

Clarion Magazine

This edition includes all articles, news items and blog posts from November 1 2010 to November 30 2010.

Clarion News

[Read 17 Clarion news items.](#)

The ClarionMag Blog

[Read 8 blog entries.](#)

Articles

[ClarionLive DevCon Day 1 Report](#)

November 3 2010

Dave Harms reports on the first full day of the ClarionLive Denver DevCon, with photos, more screen shots from the keynote and some further Clarion.NET information.

[ClarionLive DevCon Day 2 Report](#)

November 8 2010

Saturday's sessions covered an incredibly wide range of topics, from C7 productivity to SQL to .NET to the business of Clarion development. The day concluded with a Saturday panel discussion on future proofing your Clarion development. Can you move your existing application forward to .NET? The panel came to a conclusion that will be a shock to some and a relief to others.

[ClarionLive DevCon Day 3: Final Sessions](#)

November 16 2010

The final (half) day of the ClarionLive DevCon included sessions on Thin@, Software as a Service, and writing Clarion# code. The official schedule concluded with SoftVelocity president Bob Zaunere, but there were some unscheduled awards handed out after that. Dave Harms reports and offers his impression of the conference, which in a nutshell is "it

rocked".

Fooling the FileManager

November 17 2010

In a standard browse or form procedure, the FileManager handles opening and closing files as needed. But what do you do when you want all that standard file handling except the initial file open? As Dr. Parker explains, sometimes the easiest thing is to do an end run around the generated code.

Source Code Library October 2010 Update Available

November 26 2010

The October 2010 update is now available to Clarion Magazine Source Code Library subscribers. Log in to the My ClarionMag page for the download link.

Review: Thin@ Transforms Clarion Desktop Apps Into Software-as-a-Service Internet Apps

November 29 2010

David Harms reviews Thin@, a "smart client" technology that lets you run your Clarion application on a server and deliver only the application's user interface to the user across a LAN, an intranet, or the internet.

ClarionMag Tip of the Week #1

November 30 2010

Dave Harms kicks off ClarionMag's new Clarion Tip of the Week feature with an easy way to keep your DCTs handy.

Creating SQL Applications With Firebird

November 30 2010

Clarion Magazine continues its series on popular SQL databases with a look at how to install, configure, and use Firebird.

Copyright © 1999-2010 by CoveComm Inc. All Rights Reserved. Reproduction in any form without the express written consent of CoveComm Inc., except as described in the subscription agreement, is prohibited.

Clarion News

Lodestar Software Takes Over CPCS

For fifteen years Larry Teames' Creative Reporting Tools has been the de facto standard in Clarion report templates. And after fifteen successful years at the helm, Larry's ready for even more fishing and relaxation. To ensure that the report templates continue to get the same high level of service and support, Larry has arranged for Lodestar Software to take over the complete product line. Lodestar Software is run by Lee White, the creator of the very popular Report and Presentation Manager (RPM), the definitive toolkit for getting the most out of your reports after they're created. Together RPM and Creative Reporting Tools are a complete end-to-end solution for designing and managing reports. These two products already have good support for each other, but in the not too distant future the integration will be improved and become seamless. Creative Reporting Tools and RPM: a truly complete reporting solution. To ensure a smooth transition, and the same quality of customer service CPCS users have come to expect, Larry Teames' expertise will be available to Lee White should the need arise.

Posted November 2 2010 ([permanent link](#))

Convert Your Clarion Application From TPS to SQL

The latest version of DMC can convert your application from TPS tables to any one of a wide variety of SQL structures. You don't have to buy DMC before you see what it can do for you. For \$199 DMC will take your Single EXE/DLL (chosed one APP in your solution) application in Clarion 6 or Clarion 7 and convert it using DMC (with all predefined settings) on your machine. DMC will help you install Microsoft SQL 2005 or 2008 if necessary, or we can use our SQL server for testing the data. When DMC is done you'll have a compiled application (or APP if multi-DLL) with your data converted to SQL (if your tables are huge this may take some additional time) that performs nearly all your functions. Some code that works just fine with TPS files will not work with SQL. DMC will fix as much as it can (and will help also), but you may have a little work to do when the conversion is done. Even after it all works, you may want to further polish your application for better performance and lower bandwidth. Guarantee: If DMC can't make your application connect to SQL (local and/or distant), with at least part of your data converted (all if you don't swamp us with massive tables) and you have more than 100 errors at the end, you get your money back.

Posted November 12 2010 ([permanent link](#))

DMC 2.3.0.1193

DMC 2.3.0.1193 is now available. Features include: Port your Clarion Application to SQL (with full SQL code compliancy and Data Transferring option) *Clarion 5.5/6/7 - ABC and Legacy*; Full Support for MIMER SQL; Advanced level Capitalizing during Data Transfers (define always uppercased - always lowercased and direct replacements "cases"); During a Clone Table to SQL you have the possibility to apply SQL normalization settings (as per your defined options); When any Task is opened or running the Main Frame Close Application button will popup a message to Close the Task; Corrected several bugs in existing code (Identity on SQL Tables - Colons in Column Names etc...).

