2)

6. Call the temporary procedure with the original procedure as a parameter, and the temporary procedure will replace itself with the new procedure !!!! (YIPPEE !!!)
   Call procedure STARTUP/10 (#S1)

Adding Using Notation

- Format Variables

If the variable "lvVar" resolves to the string "fv_Name" then

Calculate #F as
$cformat.$fvardefs.$add(lvVar,kCharacter,kSimpleChar,1000)

will add the format variable "fv_Name" to the current format.

- Lines to a search format

Calculate #F as
$clib.$searches.S_NAME.$objs.$add(kSLbeg)
  ;; first add the object and then all attributes extra
Calculate #F as
$clib.$searches.S_NAME.$objs.1.$linetype.$assign(kSLbeg)
Calculate #F as
$clib.$searches.S_NAME.$objs.1.$fieldname.$assign('F_NAME')
Calculate #F as
$clib.$searches.S_NAME.$objs.1.$text.$assign('value')
Add lines to a search format

Set ref s_ref to $clib.$searches.S_Name.$objs
Calculate #F as s_ref.$add('',F_Name) ;;
F_Name = field name
Calculate #F as s_ref.[xx].$linetype.$assign(kSLeq)
Calculate #F as s_ref.[xx].$text.$assign("PEACE") ;; where xx is the line just added to the search format
  NOTE: There is a bug in Omnis where it fails to set the $linetype and $text of the search format in the $add() part - that's why you need the other two lines.

- A column to a List

Calculate %rc as #L8.$cols.$add('NewFieldNane')
Remember: when you add columns to an existing list, the list is cleared.

- Background objects to windows

If you add a bobj (like a line) to an open window, the parameters for the $add() gives you control for the positioning and size of the bobj, but if you want to set more additional attributes like color etc, you need the ident number of the bobj item or a reference to it. Omnis 7 gives all new bobj items a unique number; but how can you get this number after adding an item to the bobj's group. $cobj is not set to the last added item. When you add an item with the $add function it should be done via a set reference command. The reference returned is to the item added. You can then use it to reference anything about the item you need.

Example-1
Format variable MY_BOBJ (Item Reference)
Set reference MY_BOBJ to
$winds.WINDOWNAME.$bobj.$add(kLine,top,left,height,width)
Calculate MY_BOBJ.$color as puke_green
Calculate MY_BOBJ.$linestyle as dotty

Example-2
Set reference MY_BOBJ to
$winds.$bobj.$add(kLine,top,left,height,width)
Calculate #1 as MY_BOBJ.$ident ;; to get the ident of the item just added.

If you use the $add in a calculate it returns a flag indicating that the command executed or failed.

For example:
Calculate #1 as
$winds.WINDOWNAME.$bobjs.$add(kRect,top,left,height,width)

Compiled by Stephen Miller
<stephen_miller@dlagroup.com.au> from the Notation Tips Library available on DLA's ftp site.

Jumpstart your learning curve with Prometheus, Web Dev and Notation training videotapes

What is your time worth? Do you have days to spend learning how to get started with Blyth's new object oriented technologies and Internet enablement tools? Wouldn't you prefer to learn from the best OMNIS talent in the world — at your convenience? The OMNIS Underground has secretly acquired these highly classified training tapes. These tapes have been covertly filmed at the Prometheus, Web Dev and Notation sessions during the 1996 Omnis Underground Technical Conference, in Denver Colorado this November. The tapes feature the talents of David McKay, Director of Product Marketing for Blyth Software; Bill Hamilton of WMPNJ Software; Chris Clarke, Internet Evangelist for Blyth Software; and Omnis legend Leon Venter, of Blyth Software.

The Prometheus series features approximately 12 hours of training, covering OO concepts, basic Prometheus tools, and use of the new IDE (integrated development environment). Bill Hamilton focuses on conversion and issues related to porting 7.3.x to Prometheus. (A$250)

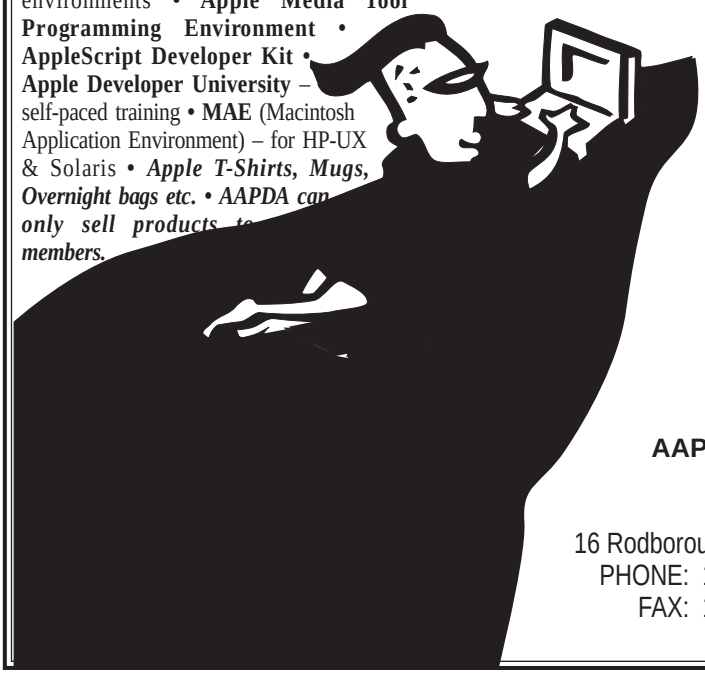
Chris Clarke provides nearly 6 hours of coverage on **Blyth's new Inter/Intranet enabling tools.** These new tools will empower the Omnis developer to access and be accessed by the World Wide Web. Web Dev enables an Omnis applictaion with TCP/IP protocols, opening a new world to developers. (A$200)

