

Clarion Magazine

Clarion News

- » [CapeSoft World Tour Cape Town Photos](#)
- » [J-Fax 1.11 Final Beta](#)
- » [New Product From Eric Jacobowitz](#)
- » [Atlanta CapeSoft Training on eBay](#)
- » [CapeSoft Training - Chicago Full](#)
- » [AdiSetTabOrder Sets Tab Order Automatically](#)
- » [FullRecord 1.52](#)
- » [J-Spell v1.04](#)
- » [RMChart Demo](#)
- » [Data Mapper 1.25](#)
- » [Class Anatomy 1.01](#)
- » [Firebird 2 Release 2](#)
- » [CapeSoft Cape Town Registrations Close Today](#)
- » [New IP Driver Demo With 50MB Data File](#)
- » [Whitemarsh April Announcement](#)
- » [Huenuleufu Competitive Upgrade](#)
- » [J-Spell 1.02 Beta](#)
- » [FullRecord 1.50](#)
- » [J-Fax 1.08 Beta](#)
- » [SetupBuilder 5.4 Build 1501](#)
- » [Huenuleufu Development Site Updated Again](#)
- » [CapeSoft World Tour for US, Cape Town, UK](#)
- » [Updated template: Drive Information Template](#)
- » [xMouseGestures 1.1](#)
- » [Locus Free Templates Updated](#)

- » [Data Ferret for Clarion Apps](#)
- » [xClarionSwitcher 1.07](#)
- » [J-Spell Template Released](#)
- » [UK CUG and CapeSoft Training Day](#)
- » [CapeSoft World Tour Registration](#)
- » [Epsilon Becomes "Epsilon Marketing & Communications"](#)
- » [New Template Set: SimDatesClass](#)
- » [SetupBuilder 5 on Microsoft Digital Locker](#)
- » [Interview with Bruce Johnson and Jono Woodhouse](#)
- » [Huenuleufu Expands](#)
- » [New Australian Clarion Distributor](#)
- » [Dev Dawn Capesoft Profiler Testimonial](#)
- » [CapeSoft World Tour USA Update](#)
- » [zAssist Alpha Test](#)
- » [Data Equity Embed Assistant 2.10](#)
- » [Class Anatomy Freeware](#)
- » [Establishing Credibility Using About & Contact Pages](#)
- » [Data Mapper 1.24](#)
- » [FullRecord 1.33](#)
- » [xTipOfDay 2.5](#)
- » [New Trial Download for Application Assistant](#)
- » [Data Equity Application Assistant 2.11 Update](#)

[\[More news\]](#)

Latest Free Content

- » [Two New E-Books](#)

[\[More free articles\]](#)

Clarion Sites

- » [SealSoft Company](#)

Podcast



[\[Track lists, more podcasts\]](#)

Clarion Blogs

Latest Subscriber Content

[Printing Unknown Queue Fields](#)

Bjarne Havnen demonstrates how to create reports from queues, when you don't know the structure of the queue to be reported.

Posted Friday, May 26, 2006

[Aussie DevCon: Windows Services, CapeSoft Tools](#)

DevCon day three coverage includes Jono Woodhouse on writing Windows services with Clarion, Jono and Bruce in a fast-paced demo using CapeSoft accessories, and the conference wrap-up. Includes a link to Russ's earlier C7/Clarion.NET presentation; Dave Beggs' presentation on Charles Edmonds' Pro-Series products will appear later.

Posted Friday, May 26, 2006

[Using SQLIdentity in Clarion 6](#)

While the new SQLIdentity feature in C6.3 9051 works well for, say, a batch operation, where you need to get the identity of a just-added record, it does not address the problem of getting an identity value on a form, when you want to add child records while adding the parent record. Nick Pattinson shows how to get the rest of the job done.

Posted Thursday, May 18, 2006

[Aussie DevCon: Mambo, Critical Sections/Threading, and Ingres](#)

The Day 2 afternoon sessions included: Tony York on building web sites with Mambo and Joomla!; Bruce Johnson and Jono Woodhouse on critical sections and threading in C6; and open source Ingres SQL by Sean Hennessy.

Posted Thursday, May 18, 2006

[**Aussie DevCon: NetTalk 4, Debugging, XML and SetupBuilder5**](#)

The Day 2 morning sessions included: Bruce Johnson on building a web interface using NetTalk 4; Russ Eggen with debugging tips and techniques; Jono Woodhouse on XML in C6; and Geoff Spillane on Lindersoft's SetupBuilder5.

Posted Thursday, May 18, 2006

[**Aussie DevCon Day 1, Part 2: SQL, New SV File Drivers**](#)

Our Aussie DevCon coverage continues with a look at Bruce Johnson's "A Tale of Three Knights" (the IP, Memory, and Dynamic drivers), Sean Hennessy's presentation titled "To SQL or not to SQL," and some pics from the Friday evening meal.

Posted Friday, May 12, 2006

[**Aussie DevCon Day 1, Part 1: ClarionTools, PHP, CapeSoft Profiler**](#)

Geoff Robinson reports on Bruce Johnson and CapeSoft Profiler, Andy Kaczmarczyk and ClarionTools, and David Griffiths' session on the PHP templates.

Posted Thursday, May 11, 2006

[**Aussie DevCon: Russ Eggen on C7 and Clarion.NET**](#)

Russ's presentation was on the last day of the Aussie DevCon, but leads our conference coverage to the high level of interest in C7 and Clarion.NET. Geoff Robinson reports.

Posted Tuesday, May 09, 2006

[**Two New E-Books \(free article\)**](#)

We've released two new e-books: Clarion Algorithms Volume 1 and Volume 2. Together, these two e-books contain all of Alison Neal's articles on classic computing algorithms translated into Clarion. Topics include dynamic stack, list, numerous sorts including insertion, counting, cocktail shaker, shell, radix and heap, binary trees and weight balanced binary trees, tries, queues, priority queues, the Huffman compression/decompression algorithms, networks, graphs, critical path analysis, and more. View the tables of contents online. These e-books are \$9.95 each for anyone with a current subscription, or \$19.95 each for non-subscribers.

Posted Tuesday, May 02, 2006

[**PDF for April 2006**](#)

All Clarion Magazine articles for April 2006 in PDF format.

Posted Monday, May 01, 2006

[\[Last 10 articles\]](#) [\[Last 25 articles\]](#) [\[All content\]](#)

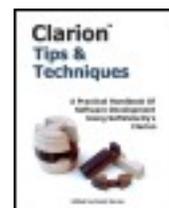
Printed Books & E-Books

[E-Books](#)

E-books are another great way to get the information you want from Clarion Magazine. Your time is valuable; with our [e-books](#), you spend less time hunting down the information you need. We're constantly collecting the best Clarion Magazine articles by top developers into themed PDFs, so you'll always have a ready reference for your favorite Clarion development topics.

[Printed Books](#)

As handy as the Clarion Magazine web site is, sometimes you just want to read articles in print. We've collected some of the best ClarionMag articles into the following print books:



- Clarion 6 Tips & Techniques Volume 1 - ISBN: 0-9689553-8-X
- Clarion 5.x Tips and Techniques, Volume 1 - ISBN: 0-9689553-5-5
- Clarion 5.x Tips and Techniques, Volume 2 - ISBN: 0-9689553-6-3
- Clarion Databases & SQL - ISBN: 0-9689553-3-9

We also publish Russ Eggen's widely-acclaimed [Programming Objects in Clarion](#), an introduction to OOP and ABC.

From The Publisher

[About Clarion Magazine](#)

Clarion Magazine is your premier source for news about, and in-depth articles on Clarion software development. We publish articles by many of the leading developers in the Clarion community, covering subjects from everyday programming tasks to specialized techniques you won't learn anywhere else. Whether you're just getting started with Clarion, or are a seasoned veteran, Clarion Magazine has the information *you* need.

[Subscriptions](#)

While we do publish some free content, most Clarion Magazine articles are for subscribers only. Your [subscription](#) not only gets you premium content in the form of new articles, it also includes all the back issues. Our [search engine](#) lets you do simple or complex searches on both articles and news items. Subscribers can also post questions and comments directly to articles.

Satisfaction Guaranteed

For just pennies per day you can have this wealth of Clarion development information at your fingertips. Your Clarion magazine subscription will more than [pay for itself](#) - you have my personal guarantee.

Dave Harms

ISSN

Clarion Magazine's ISSN

Clarion Magazine's [International Standard Serial Number](#) (ISSN) is 1718-9942.

About ISSN

The ISSN is the standardized international code which allows the identification of any serial publication, including electronic serials, independently of its country of publication, of its language or alphabet, of its frequency, medium, etc.

Clarion Magazine

Clarion News

[Search the news archive](#)

[**xTipOfDay 2.7**](#)

xTipOfDay 2.7 is now available. Changes include: Small template bug fix for compatibility with xToolTip; Error in description of exported method was fixed; Error in the class for showing the next Tip was fixed.

Posted Wednesday, June 07, 2006

[**Class Anatomy 1.2.0**](#)

Class Anatomy version 1.2.0 has been released. Debug code has been removed, and new projects no longer inherit properties from the previous project.

Posted Wednesday, June 07, 2006

[**iAlchemy Site and Email Down**](#)

The iAlchemy web site and email server will be down for a few days during a move to a new server.

Posted Wednesday, June 07, 2006

[**Clarion C6.3 Build 9053**](#)

Clarion C6.3 build 9053 has been released. Patch and full install releases are available to eligible registered users. Although a recompile of applications/DLLs that use C60Runx.dll is not needed, there is an important bug fix that may require you to recompile anyway. See the release notes. This release also includes updates to the Templates to add support for some Clarion 7 features. Specifically the template chains (both ABC and Clarion) contain

new options to configure the new UI features for Menu's, Sheet/Tabs and Tabbed MDI styles. This is for Core Subscription users who will receive the Clarion 7 tech preview release. The generated code for these settings is wrapped in a conditional compile block, "COMPILE ('**CW7**',_CWVER_=7000)", so it is only compiled if the system detects Clarion7.

Posted Wednesday, June 07, 2006

J-Spell 1.14

J-Spell 1.14 is now available for download. This build includes several significant improvements, as well as the ability to exclude controls from being spell-checked, and also an example showing how to use J-Spell to spell-check a CapeSoft FileExplorer control.

Posted Monday, June 05, 2006

FullRecord 1.53

FullRecord 1.53 has been released. Changes include: Local override to avoid tracking of unwanted procedure names (for example Report Previewer); New global setting to avoid the declaration of the zlib library (for compatibility with other templates that also use it); Fixed some local code from extension templates being generated when the template was globally disabled.

Posted Monday, June 05, 2006

iQ-XML 1.18

iQ-XML version 1.18 is now available. Changes include: All tag names now 128 characters; Changed iqxml.dll to use DOS driver so all field lengths from files can be unlimited instead of 64K; Added optional Line Terminal Flag to XML:CreateFile() so XML files can have line terminators or not, or custom ones; Added "Programmer Integrity Check" parameter to the XML:CloseParent() function so programmer can make sure they are closing the proper tags; Sped up the parser, in some instances, up to four times faster, especially with larger XML documents or ones with many elements; Corrected F10-Expand Data field in XML:DebugQueue to resize properly when losing focus; Unlimited field and record lengths, however, the storage of the field is still limited to the first 10K of data; Changed all XML:DebugQueue fields from 128 bytes to 10K so all data can be viewed; Enhanced online documentation with more examples and some fixes to the documentation.

Posted Monday, June 05, 2006

[xWordCOM 1.6](#)

New in xWordCOM 1.6: Version for Clarion 6.3 (Build 9052); New installation kit for Clarion 6.3; Updated demonstration program.

Posted Monday, June 05, 2006

[xClarionSwitcher 1.08](#)

xClarionSwitcher 1.08 has been released. This is Free Utility for easy switching between Clarion 6.3, Clarion 6.2, Clarion 6.1 or Clarion 6.0 installed on single PC. New in this version: New option to check if Clarion is running at program start; List of folders for renaming and restore.

Posted Monday, June 05, 2006

[zAssist Adds Features](#)

A new release of zAssist is now available. There is a new utility template that displays global variables.

Posted Monday, June 05, 2006

[Castle Computer Technologies Offers Testing Service](#)

Castle Computer Technologies is starting a new software testing service. Castle will do its best to break your app, and then generate a report letting you know how it was broken. Template checking also available. Introductory \$99 per application fee.

Posted Monday, June 05, 2006

[xQuickFilter 2.19](#)

New in xQuickFilter 2.19: Process extension template; Error with update of queue of filters after saving new filter was fixed.

Posted Monday, June 05, 2006

[Clarion Project Manager](#)

The Clarion Project Manager web site lets you host open and closed source projects for public access. Features include: Bug tracker; Task manager; Suggestions list; Project dependencies; Version history; Links and RSS feeds; Ratings. Feel free to register your project with the Project Manager!

Posted Monday, June 05, 2006

EasyVersion 3 Released as Freeware.

EasyVersion features include: Automatic Build number generation; Automatic generation of fields with Version information for use in Splash and About screens (using Control templates); Possibility of saving/loading of current parameters of the description of the version to/from the "Project"; Encryption of constants containing a version information; Export of class variables (with Version information); Version Stamping of EXE and DLL with the generated version information in 32 bit applications; Possibility to add the additional "free" information into the version; Possibility of using macros like %V%; %L%; %X% in a field "File Name" on the Tab "Version"; Possibility of correct formation of EXPORT (EXP) file name depending on macros like %V%, %L%, %X% which were used in the Target file name in the Project (standard templates do not make it); Documenting of process of generation of application using a LOG-file.