Posted November 12 2010 ([permanent link](#))

DMC Videos

New DMC videos are available for you to stream - download, including: Porting the SV demo INVOICE application to a FULL DUAL TPS-SQL application; Porting the MULTIDLL SV demo DllTutor application to SQL (steps only).

Posted November 12 2010 ([permanent link](#))

CHT 14D1.00

CHT's 4th quarter Build 14D1.00 for C7 is now available for download via Webupdater. The C6 version of 14D1.00 will follow in a few days. All CHT forward work is being directly in C7 so that when the C7 work is done the C6 code needs to be verified.

Posted November 12 2010 ([permanent link](#))

Dr. Explain 4.1

Indigo Byte Systems announces Dr.Explain 4.1, an important upgrade to its state-of-the-art documentation authoring tool which is now also available in 8 languages: German, French, Spanish, Italian, Portuguese, Sweden, Russian, and Dutch. Now, users can read the information on the website and use the program in their native language.

Posted November 12 2010 ([permanent link](#))

iQ-XML 2.72

iQ-XML has been updated for Clarion 7.2. This change allows both DLL and LIB (Standalone EXE). There are two LIBs now being distributed during the installation: iqxml70.lib / iqxml70.dll (DLL mode) and iqxml70l.lib (Standalone EXE). Changes include: Fix to possible trailing blank record on XML:LoadQueue; Fix when using XML:DebugMyQueue - the first record is now selected rather than no record; Local Lib compiles now should be valid. Using IQXML70L.LIB for local compile. Compiled using Clarion 7.2.7600.

Posted November 12 2010 ([permanent link](#))

FullRecord 3.00 Beta 1

New in FullRecord 2.00 Beta 1: Option to save the full record, only the changed fields, or a mix of both. Why? In previous versions of FullRecord, some users complained about the size of the audit file. Yes, it is true it becomes big. That's the price of having everything audited. However FullRecord is now able to work in three modes: 1) Full record mode: everything is saved, only one record per i/o operation as it was always. In fact, if you use this mode, you need to do no changes, and the files and programs works the same as in FullRecord 2.25. The problem with this is the size of the complete record is saved even if you change only one field. The advantage is that if you change, say, 50 fields, only one record is saved (which gives faster i/o). 2) Changed fields mode: Only the changed fields are saved. A record is saved per field changed. So if you change only one field, only one small record is saved. If you change 50 fields, 50 records are saved (50 i/o, but smaller than the whole record). The problem could be the performance if you change many fields often. The advantage is that the value of changed fields are saved in blobs and the audit file size is the smallest possible. 3) mixed mode: If the number of changed fields is beyond a number (set by you), the full record is saved as in (1), else each field is saved as in (2).

Posted November 12 2010 ([permanent link](#))

Valid RFP Development

In this article Mike Gorman tells us that if RFP technical specification components are valid, reliable and repeatable, the resulting bids are likely to enable the correct selection of an implementation contractor that will accomplish a correct implementation the first time.

Posted November 15 2010 ([permanent link](#))

DMC 2.3.1.1205

You can update to DMC 2.3.1.1205 from within DMC or you can use the full installer. Changes include: Port your Application to SQL added a Casing setting to retain existing CamelBackCased Names; Port your Application to SQL added a setting to add a new column (GUID or Identity) only if no primary key; Port your Application to SQL added when replacing a STRING to Date to extract and use the DCT PICTURE in Data Transfer; All Tasks performed in DMC are now called from the procedure timer which makes the IDE totally responsive; Clone to CSV added a renaming Destination Table option; Corrected several bugs in existing code (Identity on SQL Tables - Colons in Column Names etc...).

Posted November 15 2010 ([permanent link](#))

Icetips Taskpanel

A new build of Icetips Taskpanel is now available for download. Taskpanel is part of the Icetips Gold subscription.

Posted November 15 2010 ([permanent link](#))

Shawn Mason SQL Seminar Early Bird \$699 Pricing Ends Nov 19

Due to the demand received over the years since his last Clarion/SQL Seminar, Shawn Mason has decided to host the seminar one more time. This one is updated for Clarion 7 and SQL Server 2008 as well as a looks at MySQL and PostgreSQL. Seating is very limited due to special attention that will be given to each person. The seminar will be held Jan 27th-29th, 2011 at the Embassy Suites Buckhead in Atlanta, GA (USA). The price is \$1099 but the current early, early bird pricing is only \$699 through Friday, Nov 19th, 2010.

Posted November 15 2010 ([permanent link](#))

COM Source Generator For Sale

Andy Ireland's COM source generator is for sale. This is an improved version Andy demonstrated at the last ETC event. For those that were there, it was fast and most of the demonstration was generating Clarion code for Excel (about 8MB of source code) in only a few seconds. The new owner will have all rights to the source to use as they see fit. Contact Russ Eggen for pricing.