Finally, we offer **6 hours with the master of Notation**, Leon Venter. Leon covers everything from the basic Notation to advanced tips and tricks to the most closely guarded secrets of a wizard of Omnis DML. (A$200)

Order your copies from Amanda Turner at DLA. email <amanda_turner@dlagroup.com.au>

Apple Development Tools delivered right to your door!

• **QuickTime VR, & Developers Kit** – for Mac & Windows • ***develop***, the programmers magazine with CD • **HYPERCARD Only $156** • **SuperCard** • **CodeWarrior Gold** & **Platinum** – Mac, Power Mac& Windows cross-platform development systems • **Newton Toolkit & Newton Programming books** • **Inside Macintosh** books & CD-ROM • **MacOS Software Development Kit** • **Apple Media Tool** • **QuickTime 3D, Quicktime Conference** – coming soon • **E·T·O***Essential Tools & Objects* & **MPW Pro** *Macintosh Programmers Workshop* programming environments • **Apple Media Tool Programming Environment** • **AppleScript Developer Kit** • **Apple Developer University** – self-paced training • **MAE** (Macintosh Application Environment) – for HP-UX & Solaris • ***Apple T-Shirts, Mugs, Overnight bags etc.*** • ***AAPDA can only sell products to members.***



Every AAPDA member receives: • The quarterly "Developer Catalog" with all of Apple's latest software development tools AAPDA News • Members quarterly newsletter • Special pricing on RAM & HD upgrades • Access to the AAPDA bulletin-board service which is online to the internet • Developer University course notes for hire • Technical advice on AAPDA products before purchasing & technical support after purchasing • Invitations to Special Interest Group Meetings & Apple Professional Breakfasts • 1800 Sales & Fax lines • Nil freight charges for books purchased & much more!

Interested in becoming a member? AAPDA is the sole Australian distributor of Apple Computer Inc. Developer products & only members can purchase. Call us for a membership application form & the latest newsletter on 1800 642 732 or send us an email.

AAPDA yearly membership. $45 including a stylish six-colour designer t-shirt featuring the AAPDA wave-rider shown here. Only $30 without the designer t-shirt! Membership is payable yearly.

AAPDA is a non-profit incorporated association dedicated to the continuing provision and improvement of goods & services for its members.

AAPDA Members include Programmers, Developers & Consultants.

16 Rodborough Road Frenchs Forest NSW 2086
PHONE: 1 800 642 732 (int +61 2 9452 8245)
FAX: 1 800 674 615 (int +61 2 9975 6397)
EMAIL: cgargan@aapda.org.au

aapda THE USER GROUP FOR PROGRAMMERS

Generic Code Teaser

by Stephen Miller

The purpose of the following code is to display some backbone code for checking each and every file format containing records in the selected datafile for index accuracy. Like a lot of generic coding, exception handling is 90% of the work. Email me, stephen_miller@dlagroup.com.au, with your suggestions as to why this code will not work on every native datafile application.

```
Calculate $ignoreExternal as kTrue
Prompt for library (Do not close other
libraries,Do not call startup procedure)
{object}
If flag false
     Quit procedure
End If
Set current list fvlstFORMAT_LIST
Define list {fvchrNAME,fvboolDROP,
     FVNUM_RECORDS}
Calculate fvlstFORMAT_LIST as
     $root.$libs.object.$files.$makelist
     ($ref.$name,'','')
Redefine list {fvchrNAME,fvboolDROP,
     FVNUM_RECORDS}
Prompt for data file {objectdata}
If flag false
     Quit procedure
End If
Set current list fvlstFORMAT_LIST
Calculate #L as 1
Repeat
     Load from list
     Calculate %%MODE as $root.$libs.
     object.$files.[fvchrNAME].$head.
     $filemode
     If %%MODE='readwrite'|%%MODE=
     'readonly'
          Set main file {[con('object.',
           fvchrNAME)]}
          Calculate FVNUM_RECORDS as
           sys(83)
          Replace line in list
     End If
     Calculate #L as #L+1
Until #L>#LN
Set current list fvlstFORMAT_LIST
;  delete empty file formats
Calculate #L as 1
Repeat
     Load from list
     If FVNUM_RECORDS=0
          Delete line in list
          Calculate #L as #L-1
     End If
     Calculate #L as #L+1
Until #L>#LN
;  build list of fields for each format
Set current list fvlstFORMAT_LIST
Calculate #L as 1
Repeat
     Load from list
     Set current list fvlstFIELDS
     Define list {FVCHRFIELDNAME}
     Calculate fvlstFIELDS as

$root.$libs.object.$files.[fvchrNAME]
     .$head.$indexes.$makelist
     ($ref().$name)
     Redefine list {FVCHRFIELDNAME}
     If #LN
          Calculate #L as 1
          Repeat
               Load from list
               Set main file {[con
                ('object.',fvchrNAME)]}
               Calculate %COUNTER as 0
               Find first on [con
                ('object.',fvchrNAME,'.',
                FVCHRFIELDNAME)]
               If flag true
                    Repeat
                         Calculate
                          %COUNTER as
                          %COUNTER+1
                         Next
                    Until flag false
                    If FVNUM_RECORDS<>
                          %COUNTER
                         Calculate
                          fvboolDROP as 1
          ;; sets the flag to re-index
                    End If
               Else
                    Calculate
                     fvboolDROP as 1
               End If
               Calculate #L as #L+1
          Until #L>#LN
     End If
     Set current list fvlstFORMAT_LIST
     Replace line in list
     Calculate #L as #L+1
Until #L>#LN
Set current list fvlstFORMAT_LIST
Calculate #L as 1
Repeat
     Load from list
     If fvboolDROP
          Set main file {[con('object.',
           fvchrNAME)]}
          Set read/write files {[con
           ('object.',fvchrNAME)]}
          Drop indexes {[con('object.',
           fvchrNAME)]}
          Build indexes {[con('object.',
           fvchrNAME)]}
          Set read-only files {[con
           ('object.',fvchrNAME)]}
     End If
     Calculate #L as #L+1
End If
Until #L>#LN
```