Posted Monday, June 05, 2006

xXPframe 1.4

New in xXPframe 1.4: Support for Clarion 6.3 (build 9052); Changes in class for faster display of menu; Updated install and demo.

Posted Monday, June 05, 2006

PrintWindow Gold 1.01

PrintWindow Gold 1.01: Improved handling of drop lists (Now you can print drop lists not linked with a variable); Enhanced example apps; New demo.

Posted Monday, June 05, 2006

SetupBuilder 5.4 Softpedia Review

May 31, 2006 - Lindersoft, a widely-recognized provider of installation authoring and software configuration solutions, announced today that Softpedia.com reviewed its highly popular SetupBuilder 5.4 installation system. "This is the best setup creator I've seen so far.", wrote Cosmin Nitu, Softpedia's Software Reviews Editor. "Setup Builder is a great authoring and configuration management utility that any programmer should use when creating installation packages for their programs. It has advanced features that can be used by beginners, and it gives your program's setup a professional look in a matter of minutes. Even if you're not a developer, try it just for the sake of it!" Cosmin Nitu added.

Posted Monday, June 05, 2006

[CapeSoft World Tour Cape Town Photos](#)

Jono has uploaded some photos from the Cape Town World Tour Leg.

Posted Friday, May 26, 2006

[J-Fax 1.11 Final Beta](#)

J-Fax 1.11 is now available for download. This build includes a number of improvements, and will be the final Beta release. Early next week J-Fax will be released as Gold, and the price will be increasing significantly. If you buy it at this discounted Beta price you will get the upgrade to the Gold release next week for free.

Posted Friday, May 26, 2006

[New Product From Eric Jacobowitz](#)

Eric Jacobowitz has launched a new site with a new product called ShipMate. ShipMate keeps track of all your shipments (UPS, FEDEX, DHL and USPS). Enter all incoming and outgoing tracking numbers in a master list which Shipmate uses to retrieve delivery information from the carrier's website. Shipmate will save the delivery status in an easy to read database and checking the progress of packages. you can also look up phone numbers, area codes and zip codes, get driving directions, currency conversions and even weather reports!

Posted Friday, May 26, 2006

[Atlanta CapeSoft Training on eBay](#)

Up for grabs is the last single event registration for the CapeSoft Training to be held in Atlanta, Georgia on June 23 and 24. Bruce, and Jono from CapeSoft will be training Clarion programmers on a number of the CapeSoft accessories, including NetTalk 4, Office Inside, Hot Dates, Replicate and many others. This auction is for the event fee only. Accommodation is available at the hotel (\$59 per night including breakfast). Travel, accommodation and meals are not included in this auction.

Posted Friday, May 26, 2006

[CapeSoft Training - Chicago Full](#)

Both Atlanta and Chicago training sessions are now full. Newark and LA both have some space left - although that is also filling up quite fast. Newark registrations close on June 4.

Given that two of the other three are already full, that probably means LA will close on June 4 as well. So please register in time to avoid disappointment.

Posted Friday, May 26, 2006

[AdiSetTabOrder Sets Tab Order Automatically](#)

Bjarne Havnen has a little freeware template that will automatically set the correct tab order for controls, based on their position in the window.

Posted Friday, May 26, 2006

[FullRecord 1.52](#)

FullRecord 1.52 is now available. There are a couple of fixes and improvements in this release, including better audit storage and interaction with NeatMessage.

Posted Friday, May 26, 2006

[J-Spell v1.04](#)

J-Spell 1.04 is now available for download, along with the latest demo.

Posted Friday, May 26, 2006

[RMChart Demo](#)

This demo program using RMChart DLL (www.rmchart.com) has been graciously supplied by Jose Herera to the Clarion community.

Posted Friday, May 26, 2006

[Data Mapper 1.25](#)

Data Mapper 1.25 has been released, and is still free. If you already have it, just use the "Check for Updates" feature from the "Help" menu. This build includes some new reports, new buttons on the toolbars, new menu items, a couple of bug fixes, etc. The free version simply displays a nag screen as it starts up, and you're limited to saving two dictionaries and five views per dictionary at one time.

Posted Friday, May 26, 2006

[Class Anatomy 1.01](#)

Version 1.01 of Class Anatomy has been released. Changes include: Fixed a bug in the

export definition; Added the option to jump to a procedure in the CLW instead of INC;
Added a 'check for updates' (for the program itself) option.

Posted Friday, May 26, 2006

Firebird 2 Release 2

Kelvin Chua reports that release 2 of Firebird 2 is now available.

Posted Friday, May 26, 2006

CapeSoft Cape Town Registrations Close Today

Registrations for the Cape Town leg of the tour close today. Actually there's only one space left, so first one there gets it. Registrations for the USA leg will close early in June (to allow time for printing of notes etc...). There are only a handful of places left in Atlanta and Chicago, and both New Jersey and LA are pretty much half-full already. Register early to avoid disappointment.

Posted Friday, May 19, 2006

Clarion Magazine

Printing Unknown Queue Fields

by Bjarne Havnen

Published 2006-05-26

Now and then I make procedures that use a lot of queues which have some fields in common, but the presentation in lists requires varying numbers of LONGs to hold color, style, icon information, making it a little nightmare to keep a TYPE definition synchronized with the local queues and listboxes.

I need to create reports from these queues, and I like to use a generic report that reports the fields I know, ignoring every other field (and, for that matter, the order in which this fields appear).

I taught a Clarion class the other day where one of the participants needed to accomplish the same task. I demonstrated one way to do it, which I find rather clever.

In Clarion 6, you need to switch templates to change a report from table-based to queue-based. I don't understand why queue-based reports are managed using code overriding the standard template since a class to process a queue exists (see my earlier article about using the [QueueProcessManagerClass](#)), but the template override works just fine and is very easy to understand.

The first thing I do is design the report using locally defined variables. The report's prototype is set to (*Queue passedQ). Note that the queue is not named; the procedure just takes a queue, no matter what design that queue has.

I set the reports data source to Queue and tell the report to process passedQ. Now, I can call the report with any queue. So far, no information is visible in the report, it just loops Records (passedQ) times. I need to find the fields I want for the report and assign their

value to the local variables.

Who()?, What()?

My very first queue reporting task needed to report on five queues with record history data. All these queues had only three fields in common: Prompt, CurrentValue and NewValue. Each queue had a variety of color fields, icons and other fields with obscure usage.

I do not particularly like to design reports, so if I can get away with one report for all these queues, I am more than happy.

Among my favourite Clarion functions are `Who()` and `What()`. These functions make it possible to extract the information needed from any structure. If I want to find a field labelled Name, I can easily locate and use that field with no knowledge of the structure in question:

```

i=0
Loop
    i+=1
    If Not Who(passedq,i)      !no field
        Break                !terminate loop
    End
    If Match(Who(passedq,i), '*Name',Match:NoCase+Match:Wild)
        !field found, assign value
        Loc:Name = What(passedq,i)
    End
End!loop

```

It's that simple. Note that I'm using the Clarion MATCH function, which accepts a regular expression. I'm doing a case insensitive match, and I use a wildcard so I can handle queues with and without prefixes. All I need to do in my report procedure is to specify all the fields and embed the code in `TakeRecord` before `Print()`.

Obviously, doing the loop for every record in the queue is overkill. One time should be sufficient if I use reference variables instead. Unfortunately, reference variables can't be used in a report, so I need both a reference and an assignment to the local variable for every field I need to report. I move the code above to the beginning of the procedure and replace the line

```
Loc:Name = What(passedq,i)
```

with

```
Ref:Name&=What (passedQ, i)
```

Then , in the Print () embed, I do the field assignment

```
Loc:Name = Ref:Name
```

This is simple code in two embeds enables me to send in any queue holding a predetermined field or fields. I can even send a queue with fewer fields, all I get is an empty column.

Reader Comments

[Add a comment](#)

Clarion Magazine

Aussie DevCon: Windows Services, CapeSoft Tools

by **Geoff Robinson**

Published 2006-05-26

The final day of the conference was intended to include two hour-long sessions on Clarion 7 and Clarion.NET. Organiser Tony York was disappointed and frustrated that what he had expected from Soft Velocity had not eventuated, but took it in his stride and reorganized the schedule for the day.

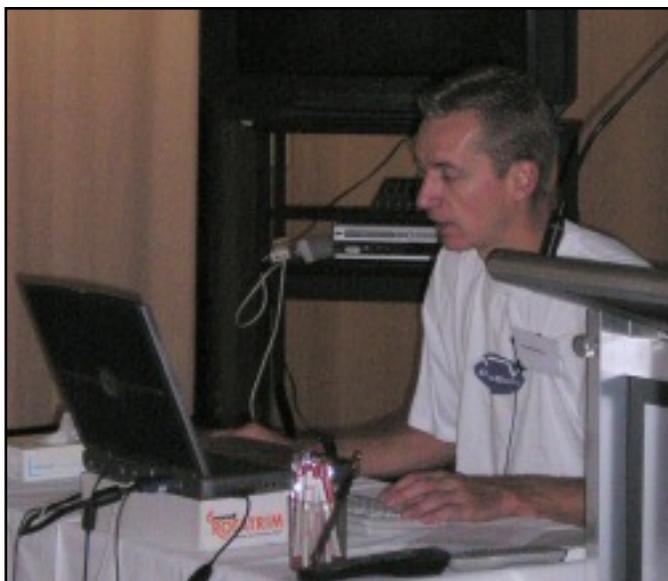
It turned out to be a very enjoyable and informative day anyway and Russ Eggen had had a long phone chat with Robert Zauere and so was able to give us an [update on the progress of C7 and .Net products](#).



Services - Jono Woodhouse

The first session for the day was on Windows services. Services are programs that start up when the PC is fired up and are running before you log on. Similarly they continue to run should the user log off, and so are appropriate for programs that you want running all the time unobtrusively in the background. Examples of services might be anti-virus programs or mail servers.

Soft Velocity's IP driver, which Bruce had covered on the first day of the conference, uses a service on the server, and services are also often useful with NetTalk-enabled applications.



In order to see what is running on your computer, Jono mentioned [Process Explorer](#) which he described as "Task Manager on steroids". Process Explorer will indicate which of the running tasks are services.

Service Manager is used to start and stop a service; this is different from normal Windows applications where you can simply double-click an exe to start it up. Also services need to be registered, a process referred to as "Installing a service".

Services run under a different username and password, and while this gives tighter security it does also add some complexity. Services can run on Windows NT, 2000, XP or 2003 but Jono did not recommend running them on NT.

Windows Services Manager can be run from the Computer Management console. Under XP click Start, and then click Control Panel. Click Performance and Maintenance from the category view, click Administrative Tools, and then double-click Computer Management. Then you can expand the Services and Applications branch on the tree (I did this when I got home and was surprised at how many services I had running!).

Jono mentioned getting information from www.blackviper.com; this site seems to have disappeared, but I found [another page](#) which appears to have much of the Black Viper information. Alternatively you can see old versions of the Black Viper site using the [Way Back Machine](#).

To start and stop a service from a DOS shell, type:

```
net start <service name>
```

and

```
net stop <service name>
```

Jono mentioned that the [ABC Free Templates](#) by Vince Sorensen allow you to build services, and that Capesoft's product [SelfService](#), which is based on Vince's work (Vince gets royalties from Capesoft for SelfService sales), makes creating services easier and is recommended. The only caveat is that SelfService does not support Clarion versions prior to C6.

All in all an informative session with lots of information packed into a short time.

Fun with Capesoft - Bruce Johnson and Jono Woodhouse

As the session name implied, this was a "fun" session - complete with Bruce in Mickey Mouse gloves doing Mickey impersonations.

This was a polished session where Jono typed furiously on the keyboard while Bruce did the [spruiking](#) out the front. The aim of the session was to see how much could be achieved in one hour, and how many [Capesoft tools](#) they could plug into an app without it exploding. The application they built was related to share prices, and it was a phenomenal performance which really emphasized how productive you could be with Clarion and some third party add-ons.



They started with a wizarded app and then started bolting on the Capesoft templates. First off was [TearOff](#) which allowed the users to make their own toolbar with the functions that they use most often.

Next was [FM3](#) so that the user could have new requirements, and easily convert the existing data when fields were added etc. A show of hands indicated nearly everyone had FM2 or FM3. Russ Eggen claimed that he never let his users have new requirements once the spec was agreed. I'm not sure if he was joking or not.

Next they added [MessageBox](#) - this gives you much flexibility compared with your usual message() statement, including logging messages to a file in case a user rang up about a message but was vague about what it said. (Who would have users like that?) You can also log what button they pressed.

[EZHelp](#) was next - so users could write their own documentation rather than you having to

do it. This was context sensitive so each field could have its own help.

The tools came thick and fast - [WinEvent](#) was next, followed by [Office Inside](#) (which was used to import share prices out of an Excel spreadsheet), and [Makeover](#) to change the appearance of the app, then [Draw](#). With regard to fonts Bruce prefers Tahoma, but acknowledged there could be problems with ClearType on LCD screens - but no worries, Capesoft's AnyFont would come to the rescue.

Progress allowed the app to have better looking progress bars, and used Capesoft Draw, while [SendTo](#) allowed the user to simply print an existing browse, or send it to a file, or an email or the clipboard or to FTP... the options seemed endless. Next [Hyperactive](#) allowed links to a web-site or to allow the user to easily email the developer. Then Jono added some simple code that would cause a GPF - and [GPF Reporter](#) was there to show exactly where in the code the problem was.