Posted November 15 2010 ([permanent link](#))

RPM/AFE 25% CLDC Discount Ends Nov 30

The 25% CLDC discount on new subscriptions of RPM and AFE expires at midnight on 30-Nov-2010. If you're interested, and you're on the attendee list for CLDC, send an email to CLDC-at-cwaddons-dot-com and Lee will send you the order form link. Don't use the standard order form since it's full price only.

Posted November 17 2010 ([permanent link](#))

iQ-Sync 1.33

iQ-Sync 1.33 is now available. This update does require a conversion on one of the files, but it is handled automatically. The big change is you can now traverse/include all sub-folders. So, instead of having an entry for each directory, you can create one entry and have it include all sub-folders. Other changes include: You can now include all sub-folders; You can ignore empty folders warnings (Setup); You can configure your Backup Drive/Directory (Setup) so it is easier to add Multiple Folders.; Checking In-Use files is now faster.; Larger files and copies are now supported showing "1.23 GB" instead of "1,230 MB"; You can quickly open the Source or Destination Folder in Explorer with icon button.; You can now configure each entry as a "Complete Sync/Clean Missing" so the system will delete all files on the target that does not exist in the source (ie: a complete copy/clean rather than leaving Orphan files). iQ-Sync is a full copy/backup application to help you make quick backups of your projects, Clarion apps, etc. easily - and it's free.

Posted November 22 2010 ([permanent link](#))

Bob Roos Takes Over Clarion QuickBooks Connect

Paul MacFarlane reports that ownership of Clarion QuickBooks Connect has been

transferred to Bob Roos. Bob has been working hard on CQBC for his own use and he has extensive knowledge of the product. You can reach Bob at qbconnect at the domain wybatap.com.

Posted November 29 2010 ([permanent link](#))

Clarion Third Party Profile Exchange - November 29 2010 Release

Encourager Software is renewing its efforts to bring updated Clarion information in the Product Scope 7 Viewer and web updated data version format.

Posted November 29 2010 ([permanent link](#))

Copyright © 1999-2010 by CoveComm Inc. All Rights Reserved. Reproduction in any form without the express written consent of CoveComm Inc., except as described in the subscription agreement, is prohibited.

The ClarionMag Blog

More CLDC coverage on the way

I've started going through my DevCon notes and photos, and I'll have several more articles in the next couple of days.

John and Arnold put on a terrific conference at a great venue. I heard a lot of positive comments from attendees, and you know it's been a good show when people start asking about next year before this year's event is over.

Stay tuned for more coverage!

Posted November 1 2010 ([permanent link](#))

Do you have C7 apps in production?

Rohan asked in the newsgroup:

Just wondering if anyone has taken the plunge and released any of their C7.2 applications into the "real" world yet?

While the responses didn't make up a huge sample size, the majority of those responding said yes, they had C7.2 (and sometimes C7.1) apps in the wild and running without any significant problems. Robert Paresi is still waiting on a few runtime fixes; I think it's safe to say he stresses the RTL more than most Clarion devs.

Posted November 8 2010 ([permanent link](#))

Getting to the SV discussion server HTML page

Last week the SV news server was down for a time, and when it came back up the URL for the news server web login page no longer worked. Didier Le Duc (and a few others) pointed out that the default port (80) no longer works - that URL takes you to the PTSS login. You need to add :8132 to the end of the url <http://discuss.softvelocity.com> to specify the port number.

I suppose I could have posted the full URL but I don't want to give the bots of the world any free opportunities.

*** UPDATE ***

To use the secure version of the page you'll need to (of course) specify the <https://> protocol and use the port 8133 instead of 8132. And you'll have to add a security exception for the certificate, which as far as I know is a safe thing to do but I accept no responsibility for the consequences!

Posted November 9 2010 ([permanent link](#))

Nifty shipping idea

Harry Bennett of [SyPort](#) posted something interesting in the [softvelocity.clarion.chat](#) newsgroup on protecting fragile items during shipping:

What I have done in the past is to use a small processor board and a tri-axis accelerometer and basically built a data logger to record the accelerations v time to identify where the problem occurs.

This allows diagnosis of when the problem occurred.

That sounds like a lot of work, so if you need that maybe you should contact Harry. He also points out that you can buy inexpensive [shock sensors](#) you paste on the packaging itself and which will change color if a specified G load is exceeded.

Posted November 9 2010 ([permanent link](#))

C7 Addins by Brahn

Check out Brahn Partidge's [ClarionEdge](#) site for some terrific C7 IDE addins. Some are specific to C7 and some are more general. Brahn is also working on packaging the C7 addins into a toolkit.

Posted November 9 2010 ([permanent link](#))

Thin@ review coming up

A draft of ClarionMag's Thin@ review has been sent to the Thin@ folks for comment. I hope to have the review up later this week. Following that, look for a NetTalk review as ClarionMag covers two of the hottest technologies in the Clarion third party market.

Posted November 22 2010 ([permanent link](#))

Clarion Polish User Group Meeting 2010

The 2010 Polish Clarion User Group meeting (read the [English translation](#)) was held on November 19-21, the first time for this gathering in a number of years. The meeting was held near Warsaw, at the Hotel Groman in the Sękokocin Forest.