TechTip TT086
Title: Resource Editing and OMNIS 7.3.5.x for Windows 95
Category: Extending OMNIS
Product Version: OMNIS 7.3.5.2
TechTip Version: 1.0

Fault N#188#: PR/ET/003FC

Question: How can I edit the resources of OMNIS 7.3.5.x for Windows 95?

Answer: Since the release of OMNIS 7.3.5.x for Windows 95, three questions related to the editing of OMNIS resources have been asked on a regular basis. The intent of this tech tip is to answer these questions in as complete a manner as possible, using a standard FAQ format.

1) I am able to use App Studio which shipped with VC++ v1.5x for Win3.1 to edit the resources contained in the OMNIS7.EXE and OMNISRES.DLL files of OMNIS 7.3.x.x for Win3.1. Why is it that Developer Studio from VC++ v4.x running on Win95 will not allow me to edit the resources contained in the versions of these files supplied with OMNIS 7.3.5.x for Win95?

The answer to this question is given clearly in the response section of the following VC++ FAQ (originally listed in the online documentation that ships with VC++ v4.2):

Q: Why can't I edit EXE resources running Developer Studio under Windows 95?

A: Developer Studio now supports the direct editing of resources within an EXE file, but only when running under Windows NT. Windows NT provides APIs which allow you to modify resources in an executable(.EXE) or dynamic link library (.DLL). Windows 95 does not support the APIs necessary to do this. For example, it does not support BeginUpdateResource( ), UpdateResource( ) and EditUpdateResource( ).

When a resource is loaded under Windows 95, Developer Studio will issue the following warning:

Developer Studio cannot save the modified resources back to this executable. The executable may be in use, or this version of Windows may not support updated resources in executables.

While the executable cannot be modified in Windows 95, the resources can be copied into an .RC file.

---

2) How can I edit the resources contained in the Win95 version of OMNIS7.EXE without using VC++ v4.x running on Windows NT?

At the time of writing, due to the vendor limitation cited above, there is no means possible to edit the resources contained in a compiled copy of the Windows 95 version of the OMNIS executable, using tools running on Windows 95.

Even though this is not an OMNIS limitation, a fault, PR/ET/003FC, has been logged in case Blyth R&D might be able to provide a means to overcome this limitation at some future date.

3) How can I edit the resources contained in the Win95 version of OMNISRES.DLL without using VC++ v4.x running on Windows NT?

The implication of the last sentence from the VC++ FAQ cited above is that it might be possible to create a modified version of OMNISRES.DLL using VC++ running on Win95 by the following means, unfortunately this approach does not work:

Copy the resources from the original version of OMNISRES.DLL into the resource script of a suitably defined new OMNISRES.DLL project in Developer Studio; modify the resources while they are in the resource script; and then build the new version of the OMNISRES.DLL.

Although it does allow a new version of OMNISRES.DLL to be built with modified resources, loading the new DLL will cause serious conflicts with OMNIS.

However, Blyth does currently support a means of creating a customised version of OMNISRES.DLL by working with its component resources. The file RESOURCE.ZIP, located in the Win95\DEPLOY directory of the OMNIS 7.3.5 installation CD, contains most, but not all, of the individual resource files and scripts necessary to recompile the resources of the 32-bit version of OMNIS from the command line; using the resource compiler, RC. EXE, and the linker, LINK.EXE, supplied with VC++ v4.x. The file MISSING.ZIP (which is currently available on request from US Support) contains the remaining 5 files required for this purpose which were unintentionally omitted from the shipped version of RESOURCE.ZIP.

Use the following steps to re-compile the resources with any desired modifications:

1. Create a directory for your new omnisres project.

2. Copy RESOURCE.ZIP and MISSING.ZIP to the new directory, and run PKUNZIP to extract their contents.

3. a) Using Developer Studio you may edit the individual resource files (.ICO, .BMP, .CUR, etc.) contained in the new directory just as you would for a normal VC++ project.
b) If you wish to make changes to the resource scripts contained in the new directory, be sure to work directly with the text of the .RC files; i.e., when opening these files in Developer Studio, be sure to open them in text-mode by selecting 'Text' under the 'Open As:' option in the Open File dialog box. This is necessary in order to prevent VC++ from unsuccessfully attempting to do an automatic compilation of these files as soon as they have been opened.