[Insight Graphics](#) quickly supplied a daily plot of the opening and closing share prices along with the high and low for the day. [Hot Dates](#) was added and made it easy to see that the shares often fell on a Friday. Bruce mentioned that often you can easily visually see trends when they are put on a calendar. Are more of your employees away on Fridays or Mondays?

As I mentioned this was a real tour de force of Clarion and Capesoft products - made most entertaining by the antics of Bruce and Jono.

Russ Eggen on C7/Clarion.NET

Read the session report [here](#).

Charles Edmond's Pro-Series - David Beggs

This session will be covered in a separate article.

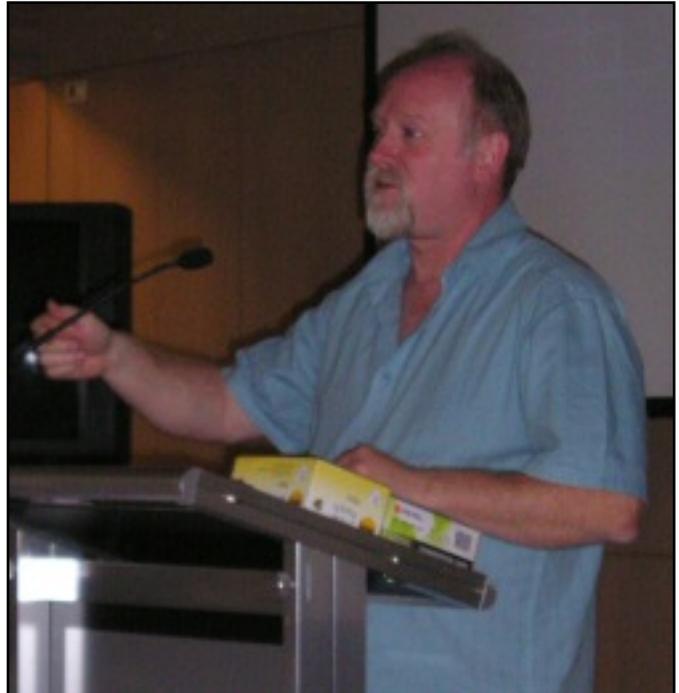
ClarionTools demo app

The final session was another one by Andy Kaczmarczyk from [ClarionTools](#) (see the [Day 1 report](#)) where he took the Clarion University (school.app) demo program that comes with Clarion and transformed it by adding his tools - especially View Wizard. Great stuff. He also mentioned using [Snazzy Listbox](#), a freeware template written by John Christ, to change colors on listbox headers.

Conference wrapup

Tony wrapped up the conference with the prize draws. Everyone present received a prize from the donors - some were lucky enough to get two! Some won a T shirt while others won software. I think the sponsors/donors deserve a free plug here:

- [Ace Icons](#)
- [Capesoft](#)
- [Clarion Magazine](#)
- [Clarion Tools](#)
- [Data Equity](#)
- [David Beggs](#)
- [iAlchemy](#)
- [Lansrad](#)
- [Lindersoft](#)
- [Prodomus](#)
- [SoftVelocity](#)
- [Solid Software](#)



Well that pretty well wraps up my report on the Aussie DevCon 2006. These events are a great opportunity to get ideas for your applications and meet other developers - some who you know well from previous events and others who you are meeting for the first time.

Organisers Tony and David received a well deserved standing ovation for their efforts.

[Geoff Robinson](#) lives near the beach in Melbourne, Australia, and is an active member of

the local Clarion User Group. His company, [Vitesse Information Systems](#), specializes in software for local government. Geoff was impressed by Clarion back in the DOS days and grabbed the early betas of Clarion for Windows when they first became available; he has been using Clarion as his primary development environment ever since. When not in front of a computer Geoff enjoys listening to music, singing bass in a local choir, and spending time with his three young children.

Reader Comments

[Add a comment](#)

Clarion Magazine

Using SQLIdentity in Clarion 6

by Nick Pattinson

Published 2006-05-18

In Bob Zaunere's [blog entry](#) on C6.3 developments and enhancements, he highlights how the SQL drivers can now fetch an identity (that is, an auto-increment primary key) from the backend when a new record is inserted.

I was particularly pleased to see this since, at a UK Clarion Users' Group meeting last year when we had a teleconference with Bob, I highlighted this issue with using Clarion with SQL, and suggested that the lack of a solution out of the box would come across as a major deficiency. He took the comment on board, and reckoned that this would be a new feature for C7. So, he's early!

Unfortunately, in C6.3 build 9050, this feature didn't work completely, but it's been fixed in build 9051; I have a couple of apps in the wild that use the feature and have found it stable and reliable.

Regrettably, this makes redundant John Griffiths' very elegant solution [SQL Identity: Another Approach](#), published in 2004.

There remains, however, work to be done in this area. While this new feature works well for, say, a batch operation, where you need to get the identity of a just-added record, it does not address the problem of getting an identity value on a form, when you want to add child records *while* adding the parent record.

When inserting a new row in an ISAM database, the database driver actually inserts the record at the start of processing and changes the request to ChangeRecord. The effect of this is that, when inserting a new child for the recently-inserted parent, a parent row exists and foreign key constraints on the new child will be satisfied.

When using an SQL backend, however, the database driver only inserts the new row when the record is saved – typically when the user presses OK. There is thus no parent record in the database while the user is filling out the new record, and therefore no parent primary key. The child's insert will fail because of violation of the foreign key constraint.

The first step is to be able to get the primary key value using the new driver feature; the second is to cause an insert to happen, so that the primary key value is available for the child records.

Getting the primary key value

A simple setting in the dictionary will cause an application to retrieve a primary key value that the server has generated during an insert; this value can now be populated into the foreign key field of new children.

The dictionary settings are well documented in the Online Help – look under "Server Side Auto incrementing" in the index,

or search for "IsIdentity". If you are using Microsoft SQL Server, then you need to add the Boolean user option to the field in your dictionary that maps on to the Identity column on the backend (See Figure 1.)

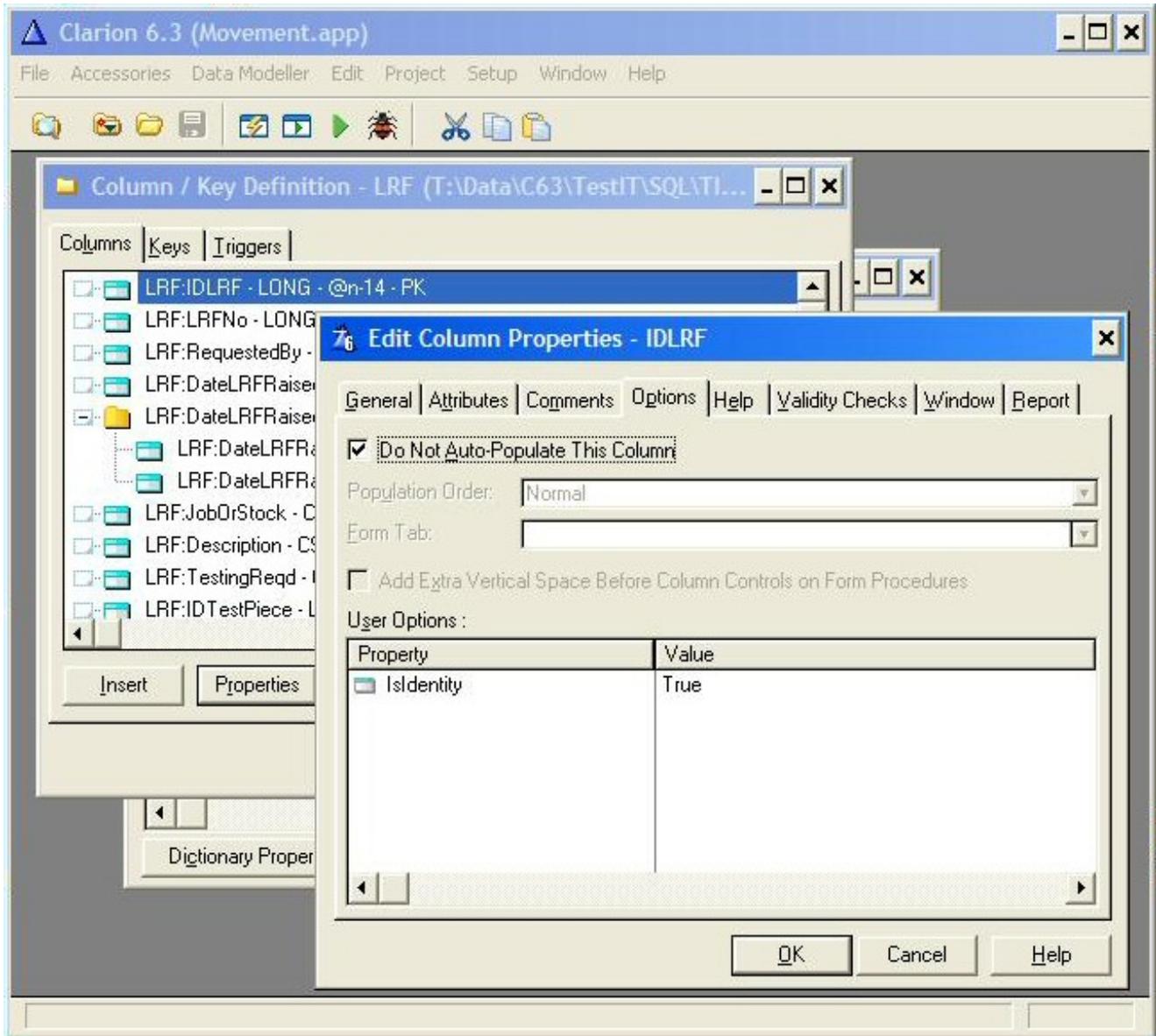


Figure 1: User Option for an Identity column ([view full size image](#))

SoftVelocity has made it easy for me to get the new value for an automatically-incremented column on the backend.

Coding was required, however, to insert the new row before the first new child can be added. When using an SQL driver, Clarion apps only insert a new row on *completion* of the form, yet I want to add children on the same form that adds the new parent.

Whilst responses to Bob Z.'s blog suggests that SV hasn't addressed this issue in C6.3, the solution is actually quite straightforward:

In any ABC window procedure, the procedures to maintain children are called in `ThisWindow.Run PROCEDURE(USHORT Number, BYTE Request):`

CODE

```
GlobalRequest = Request
```

```

EXECUTE Number
  SelectEngineer
  SelectTestPiece(LOC:Dummy,LRF:IDTestPiece)
  UpdateLRFJob
END
ReturnValue = GlobalResponse
RETURN ReturnValue

```

In the above example, the new parent needs to exist prior to the creation of a new LRFJob row by the procedure UpdateLRFJob.

The Request parameter passed into ThisWindow.Run contains the required record operation – InsertRecord, ChangeRecord, SelectRecord, etc...

If I can detect whether or not the parent exists in the database, then I can also automatically insert the new parent immediately before the call to the procedure that will insert the new child.

Further, if I take control of the form's SELF.Request, and ensure that its value is ChangeRecord after I've successfully inserted the new parent row, then I have a simple mechanism for telling if I need to create the new row or not.

Finally, I only need to concern myself when Request is InsertRecord: if it's ChangeRecord, then I know that the parent exists otherwise there wouldn't be a child to Change. Similarly for DeleteRecord and, if Request is SelectRecord, I'm looking up a foreign key record for this form's record.

In pseudo-code:

```

if SELF.Request = InsertRecord and Request = InsertRecord
  if form's data meets validation rules,
    insert new row
    change SELF.Request to ChangeRecord
  end
end
if above code didn't fail,
  Continue with remainder of procedure
End

```

In Clarion code:

```

ThisWindow.Run PROCEDURE(USHORT Number,BYTE Request)
ReturnValue          BYTE,AUTO
! Start of "WindowManager Method Data Section"
! [Priority 5000]
! End of "WindowManager Method Data Section"
CODE
! Start of "WindowManager Method Executable Code Section"
! [Priority 50]
!NGPSQLIdentity before parent call
If SELF.Request = InsertRecord and Request = InsertRecord
  If not Access:LRF.ValidateRecord() !Ensure form data's valid before Insert
    If Access:LRF.Insert()
      Message('Failed to insert new LRF record prior to inserting new child.')
      ReturnValue = RequestCancelled
    Else
      SELF.Request = ChangeRecord
    End
  End
End

```

```

Else
    ReturnValue = RequestCancelled
End
End
If ReturnValue <> RequestCancelled ! Do nothing if Insert failed
!End of NGPSQLIdentity before parent call
! [Priority 2800]
! Parent Call
ReturnValue = PARENT.Run(Number,Request)
! [Priority 5500]
IF SELF.Request = ViewRecord
    ReturnValue = RequestCancelled
ELSE
    GlobalRequest = Request
    EXECUTE Number
        SelectEngineer
        SelectTestPiece(LOC:Dummy,LRF:IDTestPiece)
        UpdateLRFJob
    END
    ReturnValue = GlobalResponse
END
! [Priority 7500]
!NGPSQLIdentity after parent call
End !If ReturnValue <> RequestCancelled
!End of NGPSQLIdentity after parent call
! [Priority 9300]
! End of "WindowManager Method Executable Code Section"
RETURN ReturnValue

```

In the above (real, working) example, I check whether I need to insert the new row (If SELF.Request = InsertRecord and Request = InsertRecord). If I do need to do the insert, I check that the new row satisfies my validity rules (If not Access:LRF.ValidateRecord()) and, if it does, I insert and change the form's request to ChangeRecord.