Clarion Magazine was one of a number of third party vendors to sponsor this event. You can see [additional photos here](#).

Topics discussed at the conference included:

- Codejock templates
- Clarion and Android
- InMemory and Dynamic drivers
- iQ-XML
- Clarion.NET mobile development
- NetTalk
- Migrating to SQL
- Version control

Congratulations to the Polish Clarion community on a successful event!

Posted November 26 2010 ([permanent link](#))

More options in the Clarion smart/thin client space

I've just published [a review of Thin@](#), the smart client solution for Clarion that created quite a buzz at the Denver DevCon. Among other things Thin@ offers a Java client (in beta) that lets you run your Clarion apps remotely on any platform that supports Java (e.g. Mac, Linux).

The smart client space is heating up. ClarioNet's Ivan Mintoff is the featured speaker for the Dec 3, 2010 [ClarionLive webinar](#), and will be presenting something to do with Android...

Posted November 29 2010 ([permanent link](#))

Copyright © 1999-2010 by CoveComm Inc. All Rights Reserved. Reproduction in any form without the express written consent of CoveComm Inc., except as described in the subscription agreement, is prohibited.

ClarionLive DevCon Day 1 Report

By Dave Harms

Posted November 3 2010

Friday was the first full day of sessions at the ClarionLive DevCon. Breakfast was served in a lobby area outside the meeting room, and included scrambled eggs, breads and pastries, cereal, fruit, and more.



John and Arnold welcomed just under a hundred attendees to Denver, with another eleven developers attending online.

The combination of an online and in-person event made for some interesting technical issues. Below you can see the control table with Arnold and John's redundant recording setup.



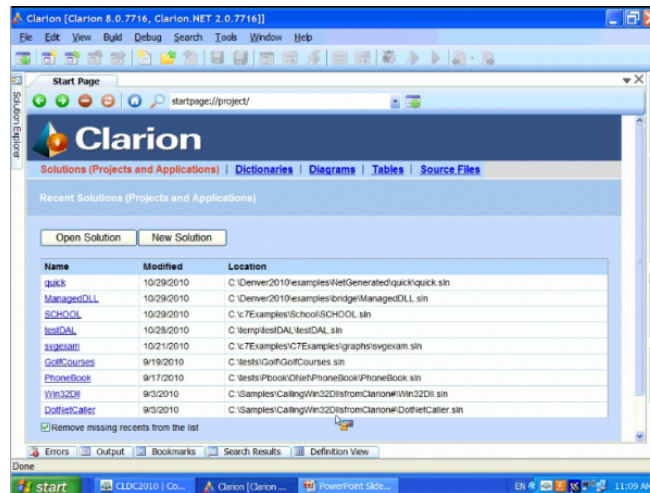
They used Adobe Connect to manage web conferencing and recording, and once it was running it generally ran reliably. But getting it running wasn't always easy. On Friday morning it took the better part of an hour to get the network running (failed router), and then there were some problems getting both video and audio working. It's not that John and Arnold hadn't tested the system; they had, but technology is technology.

The original plan was to have Bob Zaunere, who I'll refer hereafter as Bob Z to avoid confusion with Bob Foreman (Bob F), kick off the conference with a keynote address, but

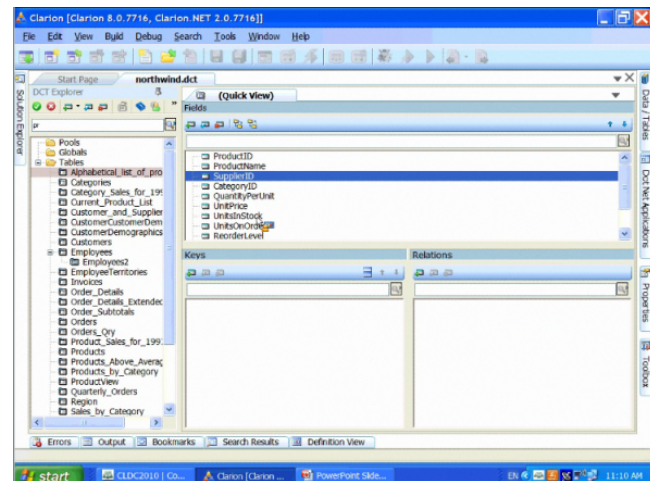
the technological gremlins got out of the box again and we lost audio for online attendees. Z stood by while Bob F started the first of two morning sessions on Clarion 7. This was more of a training session than a "what's new" session, since many if not all of the features Bob F discussed have been in the product for some time. And honestly I missed a lot of what Bob had to say because I was trying to lend a hand with the audio issues, which eventually got sorted out.

Halfway through his session, Bob F stepped aside to let Bob Z deliver his keynote. You can [read the initial report here](#). Some attendees thought Bob Z's delivery seemed a bit scripted, and no doubt it was a prepared speech. But Bob Z couldn't see or hear the audience, which no doubt made the delivery seem even more formal.