4. Once any modifications have been completed, shell out to the DOS prompt and run VCVARS32.BAT in order to set the environment variables required for compiling and linking from the command line (VCVARS32.BAT should have been created in the \BIN directory subordinate to the root directory of VC++ by the VC++ installer).

5. Change directory to the directory created in step 1) above and run OMNISRC.BAT (OMNISRC.BAT invokes rc and link). If omnisrc runs successfully*, a new, usable copy of OMNISRES.DLL will have been created by this process.
*Note: Sometimes running VCVARS32.BAT will use up all the available environment space before it has been able to set up the required environment variables for running rc and link from the command line. If any difficulty is encountered in running OMNISRC. BAT, it will be necessary to create a custom bat file to replace VCVARS32.BAT. This new bat file will set up only the variables essential to successfully running OMNISRC.BAT.

6. After first making sure that no instances of OMNIS are currently running, replace the copy of OMNISRES.DLL currently in the root directory of your OMNIS installation with the copy that was built in the previous step.

7. Start OMNIS and make sure that your current OMNIS environment reflects the modifications that you have made to OMNISRES.DLL.

8. Repeat steps 3-7 as required to produce a copy of OMNISRES.DLL that meets your specifications.

==========

In order to create this custom bat file take the following steps:

1) Shell out to the DOS prompt from Win95.
2) Run the DOS 'SET" command at the prompt, redirecting its output to file in order to capture your current environment settings.
3) Create a bat file named, for example, CUSTVARS.BAT and copy your original settings to this file.
4) At this point the contents of CUSTVARS.BAT should be similar to the following example:

```
TMP=C:\WINDOWS\TEMP
winbootdir=C:\WINDOWS
COMSPEC=C:\WINDOWS\COMMAND.COM
PROMPT=$p$g
PATH=C:\WINDOWS;C:\WINDOWS\COMMAND;C:\DOS
TEMP=C:\DOS
```

5) Using a plain text editor, modify CUSTVARS.BAT by making only the exact changes required to match the example above with the example below.

```
SET TMP=C:\WINDOWS\TEMP
SET COMSPEC=C:\WINDOWS\COMMAND.COM
SET PROMPT=$p$g
SET PATH=C:\WINDOWS;C:\WINDOWS\COMMAND;
     C:\DOS;C:\MSDEV\BIN;
     C:\MSDEV\BIN\WIN95;
SET TEMP=C:\DOS
SET MSDEVDIR=C:\MSDEV
SET INCLUDE=C:\MSDEV\INCLUDE;C:\;
SET LIB=C:\MSDEV\LIB;
```

Note that if your VC++ installation were not on your C: drive but was, for instance, on drive D: this example would appear as follows:

```
SET TMP=C:\WINDOWS\TEMP
SET winbootdir=C:\WINDOWS
SET COMSPEC=C:\WINDOWS\COMMAND.COM
SET PROMPT=$p$g
SET PATH=C:\WINDOWS;C:\WINDOWS\
     COMMAND;C:\DOS;D:\MSDEV\BIN;
     D:\MSDEV\BIN\WIN95;
SET TEMP=C:\DOS
SET MSDEVDIR=D:\MSDEV
SET INCLUDE=D:\MSDEV\INCLUDE;D:\
SET LIB=D:\MSDEV\LIB;
```

6) After saving these changes, CUSTVARS.BAT will be ready for use.

The changes that CUSTVARS.BAT makes to your environment settings will only remain in effect for the duration of the DOS session that it was invoked in. For this reason, be sure to run CUSTVARS.BAT before running OMNISRC.BAT each time that you have shelled out to DOS to build OMNISRES.DLL.

TechTip TT087

Title: Porting OMNIS Externals from Windows 3.1 to Windows 95 and Windows NT
Category: Extending OMNIS
Product Version: OMNIS 7.3.5.2
TechTip Version: 1.0

Fault N#188#: NONE

Question: I have a custom OMNIS external that was written and compiled for use with OMNIS 7.3.x for Windows 3.1. What should I be aware of if I wish to use this external with OMNIS 7.3.5.x for Windows 95 or Windows NT?

Answer: Contrary to the OMNIS 7.3.5 Installation & Update manual and the 3.5 and 3.5.1 readme files, it is not safe to assume that all 16-bit OMNIS externals written for Windows 3.1 will be loaded by OMNIS 7.3.5.x for Windows 95 or Windows NT, or that they will work correctly if they are loaded. This limitation is primarily due to the differences which exist between the APIs, memory management schemes, and DLL management schemes of the16- and 32-bit Windows platforms. Therefore, if you experience problems with your 16-bit externals on Windows 95 or wish to port them to Windows NT, your 16- bit source code must be modified to be Win32-compliant and then recompiled into 32-bit DLLs.

The notes which follow are intended to provide the various pieces of information required to complete this process successfully. These notes assume that the reader is familiar with the material in the Extending OMNIS manual, that they are already capable of programming successfully in C/C++, and that they are developing their 32-bit externals using Microsoft Visual C++ v4.x.