If, however, the insert fails, I set ReturnValue to RequestCancelled and surround the standard Clarion procedure code (priority 2800 to 7500) with a test for ReturnValue, and only execute it if ReturnValue is not RequestCancelled.

Since this code falls well into the category of something one will always do, it ought to be templated. Now, if I described myself as an indifferent template programmer, I'd be dramatically overstating my skills, so I see the following as a functional way forward, to be taken to its conclusion by others. Had I the skills, I would wish the template only to add or execute the code if the form's file has a PROP:SQLIdentity, and I would like the template to get PROP:Name for the form's file from the associated template values.

Here's the basic extension (#AT statements have line breaks added):

```

#EXTENSION(NGPSQLInsert,'SQL: Insert parent prior to Child insert'),PROCEDURE
#!
#! Inserts parent SQL row immediately prior
#! to execution of procedure to insert Child.
#! Ensures new Child won't fail FK Constraint.
#DISPLAY('This extension manages the insert')
#DISPLAY('of a new SQL row')
#PROMPT('Table into which new row is to be inserted:',FILE),%NGPTable,REQ
#AT(%WindowManagerMethodCodeSection,'Run','(USHORT Number,

```

```

        BYTE Request),BYTE'),PRIORITY(2500),WHERE(%ProcedureCategory='Form')
!NGPSQLIdentity
If SELF.Request = InsertRecord and Request = InsertRecord
  If not Access:%NGPTable.ValidateRecord()
    If Access:%NGPTable.Insert()
      Message('Failed to insert new %NGPTable record prior to inserting new child.')
      ReturnValue = RequestCancelled
    Else
      SELF.Request = ChangeRecord
    End
  Else
    ReturnValue = RequestCancelled
  End
End
If ReturnValue <> RequestCancelled
#ENDAT
#AT(%WindowManagerMethodCodeSection,'Run','(USHORT Number,BYTE Request)
    ,BYTE'),PRIORITY(8500),WHERE(%ProcedureCategory='Form')
!NGPSQLIdentity
End !If ReturnValue <> RequestCancelled
#ENDAT

```

Summary

As of version 6.3, build 9051, Clarion has the ability to reliably retrieve server-side auto-increment (identity) values. With the additional code I've shown here, you can use this new functionality to add child records while adding the parent record.

Please post any improvements to the template as reader comments.

Born in the early 60's in south England, [Nick Pattinson](#) moved to Manchester for college, where he's remained ever since. Nick's background is in mechanical/production engineering; it was during a summer job at college that he first came across IT, and, after graduating, all his jobs required bringing IT to bear on business issues. Early on he learned DataEase, and finally got to do some *real* programming in the mid-1990s when he found himself committed to delivering a new stock control system in nine months using Oracle SQL, Forms and SQL*Reportwriter (it was on time, and it worked...) Later, when looking for a replacement for DataEase, he happened across Clarion, whose demo version was instrumental in his conversion. These days, Nick lives just south of Manchester UK with his partner, three cats, two chickens and an indeterminate number of frogs (who, fortunately, keep out of the house.) He's been self-employed for five years and currently spends most of his time doing programming, analysis and system implementations for his former employer. He's carefully ignoring the fact that he'll be 45 in June.

Reader Comments

[Add a comment](#)

- » [According to a RZ blog entry, there exist an option within...](#)
- » [The EMULATEAUTONUMKEY works, although with the regular...](#)
- » [Hi What happens if the user decides to delete the...](#)

Clarion Magazine

Aussie DevCon: Mambo, Critical Sections/Threading, and Ingres

by **Geoff Robinson**

Published 2006-05-18

Read the [Day 2 morning report](#).

Let's do the Mambo - Web Sites made Easy - Tony York

We were very well catered for food-wise at the conference - so much so that I wouldn't be surprised if we went home a kilo (2.2 pounds for those in the US) or so heavier. After one of these filling lunches, we headed to the next session which was by Tony York from Perth - the main organiser of the conference.

Tony's session was on [Mambo](#) and [Joomla!](#) - open source products that enable you to easily put together websites. Joomla! is a renamed release of Mambo. Wikipedia says that Joomla! is a *phonetic spelling of the Swahili word "jumla" meaning "all together" or "as a whole"*.



In this session we were not looking at data-centric sites as we had with the PHP template and NetTalk sessions, but more of a magazine style web site. Having said that, the product is actually based on PHP and MySQL and has modules that you simply select and position where you want them.

Tony used JCE (Joomla Content Editor) which is free, and also mentioned wysiwyg.Pro editor which costs around \$50.

Tony took us through the process of building a web site, simply by choosing components, style sheets and setting various parameters. There are lots of free components available on the web, or you

can pay small amounts for commercial offering from third parties. Alternatively you can join a "club" where an annual subscription of \$50 allows you to download as many templates as you like.

Tony used SimpleCSS which is a CSS (Cascading Style Sheet) editor and also mentioned Topstyle lite (free) and TopStyle Pro (about \$80) as alternatives.

Tony then showed us a political web site that he had built. (You may guess his political leanings from the fact that Margaret Thatcher was listed in his list of most admired people in his bio on [Icetips](#) or [ClarionMag](#) <g>). It took Tony just two weeks from first downloading the components and learning how they worked to having the site up and running. Of course there was a fair bit of trial and error in this and he would do it much faster second time around.

All in all it was an entertaining session and showed how tools were available to enable you to quickly get a web site up and running. It also showed the power of CSS to simply change the look of a site.

Critical Sections and Threading in C6 - Bruce Johnson and Jono

Woodhouse

The next session was done by both Bruce and Jono from Capesoft and they took us through what we needed to do to make a C55 app thread-safe in C6. C6 has true 32 bit native threads whereas earlier versions used a co-operative threading system which caused problems - especially in C55. These new threads mean that apps run smoother and one busy window will not lock up others on different threads.

You can also create Windows services which run at startup time without the user logging in. (More details on this in a session on the third day of the conference).

C6 also has improved OLE and COM support and more bug fixes than any earlier version. In short, Bruce and Jono recommend that if you are not already on C6 then you should certainly be heading that way sooner rather than later.

In making the move to C6 you do have to make some changes, as code in a multi-threaded app needs to synchronize its use of global data - but this is not as daunting as it sounds.

You could think of the synchronization in terms of traffic lights (or "robots" as they apparently call them in South Africa) at a road intersection. One direction of traffic (thread) waits until the other direction gets to be clear of the intersection before in turn having their go, safe in the knowledge that they have exclusive use of that part of the road while the light is green. You set up your "traffic lights" by implementing critical sections - which are quite easy to set up.

If you don't use critical sections you could run in to problems:

- Old (out of date or half-changed) data is read, which leads to
- Corrupted values being saved
- Data is lost when two threads save data (the last save wins)

This can lead to some very hard to replicate and track-down bugs.

There are three steps one can use to simply avoid these potential problems:

1. Identify (and reduce where possible) all non-threaded globals
2. Create *one* global critical section
3. Wrap Global data access in a critical section

At this stage Bruce and Jono changed to the next slide and tested us to see if we had been

paying attention. (Luckily I had written it down!)

If you have global queues (which are not threaded), it is advisable to change them to memory tables instead - as these are completely thread safe. Another way to reduce globals is to remove them and pass the data as parameters instead.

A critical section is an area of code that is preceded by a `wait()` and followed by a `release()`. By putting your access to global variables (whether reading them or updating them) in the critical section, you are giving your code a green light while other threads `wait()` at the red light until you are through the intersection and `release()`. You avoid "collisions in the intersection" which could otherwise result in you getting a data variable that is half changed and totally invalid.

How to implement Critical Sections:

- `Include('cwsynchC.inc')` ! in global embeds after global includes
- In global data insert G type is Class and base type is Critical Section
- This generates `G Class(Critical Section)`
- Wrap global data access:

```
G:Wait()
Global data = whatever
G:Release()
```

One way to find all references to global data is to rename the global fields so that the compiler finds all the instances that you have not corrected. As you correct the name you also wrap the code. Alternatively use your favourite search tool to scan the CLW files for each global variable.

Make sure you keep all code between the `wait()` and `release()` as brief as possible - never allow any interaction with the user like a `message()` or `stop()` as they could go off to lunch with that on the screen!

If a variable is set up only once at the start of an application then you probably do not need to wrap it, but Bruce suggested you do it anyway in case a change is made to the application at a future date, which could introduce a hard to find bug. It doesn't take long to add the critical section wrapper, so just do it!

Every `wait()` requires a `release()`. Nesting is fine, as long as the pairs are balanced.

A few cautions:

- You should never return a global variable from a procedure. (Why anyone would ever want to do that is another question!)
- Also you should never start a new thread from within a critical section. (Why anyone would ever want to do that.....)
- Another thing to avoid is creating a number of threads in quick succession. You should either put in a delay between creating each one, or better yet get each thread to post an event and wait for that before creating the next thread.
- Global non-threaded queues can easily be corrupted and should be replaced by in memory files as memory tables are thread-safe.
- You should alter your INI file to use colored syntax for wait() and release(). I think this was referring to the [color values] section in c60edt.ini.
- You should treat static variables the same way as globals - if they are not threaded then wrap all access.

Another thing that was mentioned concerned prototypes in C6. Whereas previously you could just have the prototype with the type and not the name of the field, this can now cause problems - so you are best to have the type AND the name.

Another potential "gotcha" in C6 is in DLLs where you may previously had

```
,external
```

you will now need

```
,external, thread
```

to match the original declaration.

Another thing to bear in mind when moving to C6 is if any of your Clarion procedures are likely to be called from a non-Clarion environment, they should call the `AttachThreadToClarion()` library function at the start of the procedure to make things behave nicely.

A final pointer on moving to C6, related to the Application Converter. The converter shipped with source code in the earlier versions of Clarion and was designed to help with the conversion from the legacy/Clarion templates to ABC. (There is mention of it at the end of David Bayliss's article [Computer Aided Disaster](#). Russ Eggen pointed us to the chapter "Conversion Tools" in the pdf file "Learning Your ABCs - Making a Smooth

Transition From Legacy to ABC". And in particular he referred to the section "Making Your Own Rules" which was adapted from an article by Mike Hanson (see his original [Clarion OnLine article](#)).

Open Source Ingres SQL - Sean Hennessy

The final session for the second day was a presentation on Ingres. Sean had done the more general session comparing TPS to SQL and the advantages and pitfalls of each the previous day, but this presentation was specifically about [Ingres](#).

A straw poll indicated the majority of SQL users were almost evenly split between using MS SQL Server and MySQL, while only one person present was currently using [Firebird](#) and another was using [Mimer](#). (In my case my use of MS SQL Server has been dictated by client policy/standards and I suspect that this is usually the case where Clarion developers are working for corporate clients, as opposed to making a choice for their own development projects).



Sean was impressed with Ingres, which he described as a serious Enterprise Level database. He had initially been thinking of using Firebird but the lack of the roll-forward journaling (which he had mentioned in his previous session) and a perceived lack of up-to-date documentation (perhaps now overcome with Helen Borrie's Firebird book) steered him away from that product. Ingres's documentation, in contrast, was extensive and well laid out.

Sean gave us a run down of the history of Ingres, from its origins as a research project at

the University of California, Berkeley around 1973 or 1974 when its originators were inspired by some technical papers coming out of IBM's System R project. Ingres was originally distributed under the BSD license but then a number of commercial products came from it including Sybase (which later spawned MS SQL Server) and much later Postgres which evolved into [PostgreSQL](#). Informix was another database that came from Ingres, as did Tandem's Non-Stop SQL - so Ingres can be seen to have been very influential.

RTI (Relational Technology Inc) formed in 1980 to commercialise the Ingres code and the company was later renamed Ingres Corporation. This was later purchased by Ask Corporation in 1990 which in turn was bought out by Computer Associates four years later. CA continue to provide commercial support for Ingres but in 2004 released it as open source - probably in the hope of getting a wider user base and service/support contracts. Ingres is free to use and distribute under the CATOSL (CA Trusted Open Source License).

One feature that Ingres has is a different query/data manipulation language called QUEL. This is considered to be superior in many ways to SQL, but SQL won the market share battle so Ingres added SQL in the 1980s.

Ingres runs under Windows, Linux and many Unix varieties.

Sean stated that Ingres had no size limitations and could store terabytes of data. It comes with various utilities one would expect from an Enterprise Level RDBMS - Visual DBA, Visual SQL, Log analysis etc It also has replication and Web integration options, a QBF 4GL and numerous command line utilities.

Ingres does have a much larger client install than Firebird, however. Firebird's footprint is only about a 10MB install whereas Ingres is closer to 150MB.

The ODBC driver you need is version 3.50.49 or beyond. Sean had had some problems using earlier versions with Clarion but said the support was excellent and Ingres brought out a new version that solved his problems. He suggested setting Readlock to NoLock.

Triggers are called Rules, and call stored procedures. These are "after" update triggers. Sean also mentioned "Sequences" which I am a little vague about now but think they may have been the equivalent of auto-inc identity columns..

Security is good and at the C2 level. It is based on the Operating System user although

there are various alternatives.

You have a lot of control over the storage of data in your tables. When you create a table, you can specify the storage type (e.g. Btree, ISAM, heap etc), the page size, the journaling and partitioning and so on. Up to 64000 partitions can be used to segregate data - for example you could have current data in one partition and historic data in another.

All in all Sean is very happy with Ingres and suggests you consider it for your projects.