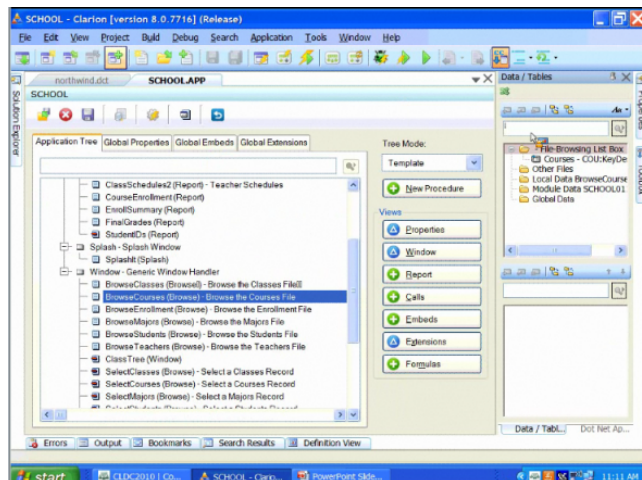
Here are a few screenshots from the keynote.



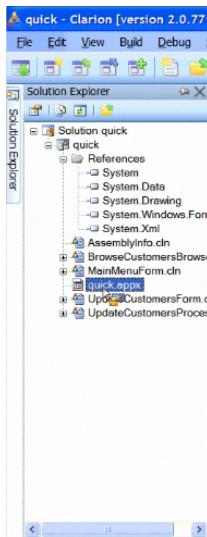
The new start page.



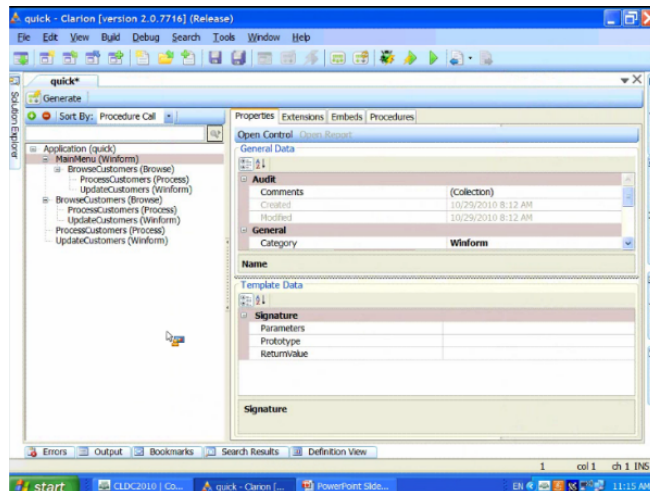
This is a combined build of Clarion 8.0.7716 and Clarion.NET 2.0.7716. Note the locators.



Icons have been prettied up.



.NET APP files have the .appx extension and are XML files.



The .NET AppGen

```
MAP
! to Export for Win32 just need both NAME and PUBLIC
SumDos      PROCEDURE (Int32 A,Int32 B),Int32,NAME('SumTwo'),PUBLIC
SayHi       PROCEDURE (),NAME('SayHello'),PUBLIC
SayName     PROCEDURE (string theName),NAME('SayHelloName'),PUBLIC
END
```

The new syntax for exporting procedures from Clarion# to ClarionWin32

The above screen shot shows the new syntax for declaring functions in Clarion# so they can be used by Clarion Win32 apps, including those created by C7. From a newsgroup post by Diego Borojovich:

There are no changes in the win32 side regarding this so it should work with C63 too. The New functionality added to the Clarion# compiler is that creating a global MAP procedure and adding the NAME('myprocedurename') attribute and PUBLIC to it will (after setting in the project the option) create a Clarion Win32 .LIB and expose the procedure in the assembly. From that point on you just use the LIB+DLL in your Win32 clarion app as it was any other DLL. In the Win32 side you just declare the procedure as EXTERNAL,DLL in the map and link the LIB as you do with any DLL. No C++ wrapper needed (IJW), no COM, no external files, no TCP, nothing, just Clarion code in both sides.

The change in C8 is that if you have both projects in the solution you can add a reference to the Clarion# project to the Win32 project and keep them in sync.

The datatypes supported are numerics(LONG,SHORT,BYTE,ULONG,USHORT) and string/cstring (for now).

After the keynote Bob F resumed his presentation.

Bob Foreman

Bob mentioned that code snippets are not by default available in the embeditor. That's because code snippet behavior is handled by the editor, and the code snippets you have available are dependent on the file extension. That way you can have different snippets for different languages.

The thing is, when you view the embeditor the AppGen is creating a temporary file, and that temporary file doesn't have a .CLW extension. If you want your snippets to be available in the embeditor you have to go to Tools | Options | Coding | Code Snippets, find . cl w; . CLW in the list and append ; APPCLW to the Extensions field.