* Provided that all of the information given or referred to in the notes below is read and applied carefully, the default settings that VC++ v4.x provides for a standard Win32 (Not MFC!) DLL project should work perfectly well.

* The first and most important thing to do before beginning with the process of creating a Win32-compatible external is to read the section of the OMNIS 7.3.5 Installation & Update manual titled "Writing Extensions for Windows NT" on pages 105-107. All of the information in this section applies fully to external code for both Windows NT and Windows 95 and should be followed to the letter. The only item in this section which may be ignore is the tip on page 107 that suggests the use of 1-byte as opposed to 8-byte packing for structures.

* A vital source of information on the general porting issues that can affect the success of a 32-bit recompilation is the 'porting book' which appears in the InfoView pane of Developer Studio under 'Key Visual C++ Topics>>Programming Topics'. This book gives a complete rundown of the changes to API calls, #defines, type_defs, calling conventions, and so on between the 16 and 32-bit Windows API's, amongst a lot of other useful information regarding the differences between 16- and 32-bit native Windows code.

* For further explanations of the differences between Win3.1 and Win32 DLLs, good places to start are the relevant chapters in "Advanced Windows" (1995, Microsoft Press) by Jeffrey Richter and "C++ Windows NT Programming" (1996, M & T Books) by Mark Andrews.

* If you are new to working with OMNIS externals, or you are experiencing difficulties of any kind, it is good practice (after having first read the remainder of this tech tip) to carefully examine and make sure that you can successfully compile each set of example source files (except those for the Replace external) supplied on the OMNIS 7.3.5 Installation CD in the WINNT(or WIN95) \OMNIS7\CSRCEXT directory. At the very least, make sure that you can build a working copy of REPLACE.DLL using the files supplied in the WINNT\OMNIS7\CSRCEXT\REPLACE\MSVC4 directory of the OMNIS 7.3.5.2 Patch Release CD.

* If you are writing an external command or function package, you will need to include an entry in the string table of the package's resource script for the entry-point in order for OMNIS to detect the package's entry-point; i.e., if the package's entry-point is YourEntryPoint(), its string table should have the following form:

```
STRINGTABLE
BEGIN
FIRSTSTR_ID, "Cmnd1:xxxxxxxxx"
.
.
.
FIRSTSTR_ID+n, "Cmndn:xxxxxxxxx"
FIRSTSTR_ID+n+1, "*YourEntryPoint" /* Note
that there is no unique ID for the entry-
point but the '*' is required */
END
```

* You must use the most recent versions of the callback source files; i.e., the versions of these files supplied on the OMNIS 7.3.5.1 and OMNIS 7.3.5.2 Patch Release CDs. Once you have the correct versions of these files you will still need to make the following minor modifications:

1) The first line of extypes.h begins with "#/* ...", this is an obvious error, it should begin with "/* ..."

2) In callback.h, it is neccessary to reverse the order of the preprocessor conditional statements that occur on lines 68-71; i.e., the following lines:

```
#elif defined(_WINDOWS) || defined(MSDOS) ||
defined( MSC7) || defined(__BORLANDC__) ||
defined (__TURBOC__)
#define WINOS
#elif defined(__NT__) || defined(_WIN32) ||
defined( msvcnt) || defined(mscnt)
#define WIN32OS
```

should be swapped as follows:

```
#elif defined(__NT__) || defined(_WIN32) ||
defined( msvcnt) || defined(mscnt)
#define WIN32OS
#elif defined(_WINDOWS) || defined(MSDOS) ||
defined( MSC7) || defined(__BORLANDC__) ||
defined (__TURBOC__)
#define WINOS
```

or if you are not concerned with maintaining callback.h as a cross-platform file, simply delete all the conditional OS defines and define WIN32OS.

* If you intend to link your external code to any modules compiled as C++, be absolutely sure that the calls you make into those modules are marked ' extern "C" ' in the relevant C++ source files, as OMNIS 7.3.5.x still expects external code to be C-compliant.

* When you want to edit any .RC file for an external project, be sure to work with it in text mode only; i.e., if you are editing the file inside Developer Studio, do not allow Developer Studio to pass the file to the resource compiler until you are ready to build the project. This is necessary because editing the compiled resources before the project is built will result in Developer Studio rewriting the resource script in an unacceptable manner.
Blyth Tech Support

Internet Connectivity

Is it the true highway or just a headache?

by Christian Kraus

Technology is said to make things easier, simpler, cheaper and better, unfortunately, things are rarely that straight forward especially with computers and communications.

While going online via an Internet Service Provider (ISP) or commercial services such as CompuServe is relatively straight forward and provides almost instant access to popular electronic mail and world wide web (WWW), it can be too awkward for multiple users and for serious business use.

Some time ago I was given a project to implement electronic mail for DLA. The specification soon developed to include Internet access and support for both the Mac and Windows platforms. We wanted a solution with the most functionality yet while keeping costs to a minimum.

This paper discusses the issues considered and selection of the most appropriate and cost effective solution.

One of our aims was to retain complete control over our electronic mail setup whilst avoiding the hassle of operating our own WWW server.

This was accomplished by dividing up our Internet system into multiple components:

1. Domain
2. FTP & WWW Hosting
3. Internet Access
4. Mail Server
5. Internal Network & Gateway

1. Domain

Since DLA had already registered a domain name there was no reason to apply for a new one. Registering your own domain name is an important step in getting connected as it gives your organisation its own home on the net. Users can then send mail and connect to your host domain name.