Fun Night

Whereas the first evening had involved a restaurant meal, dinner on the second evening was at the conference centre, followed by an evening of quizzes and games organised by David Griffiths and his wife Karen. We variously had to answer general knowledge questions, play charades, hum or whistle tunes, draw pictures (Pictionary-style), model plasticine (ever seen someone modelling a joy-stick?), unjumble words and phrases and so on. This gave us all the chance to see how mentally nimble we were and also plenty of opportunity to embarrass ourselves! We all nearly groaned when supper came out as it seemed we had been eating almost continuously. It was an enjoyable end to the second day.

[Geoff Robinson](#) lives near the beach in Melbourne, Australia, and is an active member of the local Clarion User Group. His company, [Vitesse Information Systems](#), specializes in software for local government. Geoff was impressed by Clarion back in the DOS days and grabbed the early betas of Clarion for Windows when they first became available; he has been using Clarion as his primary development environment ever since. When not in front of a computer Geoff enjoys listening to music, singing bass in a local choir, and spending time with his three young children.

Reader Comments

[Add a comment](#)

Clarion Magazine

Aussie DevCon: NetTalk 4, Debugging, XML and SetupBuilder5

by **Geoff Robinson**

Published 2006-05-18

The second day of the conference started early as we had an early breakfast in order to get to the first optional session at 7.30am.

Building a Web Interface using NetTalk 4 - Bruce Johnson

Over the years there have been lots of Clarion products aimed at the web. The previous day's sessions included the presentation on SoftVelocity's PHP templates, and one participant, Ray Creighton from Brisbane, continues to have much success using IC (Internet Connect), which I think was Topspeed's first attempt at a web product back in 1997 - close to a decade ago.

This morning's session showed a new-comer to this area - Capesoft's NetTalk 4. Whereas Capesoft generally has free upgrades, in this case there is so much extra functionality compared with NetTalk 3 (which already did a lot of things!) that there is an upgrade fee.



You use NetTalk 4 to create a web server - hence there is no need for IIS, Apache or Application Broker. This is similar in this regard to Gus Creces' approach in his [Handy Tools](#).

Bruce started by asking: why make a web interface?

- Accessible from a browser
- Accessible from the WAN
- Accessible from non-PC devices e.g. mobile phones
- Ideal interface for services

Why NetTalk?

- Low resource requirements
- Written in Clarion, so you can use your existing Clarion skills
- Included templates get you going fast

- No dependencies on IIS, Apache etc
- Easy deployment – a simple .exe

Some considerations

- Requires a Windows based server (because it is a Clarion program)
- You develop a proprietary program - so your ISP will probably not want to run it
- So you probably need your own machine and host it yourself

Your web server that you develop using NetTalk

- Listens to one or more ports (one program per port)
- Receives a request (GET or POST)
- Processes the request and sends a reply
- Uses HTTP or the secure version HTTPS

There are some fundamental differences from standard Win32 programming that you need to get your head around. For example, in a Windows program each user has their own copy of the exe in memory, and each therefore has their own copy of Global variables. On the Web many users share one EXE and globals belong to all users. This is a vital difference - other methods that use the application broker do it differently, as in that case each user has their own instance of the program.

In a windows program, if the user goes from a browse to a form, then closes the form, they are automatically back on the browse; on the web, users can jump all over the place - they don't necessarily go "back." To give the user the appearance of having their own data, NetTalk uses sessions each with a unique id, which gives a set of data that "belongs" to a user. Each request that comes into your server spawns a new thread that processes the request and closes down - all in about a hundredth of a second.

In terms of performance, sometimes people ask "How many users does it support simultaneously?". This question has meaning with Application Broker types of programs but not with NetTalk, which has no concept of a user. A more meaningful question is how many page views per second can it support - and this will depend on a number of factors including bandwidth, the CPU you are running on, the type of pages being served and so on. Internally Capesoft have tested it at 115 pages per second but generally bandwidth would be a limiting factor before the CPU - so you can usually use a pretty average box to run it on.

Your application serves HTML and Javascript - you don't need to know much about either, but a little helps as you get more involved with advanced features. You can use frames, which is good for apps but not so good for search engines like Google indexing your site, and not so good for people bookmarking pages etc.

NetTalk supports all the familiar Clarion programming components like menus, browses, forms, reports etc. Reports are displayed in PDF and can make use of C6's inbuilt functionality or use PDF-Tools (from Tracker Software Products).

NetTalk works with C55 but there are various limitations (including threading issues) that make C6 a better choice.

NetTalk uses control tags which are embedded in comments, and these look something like:

```
<!--Net:BrowseMailboxes-->
```

These tags are how you establish interaction between the HTML and your program.

Bruce then showed an app which had very little hand-code, being mainly template generated. The templates looked to be fairly rich and well thought out, although there is a long list of "nice to have" features still in the pipeline.

The demonstrated program showed different styles of tabs on forms, and browses could use the familiar (for windows users) highlighting of lines rather than having to use check-boxes to select a record. A tab in the templates allowed you to embed your own HTML (typically from an external HTML editor like DreamWeaver or FrontPage) and the templates do all the necessary conversion such as wrapping and doubling up of quotes etc.

Capesoft have integrated their Insight Graphics into the product and some of their users are pushing for integration with SecWin, Hot Dates and other Capesoft templates, so these will happen over time.

All in all it looked a very good product - so what more can I say other than I purchased a copy!

Debugging - Russell B. Eggen

Russ is known to many of us "down under" as a prodigious poster on the various newsgroups, and author of the excellent "Programming Objects in Clarion" book so it was good to finally meet him in person. He did two sessions on debugging.

A quick show of hands indicated that only about 20% of attendees used the Clarion debugger.

Some used simple `Message ()` or `STOP ()` statements, while others used a logging system or the `OutputDebugString` API call.

Russ has an `OutputDebug` class (which I think he said was originally developed by Skip Williams and then Mark Goldberg) which he will post to his [web site](#) soon. (Editor's note: You can read Skip's article on the debugging class in [ClarionMag](#))

Russ said the first thing to ascertain was whether it was your bug or not. Could it be a bug in Clarion or a third party product you are using? (Mind you a lot of programmers tend to rapidly jump to the conclusion that it must be someone else's bug so be careful that it is definitely not your own bug before you start casting aspersions!)

Also make sure that it is not a data problem before you waste time looking for a non-existent bug.

Russ says that he sometimes meets programmers who say "I don't code bugs so I do not need a debugger" - to which he responds "If you don't code bugs then you are not coding hard enough!" This sounded to me to be in keeping with the attitude of David Bayliss in his [Zero Defects => Zero Productivity!](#) article.

Another trap to watch out for is where the code is correct but the design is flawed, so the output is not correct.

Sometimes a new version of Clarion will break existing code that used to work fine. You need to investigate why. Debugging is a form of investigation - you observe and question what is going on.

Russ thinks the debugger is an excellent tool and easy to use (once you know how). However he said there are other methods like using `DebugView` or driver logging and it is a matter of what is appropriate for the task.

The first Clarion tool to have a real debugger was C3 (DOS) which Russ said was very

similar to the Micro Focus COBOL debugger with which he was familiar.

C6's debugger only does 32 bit so if you are still coding 16 bit apps then you will need to use C55. (But how many people would still be creating 16 bit apps??)

Russ then proceeded to take us through a debugging session. Obviously, first you must make sure that you have unchecked the "Build Release" option and gone for the Full debug mode.

When you fire up the debugger there are four windows. These are the Source window, the Globals, the Procedures and the stack trace. The top left window (the source window) has buttons G "Go" to run your code, B break to set a break point, S step assembler, O step over an assembler instruction, T step source, E step over source, C go to the line where the cursor is and L to locate lines.

Russ normally steps at the source level rather than watch each assembler instruction which would be rather time consuming.

Most debug sessions should be short and sharp - perhaps only three or four minutes long.

You can do the usual things that debuggers do - view the contents of fields (and alter them), stop at certain points and so on.

One limitation is that you cannot do conditional breaks - but you can get around this simply by writing a few simple lines of code that can't alter your program's behavior:

```
? IF SomeConditionToTest
?   junk" = 'XXX - set break point here'
? END
```

Russ mentioned the tip of using XXX as then your code would be easily found in the debugger's search feature.

You should shut down the program being debugged (the "debugee") before you shut down the debugger.

Russ wrote the chapter on the debugger (Chapter 13) in the User Guide and it cost him a few beers with the developer (Ole) who wrote the debugger in order to get all the details.

The second session Russ did was on Post-Mortem debugging which is when your program GPF's and you want to track down why.

GPF's are your friend - they are *not* the problem but simply *report* the problem, so don't shoot the messenger!

To debug GPFs you must install the Clarion debugger as the system debugger. (This assumes you are on your development machine - if you are having the GPF's at a client's site then perhaps Capesoft's GPF Reporter is the way to go.) Russ mentioned that you can also use a debugger chooser - and mentioned [PickDBG](#) written by Paul Attride who now lives in Phoenix, Arizona.

I notice that Russ has already written an [article](#) that covers post-mortem debugging so I won't repeat it all here.

If you have a run-away process – your code is probably stuck in a loop - and don't have Capesoft's BreakIn, you can bring up the task manager and get the PID (process id) of your task (you may need to make this column visible) and then open a DOS box and run C60DBX -pnnnn where nnnn is your PID.

Another tool mentioned was [DebugView](#) which is a free utility from [Sysinternals](#). You can tell DebugView to only show messages starting with certain characters.

Russ's class (originally written by Skip Williams) is called DebuggerClass - deliberately spelt with a single 'g' so as to not be confused with the debugger.

You simply say `DB.message('whatever')` rather than using `MESSAGE('whatever')`, which Russ strongly disapproves of! Russ has a global extension template for this that he made pretty using many of the new screen options that came with the template language in C6. The class also has an option to say

```
DB.debugbreak( )
```

which simulates a GPF so you can get into the debugger at a certain point in your code.

Russ then mentioned programs which just disappear without GPF'ing. These can be hard to debug but it could be a recursive call that has gone haywire and blown the stack. Russ also said to never do recursive calls between different DLL's - you are just asking for trouble. Another possible cause is where you use invalid string slicing e.g. `Mystring[7`

: 5] which you might do if you get confused with the SUB() function. Finally, returning from a procedure and not providing a return value when it is expected may be another cause.

Next Russ talked about tracing driver calls. The bin folder has a trace utility for drivers but this will slow down apps and create big log files. It is better to programmatically turn profiling on and off:

```
Customer{prop:profile} = 'customer.log'
Customer{prop:details} = TRUE
This is all documented in the online help.
Customer{Prop:Log} = 'whatever...' ! is put in log file
Customer{prop:profile} = '' ! turn off profiling
```

In summary, Russ gave us enough information to get in and make use of the debugger when necessary - without the fear that some of us previously held. He made it look easy.

XML in Clarion 6 - Jono Woodhouse

After Bruce's presentation earlier in the morning, it was Jono from Capesoft who took to telling us about XML, and specifically its use with C6. A quick show of hands indicated about half the attendees had used XML in their systems.

XML (Extensible Markup Language) is a standard for storing and transferring data - somewhat similar to HTML but with a stricter syntax. C6 can read and write XML.

XML advantages:

- Cross language and multi-platform
- Both human and machine readable
- Self documenting - field names as well as values
- Requires less set-up
- Adding/deleting columns is easier than with a CSV file

XML disadvantages:

- Not multi-user for writing
- Bloated size when compared to CSV
- The whole document must be written when saved

- Can be slow for big files which require lots of memory
- Can get complex quickly

For storing data there are lots of alternatives (ISAM files like TPS, RDBMS's, ASCII files like CSV or XML) and each with their own characteristics and the trick is to select the tool for the job.

Typical uses for XML include:

- Interfacing to other systems on the fly
- Importing and exporting data
- Creating an RSS feed e.g. product updates

With regards to RSS versions Jono suggested using 0.91 for simple or 2.0 for advanced.

Jono gave a bit of a history of XML starting back in 1969 with GML and flowing through 1989 when Tim Berners Lee came up with HTML and on to 1998 when XML version 1 came out. The syntax checking in XML is more fussy than HTML which is more forgiving - for example tags should be matched.

You use:

- < for <
- > for >
- & for &

CDATA is used to indicate binary data or unencoded text.

Jono showed the difference between using tags and attributes, and pointed to the C6 template option that allowed you to choose which to use. This was followed by a discussion of Advanced XML and DTD schemas (roughly equivalent to your data dictionary), XSLT for nice formatting, XML validators and more.

We went through the C6 templates:

- ToXML
- FromXML
- ViewXML
- And the report to XML report support extension

And some older ones:

- `ExportToXMLCode`
- `ImportFromXMLCode`

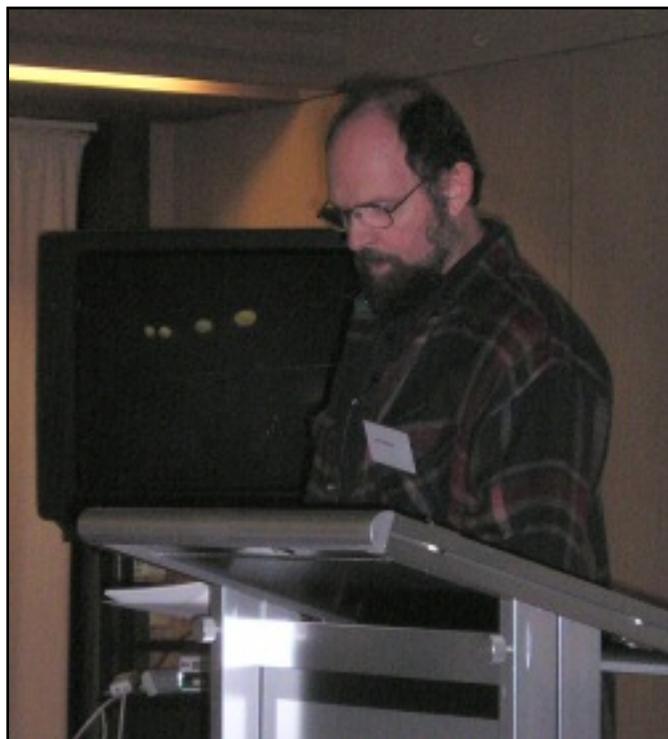
A couple of examples of doing imports and exports and an RSS feed concluded the session.