Mark Riffey later reported that he did a little digging and found the SharpDevelop-templates.xml file (which contains the code snippets) existed in both the \pathtoclari ondi rectory\data\opti ons directory and in the C:\Users\username\AppData\Roaming\SoftVel oci ty\Clari on\7.0 directory. I'm pretty sure the way this works is the Clarion install version is the default, and the one in your AppData directory is only created once you modify the templates, after which the IDE loads your personalized copy of the file instead of the shipping copy.

As Mark points out, if you wish you can add the APPCLW setting directly to the XML file instead of by changing the Extensions field in the Code Snippets options in the IDE. Just open the file and look for this line:

```
<CodeTempl ateGroup  extensi ons=". cl w; . CLW; . Cl w">
```

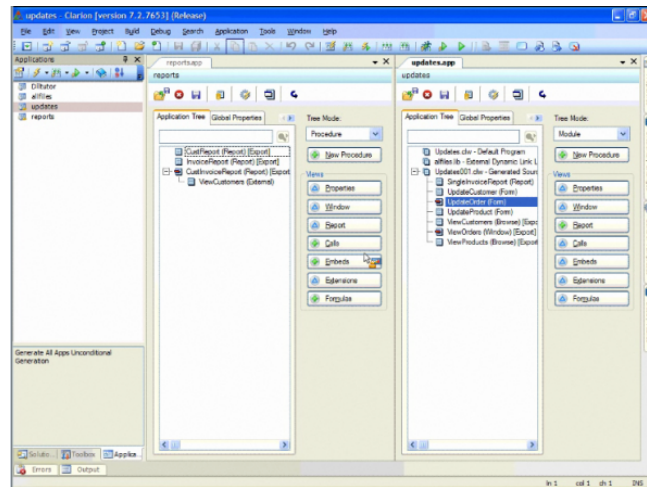
and change it to this:

```
<CodeTemplateGroup extensions=".clw;.CLW;.Clw;.clw;.ClW;.ClW;.appclw;.APPCLW"
```

Also, as Mark points out, the extensions are case sensitive which is why there are variations on the extension.

As has been seen before, Bob was using the combined Win32/.NET build.

Bob spent some time going over converting multi-DLL applications, and pointed out some of the nice IDE features such as the ability to open two applications side by side.



He also covered project options and project properties, commenting on the typical confusion over where you change the actual project mode from debug to release and showing how you can use the project system to organize projects that are related to each other but not necessarily dependent on each other (such as the IP driver application components).

As usual Bob's presentation was informative, although pretty much about the existing product rather than any new features.

Mike Hanson

Mike Hanson had the second session, titled Best Coding Practices. There's been a lot of discussion on this topic in recent ClarionLive! webinars, and Mike did a great job of hitting the high points.

Most of our time is spent reading code not writing code, so the point isn't to write the most incredible code, it's to write code that you can read later. Some of the things developers do wrong are copy and paste code, making minor changes, and use non-descriptive names.



Mike Hanson

Comments, Mike said, are failures. If your code needs comments then the code itself is too hard to understand, and you need to break it down into smaller, more descriptively named parts. Organize your code, and use white space appropriately.

Best practices include refactoring, using naming conventions, and encapsulation.

Don't mix conceptual scope - this was Mike's take on the single responsibility principle. He also talked about how to structure your code using local procedures and local maps. Keep your procedure parameters to a minimum.

Mike went into a more detailed discussion of encapsulation, and not restricted to just OOP. He covered design patterns, which are standardized ways of solving certain kinds of problems. If you have a code design issue, chances are someone has already invented that wheel.

Mike touched on Robert C. Martin's SOLID (Single responsibility, Open-closed, Liskov substitution, Interface segregation and Dependency inversion), test-driven development (TDD) and agile development.

Lunch!

No box lunch for this conference! We all trooped over to the hotel restaurant for an excellent and varied buffet lunch.





I don't have any detailed coverage of the afternoon session. My wife and daughter came with me to Denver, and we'd planned to be tourists for a couple of days before the conference started. That got nixed by a record-breaking storm in the Midwest that delayed us by two full days. Arnold and John very graciously excused me from moderating for the rest of the day so I could have some family time.

Rick Martin

Both of the afternoon sessions were by Rick Martin. His topics included Controlling the ABC Classes, Writing a Clarion Class Wrapper for a C DLL, Debugging in SQL and Cool Tips and Tricks with Useful Classes, Debugging SQL Stored Procedures and Functions, and Cool tips & tricks with useful classes from Rick's Toolchest. I'm not at all surprised to hear he did an excellent job. I saw Rick later that night in the Spotted Dog and he seemed fine, but the next day he was looking decidedly unwell. He perked up a bit again by Sunday, but he went to his doctor when he got home and was diagnosed with pneumonia. The good news is he's on the mend.

Third party presentations

After Rick's session there were presentations by some third-party vendors including Phil Will from ProDomus, and Marko and Daniel who travelled all the way from Croatia to give an impressive demo of their new Java client for Thin@. You host your Clarion app on a server, and they translate it to Java which lets your app run in any browser that supports Java (which is most of them). This means you can have your Clarion app's user interface running on multiple platforms like Windows, Mac, and Linux.