When a client connects to a server connected to your domain it does so by first resolving the address through the use of a Domain Name System (DNS).

In the early days almost 16 years ago, host names and their Internet Protocol (IP) addresses were stored in a single file called hosts.txt, maintained by the Network Information Center in the US. However, as the Internet grew so did this file, and by 1985 it had become impractically large to distribute to systems over the network. It was also almost impossible to keep up to date.

The Internet Domain Name System (DNS) was developed as a distributed database to solve this problem. Its primary goal is to allow the allocation of host names to be distributed amongst multiple naming authorities, rather than centralised at a single site.

We created our DNS entry to essentially direct all traffic to our domain based upon the type of incoming traffic. SMTP (Simple Mail Transfer Protocol) was directed to our Mail Server, HTTP (Hyper-Text Transfer Protocol) directed to our WWW server and FTP (File Transfer Protocol) to our FTP Server.

Because we had setup our DNS entry with delegation in mind we were able to distribute the load on our system to a number of different servers even though they are not necessarily located in the same location. In fact your mail server could be in Australia, your WWW server in the US. What makes the system work is the DNS system and IP addresses that point to a particular connection.

2. FTP & WWW Hosting

Our requirements for the FTP & WWW Components of our solution included reliable high speed connections to the Internet. Whilst we could easily have setup our own server it would mean that we would also have to increase the bandwidth of our connection in order to sustain an acceptable level of

performance. We decided to outsource our WWW & FTP server by renting space on a server which already had a high speed connection to Internet.
Internet solutions can become very expensive very quickly when considering it is not unusual to pay $15,000+ on a medium speed ISDN connection. At these rates such a connection is hard to justify for small to medium sized companies. Creating the right balance between doing it yourself and outsourcing is important and will ultimately dictate whether or not your solution will be a success.

3. Internet Access
For Internet Access we chose a local Internet Service Providor (ISP). The only requirement here was for a semi permanent connection with a fixed IP address so that our DNS entry is able to point mail to a dedicated server in the Internet. Many ISP's allocate a dynamic IP address upon connection from their pool of available IP addresses.

A fixed IP address is required in order to link our internal network with the Internet as we needed to direct incoming traffic to the server which handles our mail.

4. Mail Server
There are many different mail clients, or mail user agent (MUA) available which allow users to compose, send, receive, and read electronic mail.

Some are proprietary systems while others are based upon Internet standards such as Eudora. We choose Eudora Pro v3 for Mac and Windows as the standard mail software for both PC's and Mac's due to its easy to use yet sophisticated features.

One of the reasons we wanted to avoid a proprietary system was that one is often locked into using their proprietary gateway software. Many email networks aren't based upon TCP/IP but use AppleTalk or other network protocols. While this doesn't usually effect the email system, we preferred to use Eudora as it will operate quite happily on a TCP/IP network and also allows us to provide WWW and FTP connections over the office LAN.

A mail server, or mail transport agent (MTA), handles the delivery of mail to mail clients, and the routing of mail between different servers. As our server was going to be run on a Performa PowerMac we choose Apple Internet Mail Server (AIMS) as our mail server.

5. Internal Network & Gateway
Our office is a mix of Mac's and PC's so any solution would need to be a cross platform one and this is why we chose to implement a TCP/IP network. On the PC's, Microsoft TCP/IP was used and on the Mac's MacTCP (part of system 7).

The VICOM Internet Gateway is a clever software based router/gateway solution which also implements a firewall to ensure security. While a hardware router is normally used in these types of situations, this software solution has definite advantages over the traditional methods as it is cheaper and just as effective as a hardware solution for small to medium sized networks.

The gateway software which is running on the server automatically maps any network traffic from machines using IP addresses that is not addressed to or from a machine within the local subnet.

The gateway also has an alias address which is in fact the same address that was assigned to as as our static IP. The router functionality of the VICOM software allows the forwarding of IP packets between the two IP addresses and hence creating the gateway and allowing internet access from the local area network.

In this situation your whole network is sharing a single real world IP address. Another benefit of the VICOM software is because you can "share" the single static IP as your connection/gateway you don't need to apply (and pay) for additional IP addresses for each machine connected to the gateway as you would if a hardware solution was implemented.

Security is extremely well implemented due to the properties of the gateway. Incoming traffic is permitted to enter the network as long as the destination is the same as the gateway address which we forward directly to our mail server; so you can only go as far as our Gateway/Mail Server. As an additional measure we have only installed TCP/IP as the network layer so there is no file system support or shell accounts (telnet) to login to. Even if someone could break past our server there is nothing else they could do!

Summary
While this type of setup is quite involved, the principles are quite simple. If you have ever thought of utilising email to its limits this is a solution that is worth considering.

The benefits from using a full time Internet/mail connection are considerable. Its cheaper than calling 3 times a day to collect your mail, permits almost instant delivery and can improve your response time to customers, assuming they have email as well of course.