Installations and Updates - Geoff Spillane

Geoff and his company [Data Down Under](#) are a new distributor of Clarion in Australia. Geoff is hoping to provide a better service and cheaper prices than has previously been the case.

In this presentation he gave a rundown on the requirements of installation software and then gave a demonstration using Lindersoft's SetupBuilder5 (which he also distributes).

First impressions are important - the installation of your software may be the first contact your client has with your system so you want it to go smoothly.



There are lots of requirements for a modern day installer - Geoff listed lots of these and it was no surprise that SetupBuilder - an excellent product written in Clarion by Friedrich Linder, fitted the bill. It is well known in the Clarion community that SoftVelocity distributes C6 and the various patches using SetupBuilder.

SetupBuilder 4 won many awards, and SetupBuilder 5 is much more powerful having a built in scripting language.

Geoff ran a bit short of time, given the numerous functions in the product, but we all got to see the power available in SetupBuilder.

Read the [Day 2 afternoon report](#)

[Geoff Robinson](#) lives near the beach in Melbourne, Australia, and is an active member of the local Clarion User Group. His company, [Vitesse Information Systems](#), specializes in software for local government. Geoff was impressed by Clarion back in the DOS days and grabbed the early betas of Clarion for Windows when they first became available; he has been using Clarion as his primary development environment ever since. When not in front of a computer Geoff enjoys listening to music, singing bass in a local choir, and spending time with his three young children.

Reader Comments

[Add a comment](#)

Clarion Magazine

Aussie DevCon Day 1, Part 2: SQL, New SV File Drivers

by Geoff Robinson

Published 2006-05-12

A Tale of Three Knights - Bruce Johnson

This session was officially titled "Tale of Three Drivers" but Bruce gave us a lighthearted look at three knights trying to slay the dragon and save the maiden in distress.

Each knight (driver) has a different role to play, but the three often work well together.

The various dragons were identified as:

- Bandwidth ("we have more than ever before but never enough")
- Reliability (data corruption)
- Speed
- and perhaps Time (this was a bit vague)

The three knights (new drivers) were

- IP Driver
- Memory Driver
- Dynamic File Driver

Knight 1 - The IP driver:

- Acts as a layer between your application and a server
- Can work in combination with any other driver
- Makes your application Client Server



TPS files are shared files and this works well most of the time, but most of us using TPS files are familiar with data corruption, which can cause much frustration.

In making your app client/server, the client is your program and the server is the data manager DLL and IPREQ service.

Note that this type of client/server is not SQL. You could possibly use this with an SQL database but there is little incentive to, except perhaps for security reasons.

One key benefit of client/server is reduced bandwidth. The server, rather than each client, interacts with keys, and you can have server-side filtering. Perhaps only 10% of the bandwidth is required.

The other major benefit is reliability, as all file operations are done by a single machine.

Note that if the bottleneck is not bandwidth then the IP driver can slow things down rather than speed them up.

Bruce commented on the fact that a developer may be using an old PC for his server and using the latest/greatest PC on his/her desktop where development occurs. In the real world it is generally the other way around - the server is powerful whereas the client PC's can be old and relatively slow - so you need to bear in mind that what you are seeing on your development setup is not necessarily the same as what the client will see. Remember you always need to identify and attack the bottleneck.

Joke: How many programmers does it take to change a light globe [bulb]?

Answer: Why change the globe - it works fine in my office!

Why not just change to SQL?

- SQL is better but it is a lot more work to convert from TPS to SQL than it is to use the IP driver. You should be able to have your IP driver-based system up and running in a couple of hours.
- There is no easy "off" switch with SQL – once you switch, it's difficult to go back.
- Many improvements you make for IP will apply if you later decide to move to SQL
- Changing to SQL and using it just to store the data is a waste of its potential. (you should also move an appropriate amount of processing to the back end etc.)
- IP driver could be an interim solution - which if good enough may become permanent

The IP driver supports multi-DLL apps; you create a data manager DLL. You can also add procedures to the data manager DLL that are called from your application. There is enhanced security similar to many SQL systems in that you can lock down access depending on which user is logged in.

There are considerations about where files actually reside. For example, if you are doing an export you probably want the exported file on the client's PC and not on the server.

Bruce went into some detail about client-side and server-side filtering. The bottom line is that you want to do as much as possible on the server and so you do *not* want to have any of your own functions in a filter statement that need to be done at the client, as that would force the whole filter statement to be done on the client side. So put ranges etc in via the template prompt (which uses the filter statement) so they are done server-side and add any

calls to your functions to the ValidateRecord embed so they are done locally. This is different from straight TPS files, where all filtering is done on the client.

In conclusion Bruce said the IP driver was perfect for LANs, and might help for WANs, but was not a magic bullet. There is no way of knowing what will happen on a connection across the Internet as the performance is out of your hands and may be very inconsistent. It all comes down to throughput and latency (turnaround time). Version 2 of the IP driver has MRP (Multiple Response Packet) and reduced latency so it is better on a WAN than previously.

Knight 2 - The Memory Driver

The memory driver, as the name suggests, store data in memory and not on disk. It is similar to queues but you can use normal file syntax on it. Advantages over queues include:

- It is thread safe whereas (unthreaded) global queues are not.
- You can do set/next
- It works with browse/report templates etc.
- Multiple (unique) keys (queues do not have unique keys)
- Can be included in views

If converting to C6 from an earlier version it is a good idea to convert global queues to the memory driver to get around threading issues. In C6 you can load and save to XML. Version 2 has more ability to prime data and save changes.

Finally Knight 3 - The Dynamic File Driver

The Dynamic File Driver is not really a driver, but it allows you to create structures using other drivers at runtime. It also allows you to change the properties of existing structures (specifically the driver). You can clone structures - for example, to cache some TPS data you could create a clone of the file using the memory driver, for a more flexible approach than just using the memory driver alone.

There are several options for caching including:

- Global caching at start-up, which works great for lookups which have fairly static data
- Caching on one thread only - good for read-only processes and reports
- Caching just one part of a file

One caveat is that dynamic files created at run time are not in the dictionary, and so cannot be used with templates that use the dictionary to get file details.

As Bruce demonstrated, the three knights can co-operate and slay the dragons and save the day!

To SQL or not to SQL - Sean Hennessy

Sean started his session with a paraphrasing of Hamlet - sorry if the photo is not clear - and went on to discuss the pros and cons of using SQL.

Sean gave a little of his history, including programming since 1984, initially in assembler and later in COBOL. He later developed a reservations system in Clarion 2.1 (DOS) and eventually moved it to Clarion 3 (DOS) where he used TPS files. Last year he finally went to Windows and decided to go to SQL at the same time. He said he did not necessarily recommend doing both ports at the same time!

Sean ran through the advantages and disadvantages of both TPS and SQL.

- TPS is fast, but has no relational checking built in (the constraints you put in the dictionary are implemented by the templates but there is nothing stopping you hand coding source code that would violate them).
- TPS is susceptible to corruption from crashes, faulty setup, faulty network cards or cabling etc. SQL does away with these problems (well, almost).
- TPS files are relatively slow to open and close. (Sean found that by using Samba on Linux where opening/closing was faster, he could get a 40% speed increase on one of his apps.)



Sean briefly covered some relational theory from Codd and mentioned that you could think of an SQL engine as "the ABC relationship manager on steroids."

In some ways the advantages of SQL are advantages of client-server in general - they also applied to the IP driver as covered by Bruce Johnson in the earlier session. They include:

- A central data store
- A single point of access
- Data can be cached
- Physical access is local

Unlike TPS the SQL engine enforces your relationships.

Validation raises the question of whether it should be done on the back-end (the server) or the front end (your app). Sean suggests doing both as you can catch problems earlier in your app and give nicer messages but still have the enforcement of rules if someone alters the data through means other than your program.

Next Sean spoke about stored procedures which run on the backend so there is no network traffic - any Clarion procedure without a user interface component is a candidate for changing to a store procedure. Triggers are server-side procedures which are activated when an event occurs on a table, such as on an add, change or delete. These can be simple, like writing to an audit log, or can be complex and could entail cascading triggers. Business rules can be coded here.

There are a few traps for the unwary. If you are using a view, only the fields in the view are returned to your program, reducing network traffic - which is good - but if your program previously used a TPS file it would be expecting the whole record to be current. The trick is to make sure you make hot-fields of any column you need (hot fields are included in the view). Another trap is the use of set(key,key)/next. This is dreadfully inefficient as you are returning everything from that point in the key sequence to the end of the file. So it is better to use PROP:Where or use a filtered view.

Security is an area where RDBMSs excel. You need to login to the server and that will determine what access rights to data you have.

SQL views (defined on the back-end) can be thought of as a select statement which has been given a name. These are handled just like tables as far as Clarion is concerned (at least providing you are not updating).

Finally Sean talked about journaling and mentioned a case where an insurance agent had lost his main data. It was all on his laptop and the last successful backup was eight years ago! He had been told by his IT department that there was no way to recover it (and presumably was told this was a lesson in doing regular backups!). The database was SQL Anywhere, and Sean found there was an old journal file. He was able to go to the old backup and roll forward eight years of transactions in a couple of hours. (At this point one wag sitting near me commented that the bloke had obviously only done two hours work over the past eight years!) Anyway, all was recovered and Sean was a hero.

When looking for a free open source database, Sean rejected Firebird because it did not have this facility available (although you could simulate it using triggers).

Sean concluded that if you had large tables, a large number of users or the need to avoid

data corruption on networks that you don't necessarily control - then SQL is the way to go.

Friday evening restaurant meal

At the end of the first day we all boarded a bus which took us to a local restaurant...







[Geoff Robinson](#) lives near the beach in Melbourne, Australia, and is an active member of the local Clarion User Group. His company, [Vitesse Information Systems](#), specializes in software for local government. Geoff was impressed by Clarion back in the DOS days and grabbed the early betas of Clarion for Windows when they first became available; he has been using Clarion as his primary development environment ever since. When not in front of a computer Geoff enjoys listening to music, singing bass in a local choir, and spending time with his three young children.

Reader Comments

[Add a comment](#)

Clarion Magazine

Aussie DevCon Day 1, Part 1: ClarionTools, PHP, CapeSoft Profiler

by Geoff Robinson

Published 2006-05-11



This year's Aussie DevCon was held from Friday 28th to Sunday 30th April in Geelong, Victoria which is about an hour out of Melbourne on the western side of Port Phillip Bay.

The Geelong Conference Centre (<http://www.salvationarmy.org.au/gcc/>)

was also the venue for the previous two conferences (then called ConVic) and is well suited to a Clarion conference due to its proximity to a major city, and having flexibility in meeting room sizes as well as plenty of on-site accommodation.

There were about 35 participants and a number also brought their partners and children. Many were from Melbourne and

regional Victoria, while others were from interstate (New South Wales, Queensland and

Western Australia) and overseas (New Zealand, South Africa and USA).

The conference was organized by Tony York, assisted by David Griffiths, both from Perth, which is around 3,400km (more than two thousand miles) to the west.

The schedule had a number of optional sessions, mainly where third party vendors like CapeSoft and Clarion Tools showed off their products as well as the "core" sessions.

I'll give a summary of each of the sessions...

Capesoft Profiler - Bruce Johnson

Capesoft's profiler will most probably be familiar to ClarionMag readers due to Dave's [recent review](#) and his use of it in the latest [Clarion Challenge](#).

Bruce gave an entertaining presentation based on an actual interest loan calculation application that had been developed for one of his clients. The client had originally used a spreadsheet which took many hours to calculate a ten year projection.

The Clarion version did the job in about 12 minutes. You would have thought the client would have remained delighted, but after a while they forgot how long it used to take and started wondering if it could be sped up further.

Bruce said that it is often possible to speed things up by a factor of



ten, especially if you do not set the bar too high in the first place!



Optimization often involves making the code more complex and therefore less maintainable. So start simply and only optimize if your client really needs it (and is willing to pay for it). There is no point optimising something that is rarely run or is fairly fast already. For example if a report takes ten minutes to run it might be worth optimising if it is run frequently, but if it is only run once a month then it would be a waste of your time.

Clarion code compiled with debug on is much larger and so it is often thought that it must therefore run slower - but this is not true. Most of the debug code is not normally loaded by Windows, and there will only be a tiny slowdown where there is array bounds checking.

The trick with running Profiler is to profile as little as possible - this is because the profiled code will run a lot slower so you do not want to profile your whole application.

Start with an area that you believe may be slow and then profile that. You can soon see which procedure calls are taking the most time and then you progressively drill down until you get to raw code that you look at optimizing.

As you get closer to the slow code, your simulation will take progressively longer, as the profiler is spending time on the code that is taking the longest to run - so a slowing simulation is a good sign that you are actually making progress!