Shawn Mason

Arnold and John believe in packing a lot into a conference. The evening session started at 8:00 p.m. and featured the always entertaining Shawn Mason and his well-received SQL Extravaganza. I got back from my evening off around 10 p.m. and headed down to the Spotted Dog, assuming that John and Arnold would end the evening session with their usual punctuality. Mistake! I should've gone to the session room; the extravaganza didn't end until closer to 11 p.m.

David Harms is an independent software developer and the editor and publisher of Clarion Magazine. He is also co-author with Ross Santos of *Developing Clarion for Windows Applications*, published by SAMS (1995), and has written or co-written several Java books. David is a member of the American Society of Journalists and Authors (ASJA).

Article comments

 [BACK TO TOP](#)

Copyright © 1999-2010 by CoveComm Inc. All Rights Reserved. Reproduction in any form without the express written consent of CoveComm Inc., except as described in the subscription agreement, is prohibited.

ClarionLive DevCon Day 2 Report

By Dave Harms

Posted November 8 2010

Saturday morning kicked off with the breakfast-time application showcase featuring presentations by Bruce Johnson, Gordon Holfelder, Jim Morgan and Mike Hanson.

If you weren't there, then imagine that you walk up to the conference room around 7 a.m. Yes, 7 a.m. You'd better arrive at the DevCon well rested, or you'd best avoid the Spotted Dog after the last session.



There's a continental breakfast waiting for you just off the lobby in the conference room lobby (if you can get past Bruce Johnson and his camcorder, that is. Along with the goodies you'd expect in such a breakfast there's one hot dish; it might be scrambled eggs or oatmeal. You grab a little chat, some coffee and some chow and you head into the conference room. (And if you're not an early riser, or your stomach isn't, or you're part hobbit and one morning meal doesn't cut it, breakfast is served until 11:00 a.m.)



John Rae being interviewed by Bruce Johnson

At 7:30 the Application Showcase begins; selected presenters showing some of their favorite products.

Your Devcon day has begun!

On Saturday morning Bruce Johnson, Gordon Holfelder, Jim Morgan and Mike Hanson all gave short presentations on some of their favorite apps.

Bob Foreman

At 8:30 the conference proper got under way with Bob Foreman's second session. Among other things, Bob took a look at what he called "Clarion's Hidden Treasures".

These included some browse nuggets such as the ability to set browse line height application-wide, sort headers, the list format manager, additional sort orders, higher key components, conditional totalling and browse styles. He reviewed the browse control templates, and urged all developers to use the newer EIP templates instead of the original EIP templates.



Bob Foreman presenting

Bob also talked about the query center template (an alternative to older QBE options), advanced SQL support (sums, aggregate totals, etc.), and blob support. He demonstrated the

BlobInControl extension template as based on a show of hands almost no one is using it.

Other topics Bob touched on included the detailed EIP interface, the resize template, the "anchor and grow" sample app, the report child files extension (another one no one seems to be using), and the help. Although context sensitive help is currently only available in the editor, the help file covers a great many topics. Use it.

Bob noted that the Clarion 8 RTL will have support for HTML help.

Another show of hands indicated that lots of people use the in-memory driver. Fewer use the IP file driver, and hardly anyone is using Report Writer 7 right now. Bob showed his report writer launcher which can be found under the Clarion examples folder.

Bruce Johnson

Arnold had titled Bruce's session "Let's Have Some Fun With Clarion." Now while Bruce is always entertaining (and only getting better, it seems), fun doesn't seem like exactly the right word for his presentation. He spoke on two subjects: development for mobile applications, and how to run a software development business.

Bruce's mobile world

Bruce asked some hard questions about developing for mobile applications. What is a mobile device? How do we know what to program for? Don't confuse mind share with market share. Apple has huge mindshare with its iPhone, but even inside the US Apple doesn't have the majority of market share. Worldwide, Apple's market share is tiny.

Beware of the outlier effect - what you hear about in the news is stuff that isn't happening every day. Someone making hundreds of thousands of dollars per day makes the news because that isn't the norm; it's not what almost all developers will experience.

Your two development choices are web app or native app. Native apps advantages include mind share, access to customers via app stores, access to hardware features, and constant application availability. The cons include a smaller available market, the need to keep with new devices, and restrictions imposed by app stores.

Web app pros include targeting multiple platforms, quick deployment and vendor independence. Cons include appearance (they don't look like native apps), limited hardware access (you can typically get geolocation via the browser), need for internet access, and browser differences (WebKit vs the rest).

Also ask yourself what kind of app you're building? Is it data centered? And if so, will you periodically sync your data or access it in real time over the internet?

Show me the money

Bruce's second session was entitled "If Clarion is so good, why aren't I rich yet?" Bruce's assumption was that everyone is doing the programming right, so this was a session about

the business side of things.



Unless you are rich, step one is to acknowledge that you have a problem. If you don't do that the problem will never get fixed.