The DLA technical team can assist you with any connections, gateways, servers or even designing of a FTP or Web site to suit your requirements. For further information please do not hesitate to contact me.
Christian Kraus <christian_kraus@dlagroup.com.au>

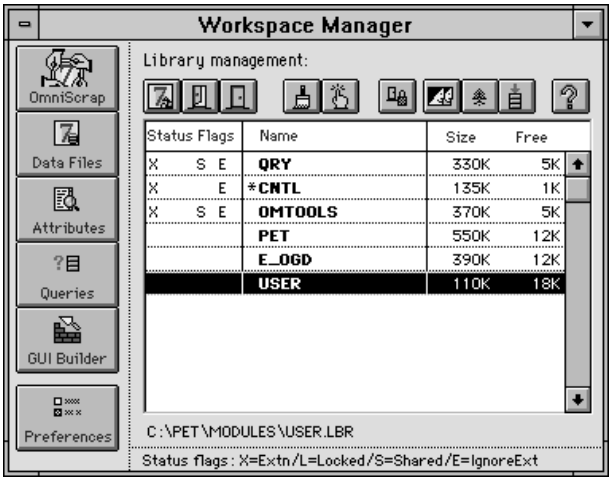
Christian Kraus

OMTOOLS

A Development Shell for OMNIS

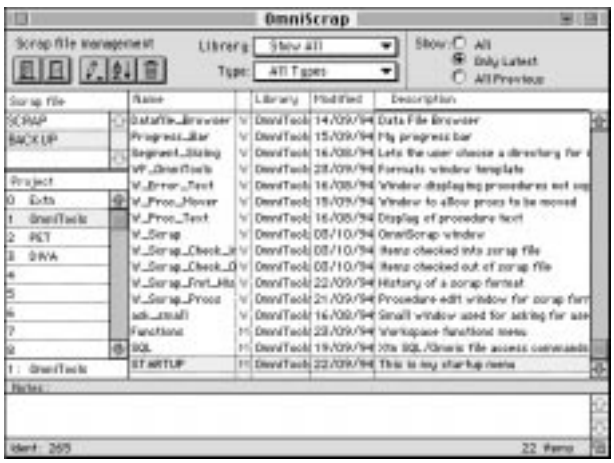
OMTOOLS is a modestly priced development shell that can be used to replace the design shell provided in OMNIS 7v3.x. The purpose of this product is to produce a less complex product directed towards the individual or small work group of developers. The authors of this tool maintain that it operates with considerably improved performance compared to the Blyth supplied counter parts.

The design menu is replaced by a menu with several new options in particular the Work Space Manager. The Work Space Manager is depicted below:



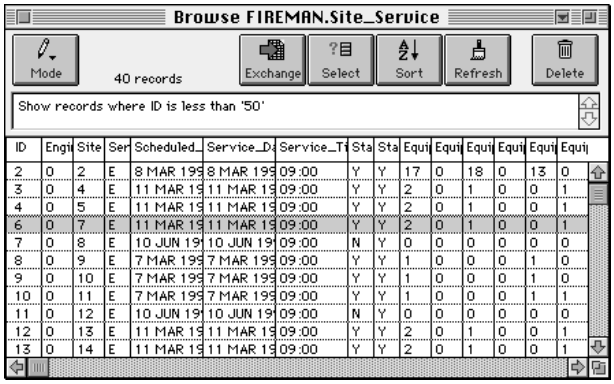
The Blyth supplied attribute inspector and data file browser are still accessible.

This product comes with its own version control system called OmniScrap which relies upon the OMNIS native data file for the storage of the binary versions of OMNIS formats.



OMTOOLS includes a feature called ProcMover which allows you to drag and drop procedures to a new format slot and then update all links to point to the new format slot.

The product also contains a single table browser format which you can then include in your own libraries. This browser allows you, through the use of generic coding, to access any file format in a data file and produce a table of the contents subject to search criteria, chosen by yourself or your user. This feature alone makes the purchase of OMTOOLS worthwhile. (Ed.)



ID	Engi	Site	Ser	Scheduled_	Service_D	Service_T	Sta	Sta	Equi	Equi	Equi	Equi	Equi	Equi
2	0	2	E	8 MAR 199	8 MAR 199	09:00	Y	Y	17	0	18	0	13	0
3	0	4	E	11 MAR 19	11 MAR 19	09:00	Y	Y	2	0	1	0	0	1
4	0	5	E	11 MAR 19	11 MAR 19	09:00	Y	Y	2	0	1	0	0	1
6	0	7	E	11 MAR 19	11 MAR 19	09:00	Y	Y	2	0	1	0	0	1
7	0	8	E	10 JUN 19	10 JUN 19	09:00	N	Y	0	0	0	0	0	0
8	0	9	E	7 MAR 199	7 MAR 199	09:00	Y	Y	1	0	0	0	1	0
9	0	10	E	7 MAR 199	7 MAR 199	09:00	Y	Y	1	0	0	0	1	0
10	0	11	E	7 MAR 199	7 MAR 199	09:00	Y	Y	1	0	0	0	1	1
11	0	12	E	10 JUN 19	10 JUN 19	09:00	N	Y	0	0	0	0	0	0
12	0	13	E	11 MAR 19	11 MAR 19	09:00	Y	Y	2	0	1	0	0	1
13	0	14	E	11 MAR 19	11 MAR 19	09:00	Y	Y	2	0	1	0	0	1

An additional product from Dr Khoo is his Qry Library which includes a multiple table browser and automatic generator of SQL.