Bruce gave a number of tips, for example
:

- If an assignment is slow check that both fields are of the same type.
- Use a LONG rather than a ULONG for a counter
- Put AUTO on a field declaration if you will be initializing it in your code anyway
- If file access is slow can you cache the file (perhaps in a queue or by using the in-memory driver)
- If a string field is long then clipping it will be slow, so use an index to keep track of where you are up to in the string and append using string-slicing rather than `clip(mystring) & 'something else'`
- Look for boundary conditions, and if these occur regularly code for them separately. For example you might check if an index was 0 or 1 and do simpler code in that instance.



In the loans calculation application that Bruce was demonstrating he was using the NPV (Net Present Value) procedure in the Clarion Business Maths library, and this turned out to be slow. By doing the code manually he was able to speed things up considerably.

Bruce gave the amusing advice that rather than trying to squeeze the last bit of performance out of an application you should stop when you have sped it up by around 50%. This allows you to do further work on it down the track: "Don't get the glory once when you could get it twice!"

Capesoft had recently done some profiling on their Replicate product and sped it up so parts of it are around nine times faster now (but how much faster could it be? ed.).

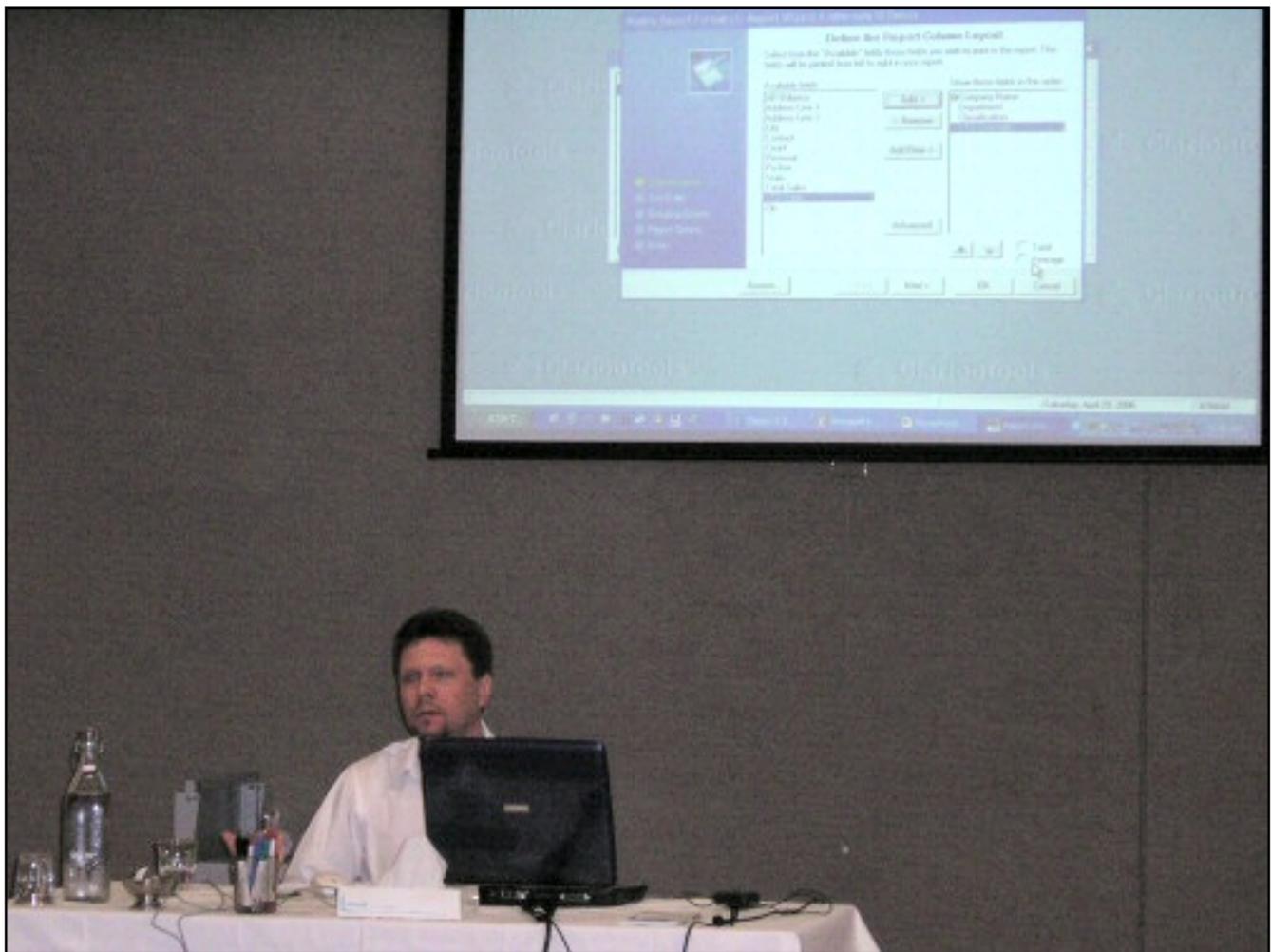
Profiler can handle multiple DLL apps as well as single .EXE's.

In conclusion, Bruce summarized profiler thus: "[It] saves you time, saves you money,

makes you look like a hero".

ClarionTools Wizard - Andy Kaczmarczyk

As many of you know Andy now maintains and develops the suite of Wizard Tools including Query Wizard, View Wizard, Report Wizard, Spreadsheet Wizard and Crosstab Wizard.



Andy did a session on each of these products during the weekend, but I'll summarise them all together here as there are features common to them all. I should be up-front here and point out that I have long been a fan of these products and have them included in my applications.

There is a consistency among the tools so that once the user gets the hang of, say, Report Wizard, they find Spreadsheet Wizard a breeze. The controls and sequences are well laid out and the programs have improved greatly, with new features being added in recent upgrades.

Basically you (the developer) give the user a list of fields to choose from (possibly from many related files) and the tools enable your users to define what they want to see and how they want to see it. The users are then able to name and save their choices for later use.

Anyway, to briefly summarise the tools that Andy went through:

Query Wizard

- The original tool in the series this allows your users to develop sophisticated filter statements for browses, processes or reports simply by stepping through a few screens and choosing the fields to be tested and the conditions.
- Users can create and save their own queries
- Fields can be compared to other fields, constants or expressions (which can be created using a built in expression builder)

View Wizard

- Used on browses to format columns how the user wants to see them
- Users can drag and drop columns, hide them, etc
- Sorting can be achieved through a wizard interface enabling up to four levels of sorting, or can be achieved by clicking on browse headings (this works in Clarion/Legacy as well as ABC)
- Again, views can be named and stored for later use by the users

Report Wizard

- You set up a standard Clarion report with all the fields you want the user to be able to choose in a detail band
- The user then is able to choose fields, sort orders etc.
- You can add breaks with totals, averages, counts etc.
- You can eliminate the detail lines to produce a summary report
- You can call a report wizard from a browse to use the browse's format as the default column layout

Spreadsheet Wizard

- Similar to the Report Wizard except that you are typically exporting to Excel rather than a report.

- You can use an existing spreadsheet as a 'template' and also choose what row and column to start in
- Lots of output options like CSV, HTML, dbf etc

And finally, CrossTab Wizard

- Again your users choose which fields etc and can cross-tabulate data
- As with Spreadsheet Wizard, the CrossTab Wizard makes use of Excel

To get a better idea of how these tools all operate and co-operate you should probably download the demos from clariontools.com and check them out yourself, but hopefully this has given you a brief idea of what they are all about.

At his final demonstration Andy had a competition where he was handing out half a dozen or so "mouse traps". These were actually clever book lights that fold up. Questions tested our knowledge on topics as varied as Ethernet, PC compatibility testing, and episodes of Red Dwarf. It was surprising that none of us knew the answer to the question regarding Bill Gates' middle name (it is Henry in case you were wondering!). The Capesoft lads (Bruce and Jono) had us chuckling as they tried to quite openly Google answers to any tricky questions.

PHP templates - Dave Griffiths

Dave started out by giving a bit of history on how he had started working in the dark ages on punched cards, progressed to writing systems in C calling C libraries, and then later found Clarion 2.1 (for DOS), and what a revelation that had been. Systems could now be knocked up in days and weeks instead of taking many months. After a while his clients started wanting Windows apps, but there was no Clarion for Windows product as yet so he was forced to battle on for a while using Access until CW came along.

He said he felt in a similar situation more recently as clients had started wanting Web applications.

What was he to use?

He considered ClarionNet (not to be confused with Clarion dot Net - more on this in the Day 3 report) but it was not the solution his clients were after. ClarionNet effectively runs your windows app remotely and renders the screen etc on your machine - a little like Citrix - and uses the application broker.

He then considered the SoftVelocity Clarion/ASP and Clarion/PHP templates. His local gliding club used an ASP program but lots of other clients had lots of data on Linux, so Dave decided to use the PHP templates. PHP is widely supported on Linux, Mac and Windows whereas ASP is mainly limited to Windows (Linux ASP solutions do exist, such as [Sun Java System Active Server Pages](#), formerly Chilisoft – ed.). PHP also has lots of free apps readily available on the web.

Dave was able to get web hosting with MySQL and PHP for about \$4 a month, which was another factor.



A quick show of hands indicated about a third of developers present had demand from their clients for Web apps.

Some information on the history of PHP followed:

- PHP originally stood "Personal Home Page"
- It is now a recursive acronym **PHP: Hypertext Preprocessor**
- It is a scripted programming language that can be used to create websites
- PHP is open source
- Wikipedia is developed in PHP

And some information on using the PHP templates:

- You need C6 or higher
- To develop on your machine, you need to get Apache, PHP, and ADOdb (are all free downloads)
- The easiest way to get started is to get the Xampp Apache distribution from www.apachefriends.org/en/xampp.html and simply install what you need.
- Dave used MySQL and needed to get the ODBC driver from www.mysql.com

Like all Clarion developers, Dave would have loved a "switch" - for example to be able to simply add a global extension and *viola!* have his app magically transformed into a web app. Unfortunately things are never that simple. For example in Windows we scroll down a browse, but on the web you do a page at a time and have buttons for next page etc.

In the case of the PHP templates you do not convert an existing app, but start a new app using the same data dictionary. Dave suggested you look at the SV example pages and see what they do.

Basically you have a global PHP template on your app and then a template on the browse which allows you to specify colours for text etc.

SoftVelocity are still providing updates. Version 2 is out and includes code for logins etc.

Dave had some gremlins in his computer so unfortunately we were unable to see all of the presentation he had prepared, but he said that while the PHP templates had not done everything that he had hoped, they were certainly a good start. You can get a simple application up and going without knowing much about PHP - perhaps similar to a Clarion newbie generating a basic application without knowing much about Clarion yet.

The resulting app uses the familiar browse/form paradigm and you can use stylesheet options from within the template.

[Geoff Robinson](#) lives near the beach in Melbourne, Australia, and is an active member of the local Clarion User Group. His company, [Vitesse Information Systems](#), specializes in software for local government. Geoff was impressed by Clarion back in the DOS days and grabbed the early betas of Clarion for Windows when they first became available; he has been using Clarion as his primary development environment ever since. When not in front of a computer Geoff enjoys listening to music, singing bass in a local choir, and spending time with his three young children.

Reader Comments

[Add a comment](#)

Clarion Magazine

Aussie DevCon: Russ Eggen on C7 and Clarion.NET

by Geoff Robinson

Published 2006-05-09

Editor's note: Russ's presentation was on the last day of the conference, but I'm presenting it out of sequence due to the high level of interest in C7 and Clarion.NET. The remainder of our Aussie DevCon coverage will appear in chronological order, this week and next.

Russ Eggen gave us an update on SoftVelocity's progress; prior to coming down to Australia, Russ had a long (about 90 minute) phone call with Bob Zaubere.

We were told that previously it had been SoftVelocity's intention to release C7 first, followed shortly after by the .Net product. This was no longer the case, and the initial versions of both would be released simultaneously. Someone asked if this was simply another way of stating that C7 was delayed and the answer was "yes".

While these are separate products they do share a common new 32 bit IDE and new report and window formatters. The IDE is based on SharpDevelop 2.1, described as "bleeding edge."

However, the IDE was greatly customised by SV and was more like Clarion's IDE rather than SharpDevelop. (Sharp Develop is an open source development environment for .NET that SV has licensed for their own use rather than developing one themselves, or creating a plugin for environments like Eclipse or Microsoft's Visual Studio).

The new IDE has unlimited undos and column copying (which someone pointed out had been available years before in the DOS versions) and all the goodies that would be expected in a modern editor.



Different versions of Clarion can be used by the new IDE so, for example, you could have one project in C55 and another in C6 and another in C7. Having said that however, SoftVelocity is at pains in this release to make sure that code that works in C6 moves over to C7 without any problems. Any fixes being done in C6 are being kept in synch in the C7 code. This is being done so there will be no reason not to move to C7 - it will be much easier going from C6 to C7 than moving from C55 to C6. It only makes sense that SoftVelocity wants everyone to upgrade to the latest versions.

Someone asked how the Core Subscriptions Programs (CSP) were going and Russ said they had been widely taken up by the Clarion community and SoftVelocity was financially

very solid.

Russ mentioned that there had been requests since about 1997 for the ability to know when you had page breaks in reports; Bob Z said he would give this request to the report team for consideration. Russ has also put in various requests for enhancements to the Windows debugger and is hopeful these are coming - but not in the initial alpha release. The .Net product will use a completely different debugger.

The name of Clarion.Net may change as Bob Z is concerned that it is too similar to another product - ClarionNet - and this may cause confusion. Bob had been thinking of Clarion # (Clarion Sharp) but Russ had jokingly suggested Clarion++.