Dr Steven Khoo can be contacted in the United Kingdom on CIS 100275,3200 or at artscan@btinternet.com

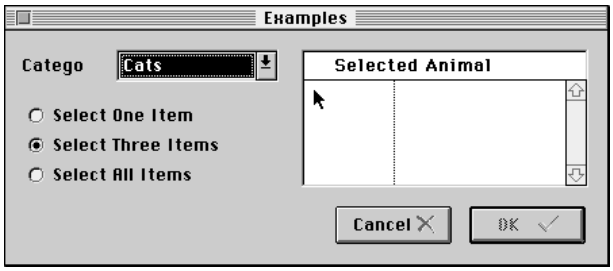
Compiled by: Stephen Miller
stephen_miller@dlagroup.com.au

A FreeWare Data Driven Popup List Model!

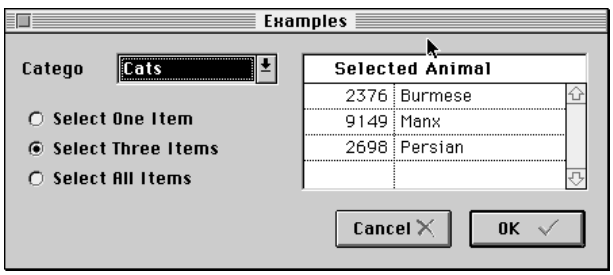
Dinamis.lbr is an external library which contains the wxListView window format. wxListView is a data aware, parameter driven, reusable Omnis object which can be used to display the contents of any arbitrary Omnis list. It is primarily used where it is necessary to provide the user a list of items from which one or more items can be selected. This eliminates the necessity of creating one window for displaying single column lists, another for two column lists, yet another for six column lists, and so on.

Example

From the following Popup Window in response to selecting a list line, the simple poplist or complex popup list window is dynamically created...



Then the results are returned...



Requirements: Omnis 7.3.5

Other Offerings:

The latest offering from Dinamis Corporation is OmniRAD™. Currently, OmniRAD™ is only available as a proprietary tool which is used by our consulting team to enhance the value of our services to our clients. It is an object oriented, metadata driven, Omnis based, rapid application development (RAD) tool. Unlike most RAD tools, OmniRAD™ does not generate the entire application since we do not believe that non—trivial applications can be generated. Instead, the focus of OmniRAD™ is upon solving a problem which is definitely solvable, that of managing dependencies between Omnis objects. If you have ever had to struggle with spaghetti Omnis code, you will find OmniRAD™ to be a boon.

For questions about wxListView, OmniRAD™, or other offerings, contact:

Clifford Ilkay
e—mail: dinamis@interlog.com
Compiled by:
Stephen Miller
stephen_miller@dlagroup.com.au

Christian's Dream Machine...

To provide a platform which permits the development and testing of applications under a variety of Intel operating system platforms without wasting resources on additional hardware.

Hardware Specification :

- Pentium motherboard with 256k secondary pipeline cache, support for up to 6 72pin SIMMS, 4 EIDE devices using LBA mode, EDO Ram capable and supporting up to a 200Mhz Pentium, PS/2 Mouse port, 2 Serial, 1 Parallel port.
- 32Mb EDO Ram
- 2gb EIDE PIO 4 HDD
- Intel EtherExpress Pro/10 PCI NIC
- 2940 PCI Adaptec SCSI Controller
- 6x CD-ROM
- 15" .28 Dot Pitch 1024x768 Non Interlaced Digital Monitor with built in degauss function.
- Medium Tower Case with 110/250VAC switchable power supply - 101 Enhanced keyboard
- Microsoft PS/2 Mouse
- 1gb SCSI Tape Drive

Software Specification:

- Windows 3.11 for Workgroups
- Dos 6.22
- OS/2
- Windows NT 3.51
- Windows NT 4.00
- Windows95
- System Commander
- MacLan with Long Filename support

OMNIS®

OMNIS workgroup edition

Includes 10 User Runtime for Internal Use!

Blyth Software's OMNIS Workgroup Edition is an uniquely designed introductory client/server application development tool to meet the needs of developers building LAN based applications. The OMNIS Workgroup Edition uses the power, scalability and the built in OMNIS SQL Database to make it possible for you to deliver successful client/server applications- the first time!

A$699 Free Delivery anywhere in Australia

System Requirements:

WINDOWS:	386, 486 or Pentium, WIN 3.1, NT & WIN 95, 8MB RAM (16MB NT), 40MB Disk Space and a CD-ROM Drive.
Mac OS:	68030, 68040 or PowerMac, MacOS 6.08 or later, 8MB RAM (16MB for PowerMac), 40MB Disk Space and a CD-ROM Drive.
Network OS:	Novell Netware, Appleshare, LAN Manager, TOPS, 3Com, NT.
OS/2 Warp:	Now available!

Ms/Miss/Mrs/Mr/Dr _____

Company _____

Address _____

Telephone _____ Fax _____ Email _____

I enclose cheque for A$_____ payable to DLA Technology Please Specify: Macintosh ❒ Windows ❒

OR

Please charge my Mastercard ❒ Visa ❒ Amex ❒ Card no. |_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|

Expiry Date _____ Signature of Cardholder _____

DLA Technology Level 3, 35 Clarence Street Sydney NSW 2000 AUSTRALIA

Tel: (+61 2) 9262 2255 Fax: (+61 2) 9262 2290 E-mail: info@dlagroup.com.au