Personally I don't think it will be Clarion++ as while we can increment counters

```
x += 1
```

rather than

```
x = x + 1
```

we cannot use

```
x++
```

(unless that is changing in the language like the ability to use a dot instead of 'END').

The first release of both C7 and the .Net products will be hand-coder's releases and will not have support for the APP and DCT files – that comes later.

Russ was asked if he was sure that this applied to C7 as well as the .Net product and he said yes - both products. He also stressed that this was only for the initial release and that full APP and DCT support will be available in the Gold versions of both products.

For the initial release you could simply take your C6 CLW code and export a project. This can then be compiled and tested in C7 to get more modern looking programs. Some Clarion users are apparently 100% hand coders and will not miss the generation of code. (They must surely be a tiny minority as I've yet to meet one - but I'm told they are definitely out there...)

Asked the "when" question, Russ responded with the familiar "soon™" response. No further information about time frames was available.

Those who had CSP subscriptions will get the Gold release of C7 - no matter how long that takes. Clarion.Net is a separate product and is not included in CSP. It is not available for purchase yet (although Ray Creighton was lucky enough to win a copy later in the day!).

Batch compilers will no longer be needed as this capability is built in, and you can compile in the background while working on other apps.

Other changes: Classes no longer have to be in libsrc; A new template dialog builder will most likely appear in a later release; The Report Writer has a better interface and various other improvements.

One of the things that has slowed SV down is that they have migrated to .Net framework 2.0 from 1.1. This involved a lot of work, but SV did not want anyone to accuse them of using obsolete technology!

Editor's note: Bob Z has said that the C7 hand coder's edition should more accurately be called the C7 hybrid edition. The next release of C6 will have templates that conditionally generate C7 code for all the new nifty UI stuff. You'll generate your code in C6, but make it in C7. Although you'll be able to hand code for C7, you won't have to; just prime the C6 template with the desired switches.

[Geoff Robinson](#) lives near the beach in Melbourne, Australia, and is an active member of the local Clarion User Group. His company, [Vitesse Information Systems](#), specializes in software for local government. Geoff was impressed by Clarion back in the DOS days and grabbed the early betas of Clarion for Windows when they first became available; he has been using Clarion as his primary development environment ever since. When not in front of a computer Geoff enjoys listening to music, singing bass in a local choir, and spending time with his three young children.

Reader Comments

[Add a comment](#)

Clarion Magazine

The ClarionMag Blog

Get automatic notification of new items! [RSS feeds](#) are available for:

XML [All blog entries](#)

XML [All new items, including blogs](#)

Blog Categories

- » [All Blog Entries](#)
- » [Clarion 7 Clarion.NET](#)
- » [Future Articles](#)
- » [News flashes](#)
- » [Nifty Stuff](#)

SV adding C7 template options to 9053

[Direct link](#)

Posted Wednesday, May 31, 2006 by Dave Harms

Bob Z has [blogged](#) that 9053 is in the hands of third party vendors for compatibility testing. This release includes template options to generate C7-compatible code.

Specifically the templates (both ABC and Clarion) contain options to configure the new UI features for Menu's, Sheet/Tabs and Tabbed MDI styles. We included the new template options in this release so that when Core Subscription users receive the Clarion 7 tech preview release they'll immediately be able to make use of them, and in preparation for moving .APP files into version 7.

That is, in the first C7 release you'll still create APPs in C6, and generate the code, then compile the generated code with the C7 compiler to take advantage of the new UI features in your applications.

Reader comments are working again

[Direct link](#)

Posted Thursday, May 25, 2006 by Dave Harms

Nardus Swanevelder pointed out to me that a comment he'd posted did not show up on the article. I did a little digging and discovered that a code change I made in early May in a completely different area of the server app resulted in an unhandled exception in comments, with the result that any comments posted after May 1 have gone into the bit bucket. My apologies! Comments are working again, so if you posted after May 1 please do so again. Thanks.

Replacing Notepad with TextPad

[Direct link](#)

Posted Tuesday, May 16, 2006 by Dave Harms

There's a thread in comp.lang.clarion (How to store or treat this file-Blob?) which has turned into a discussion of the [TextPad](#) editor. Lee White posted this message:

Worth the money. I've been using it since v1.2!

I even wrote a wrapper for it in a local compiled Clarion app named Notepad.exe so TextPad is used everywhere - even apps that have the call to Notepad hard coded!

All it does is take the command line argument and crank up TextPad. The tough thing was getting the original Notepad.exe replaced. There is an order you have to use to get it done.

Delete the prefetch file. Replace the copy in dllcache. Replace the base copy(or copies).

Windows will inform you that something is awry but once ignored it's fine.

Here's the source...

```
PROGRAM
MAP
END
Cmd CSTRING( 256 ), AUTO
CODE
Cmd = CLIP(GETINI( 'NotepadTextPad', 'Path', '' , ))
IF Cmd
  Cmd = '"' & Cmd & '"' & CLIP(LEFT(COMMAND( ' ' )))
  RUN( Cmd )
ELSE
  MESSAGE( 'Path to TextPad was not located!' )
END
RETURN
```

Yes, the path is maintained in win.ini, so sue me!<g>

-- Lee White

Nice one, Lee! Note that prefetch is a subdirectory of your Windows directory, and dllcache is a hidden system directory under system32. If you don't want to display system folders in Explorer, you can always navigate to dllcache via the ol' DOS prompt. And the version in prefetch will begin with notepad.exe and end in.pf; on my machine it's called NOTEPAD.EXE-336351A9.pf.

DISCLAIMER: These changes are only guaranteed to have worked up until now on Lee's

machine, and not on the future on Lee's machine, or on anyone else's machine. Make these changes at your own risk! If your computer asplodes, it's not my fault, and it's not Lee's fault.

On the status of C7 and Clarion.NET

[Direct link](#)

Posted Tuesday, May 09, 2006 by Dave Harms

In the SoftVelocity.public.clarion6 newsgroup, Russ Eggen indicated that the slippage from the January alpha release date was due to conversion part of the new code base from .NET 1.1 to .NET 2.0. Several posters questioned Russ on this, saying they had never seen any public confirmation of this from SoftVelocity. This is the text of my reply in the newsgroup:

I'm not aware that SV has ever confirmed this *publicly*. But it's my understanding also, independent of Russ, that while the C7-specific IDE work was done using .NET 2.0, the Clarion.NET work, which was begun much earlier, was initially done with .NET 1.x, because that's what was available. On the original schedule there would have been a .NET 2.0 IDE for C7 and a 1.1 for Clarion.NET. After that initial release Clarion.NET would have been ported to .NET 2.0.

I think it's safe to say that having two different IDE platforms would have been a bit ugly, and SV's decision to migrate the Clarion.NET work to 2.0 before release, so that one IDE can serve both products, is a good one. I think it's also safe to say that the 1.1 to 2.0 port of Clarion.NET (in its current state, whatever that is) took longer than anticipated, but as far as I know that is all done.

I'm certainly hoping we'll see both C7 and Clarion.NET very soon. I've been hoping that for a while, and I know Z has been hoping to release product too. To my mind, there are some clear, macro reasons for the delay from the original alpha release date (if you can call it a release date). And as all of us who have developed software under deadlines know, sometimes it's the little unexpected things that cause the biggest delays.

This week I'll begin putting up Geoff Robinson's coverage of the recent Aussie DevCon, including a report on Russ's session on C7 and Clarion.NET, which will go up later today, so y'all may want to check that out. That's out of sequence - Russ's session was on the last

day, but there's been so much discussion of the subject that I figured I'd bump it to the front of the line.

I'll also be including, with Russ's report, a brief note from Z on the C7 "hybrid edition"...

Successful Selling Tips for the Technically Gifted

[Direct link](#)

Posted Monday, May 08, 2006 by Dave Harms

Here's a [nifty \(and short\) article](#) for all of us techies who are only separated from success by, um, lack of sales.

HT: Brian Staff and John Verbeeten

Free VS 2005 Standard Edition (and some nifty webcasts)

[Direct link](#)

Posted Thursday, May 04, 2006 by Dave Harms

Rhys Daniel posted a [link](#) in the SV comp.lang.clarion newsgroup to a DDJ article titled [Visual Studio 2005: Unstable and Highly Recommended](#). Leroy Schulz responded by nothing that if you click on the page curl in the upper right hand corner of that page, you can register for free ASP.NET training. Attend three webcasts and you will receive:

- MS Visual Studio 2005 Standard Edition (not for resale)
- Partial ASP.NET book
- MS developer security DVD
- MS certification exam discount
- MDSN Professional Subscription renewal pricing

You'll need to run IE6 to view the webcasts. If you've arrived at the above page with FireFox, just right-click on the page and choose View this page in IE.

I'm having a look at the ASP.NET for JSP developers webcast right now, and it's about an hour long.

Aussie DevCon

[Direct link](#)

Posted Monday, May 01, 2006 by Dave Harms

Nice post at [DevDawn](#) on the Aussie DevCon. Look for a more detailed report in ClarionMag this week.

.NET sample code

[Direct link](#)

Posted Monday, May 01, 2006 by Dave Harms

Updated .NET [sample code](#) is now available for Visual Studio 2005 (and remember that the express editions are now perpetually free). All 101 samples are available for both Visual Basic and C#. Why am I posting about MS products in a Clarion blog? Because this is a taste of what's waiting in Clarion.NET. If there's something here that's of interest, you can either port it to Clarion.NET (C# is your best bet for porting) or simply create the assembly in VS 2005 and call it from Clarion.NET.

Of course, you can also call .NET code now from Clarion using the interop libraries, as [explained](#) by Wade Hatler.

Here's the list of samples:

Base Class Libraries

- Changing ACL (Access Control Lists) on files

- Console Enhancements including buffer control, simple animations and colors
- Downloading files using FTP (File Transfer Protocol)
- Network Programming including PING, UDP, and TCP network statistics
- Using the new Stopwatch class to measure time
- Compressing and decompressing files in the GZip format
- Retrieving Drive Information
- Using Generic Collections
- Creating Regular Expressions to parse text
- Building a tracing infrastructure to track errors and application performance
- Controlling cache policies with the new System.Net.Cache namespace
- Creating a process viewer
- Creating a Windows service manager
- Using the data protection API
- Encrypting and decrypting data
- Using platform invoke to access legacy or unmanaged code from within managed code
- Building in support for localization
- Using the NGEN utility
- Using the new SecureString class
- Using new socket class methods
- Using serial ports with the new Port class
- Working with the file system
- Working with threads to manage multiple tasks simultaneously
- Building a simple web server with HTTPListener

Data Access Samples

- Asynchronous Queries
- Attaching a database with your application
- Creating and using User Defined Types with SQL Server 2005
- DataReader vs. DataSet comparison
- DataSet and DataTable Enhancements
- Performing Batch Updates and Data Paging
- Performing Bulk Updates
- Reading and Writing Images from a Database
- Using Factory Classes
- Using Managed Stored Procedures and User Defined Functions with SQL Server 2005
- Using Multiple Active Result Sets with SQL Server 2005
- Using Notifications with SQL Server 2005

- Using the XML data type with SQL Server 2005
- XPath and XSLT Transformations Enhancements

Web Development Samples

- Using SQLCacheDependency for caching
- DataBinding using the ObjectDataSource, SqlDataSource, and XmlDataSource objects
- Create a Master/Details view using the DataGridView and DetailsView controls
- Basic Master Pages, Nested Master Pages, and Browser-specific Master Pages
- Using the new membership controls including the LoginStatus, LoginName, LoginView, CreateUserWizard controls
- Using the Menu and SiteMapPath controls
- Using Profiles to store user properties
- Using the Membership and Roles API for custom roles
- Using the TreeView control dynamically and responding to TreeView events
- Creating a custom portal

Windows Forms

- Running Asynchronous Tasks
- Storing Custom Client Settings
- Creating Master/Detail Forms
- Playing, Looping, and asynchronously using Sounds
- Using BindingNavigator for static and dynamic data
- Using BindingSource for static and dynamic data including objects
- Extending ClickOnce to programmatically check for updates
- Customizing the DataGridView control with different column control types
- Using LayoutPanels
- Using the MaskedTextBox control for built-in data types and custom data types
- Controlling a simple rich texteditor using the Menus, StatusStrips and ToolStrips controls
- Using My for application, computer, settings information and more. Note: This sample is only available for Visual Basic
- Using the SplitContainer control
- WebBrowser control basics and using the WebBrowserControl

HTMLDocumentModel for Form-to-browser two-way communication.

Visual Studio Tools for Office

- Creating custom menus in Outlook
- Creating smart tags in documents
- Creating custom toolbars
- Data islands and caching
- Integrating Windows forms into Outlook
- Using bookmarks in Word
- Using managed controls in Excel
- Using VBA and VSTO together
- Working with charts in Excel
- Working with contacts and tasks in Outlook
- Working with InfoPath 2003
- Working with named ranges in Excel
- Working with the In Box
- Working with XML in Excel
- Working with XML in Word
- Using the List object

Tablet PC

- Utilizing the back of pen erase feature
- Creating a content enabled form
- Working with drawing attributes
- Enabling gestures
- An ink-enabled text box
- Creating power aware applications
- Simple text recognition
- Speech recognition

Compact Framework Samples

- Creating a setup CAB
- Creating a data-centric application
- Creating a data-centric application (part 2)
- Using the date/time picker feature in the month calendar

- Document list
- Embedded resources
- Handling the screen orientation
- Handling the hardware button screen orientation
- Help and notifications
- Registry sample
- Creating a SQL CE result set
- Using Microsoft Message Queue (MSMQ)
- Web browser sample

Get the source at [MSDN](#).