Don't value yourself poorly. And value is not the same as price. You set your own value with your hourly rate. If you lower that rate for some circumstance, you're providing extra value.

Don't just charge for the perfect hours. We make stupid mistakes for a living. And it's pretty unlikely you'll bill for all the hours in your day. You may spend half your time doing things you can't bill for, so your rate had better take that into account.

Bruce quoted Mark Riffey, the Clarion world's favorite business blogger (a.k.a. Montana's Seth Godin): "If your business isn't making money while you're away, it's not a business it's a job."

Is your business an accident? Are you planning financially? You're not fated to be poor - you really can change your circumstances.

Manage your cash flow. Even squirrels know to have reserves.

Do your customers know what you look like? The only product you have to sell is customer service. The day the boss stops taking phone calls is the day customer service goes downhill.

I'm not doing a shred of justice to Bruce's material. It was one of the best presentations I've seen at a conference anywhere. It wasn't a whole lot of fun, mind you. But it was absolutely excellent.

Remember, if you weren't at the conference you can still buy the recorded sessions.

Gordon Holfelder

After the midday meal Denverite (Denverian?) Gordon Holfelder showed a set of utility classes for working with SQL database. These classes (which will be made available to attendees) use the dynamic file driver, but can be modified to use queues instead.

Gordon not only uses the classes for queries and updates, but also for managing the database itself. SQL databases contain a lot of metadata, and you can use that to extract the structure of the database and any triggers, stored procedures, etc.

Gordon uses this metadata to manage database upgrades within the product itself. The database management console lets users extract data, rebuild the database structure, and re-import the data. The system supports both MS SQL and Sybase.

During the break many developers headed for the solutions room. I missed getting a shot of Gordon during his presentation, but he's in the center of the pic below, behind Rick Martin (in the green shirt).



Dave Harms

Following the afternoon break it was my turn. My topic was "90 Things Every Clarion Developer Should Know About .NET (in 90 minutes or less)." There's no way to absorb nearly a hundred concepts in that short a period of time, so the purpose of my talk wasn't so much to teach those concepts as to build a bit of a picture of the immensity and complexity of .NET development.

Topics I covered include .NET basics, differences from and similarities to Clarion Win32 development, the major .NET application platforms and significant changes in application architectures in .NET (as compared to Clarion Win32).

I offered some recommendations for those considering .NET development, and suggested that the platform shift from Win32 to .NET is a greater change than from DOS to Windows. I concluded that while there are a great many benefits to mainstream .NET development, we should also be grateful for the capabilities we have right now in Clarion Win32.

Show and Tell

Once again, after the last afternoon session and before supper there were several short presentations by third party vendors. Bruce Johnson and Geoff Thompson did some CapeSoft demos, I showed Clarion Magazine and (briefly) DevRoadmaps, and Tony Tetley gave an RPM/AFE presentation on behalf of Lee White.



Panel Discussion: Future Proofing Your Development

Saturday evening was a highlight for me. I was privileged to sit on a panel with Mike Hanson and Bruce Johnson as we discussed the topic "Future Proofing Your Development." In recent years there's been a lot of anxiety in the Clarion community; it was a long wait for Clarion 7 (admittedly, some are still waiting) and an even longer wait for the Clarion.NET AppGen. Many Clarion developers are concerned about their future development path.

OOP was a major theme throughout the panel discussion. In typical Clarion development the code that we add to our applications, which is the code that for the most part gives our applications their unique value, exists in embed points. This is a problem, because code in embed points is difficult to test and reuse. So how do you retain the benefits of Clarion development while improving your code quality? By isolating your business logic (which is where most of our applications' unique value resides) into classes. Think of your applications in terms of the UI layer, the business logic layer, and the data access layer. Keep those layers separate as much as possible (even though all three are mixed up in a standard Clarion app).

We had an extensive discussion of classes versus interfaces, with Mike strongly advocating an interface-based approach over a purely class-based approach.

Bruce Johnson gave a good illustration (with lots of hand waving) of virtual methods, and Mike followed with his version. Bruce opined that there is no good argument for protected methods and properties - use either public or private.

We also batted around the SQL vs TPS question. Load is a huge factor in that decision, although the IP driver can significantly increase the load that can be managed by TPS.

If you can defer the decision to move to SQL do so, as migrating to SQL is not without cost.

All of these points enable existing Win32 applications to last longer, which is certainly one important kind of future-proofing.

The other kind of future-proofing we discussed is how do we bring our applications forward to .NET?

The panel's conclusion was that there is **no viable migration path for taking an existing, mature Clarion Win32 application forward to .NET.**

That may come as a shock, or it may come as a relief.

Bruce gave a terrific argument *against* rewriting applications, which I'll try to summarize (parenthetical comments are my own).

It's all about time and effort. Let's say you have a major application that you started to develop ten years ago. It's a mature application now. It's had ten years of tweaking and polishing. It does just what the users need it to do. But it's a Win32 application, and you feel a pressing need to create a .NET version for whatever reasons - extensibility, or a better user interface, or whatever.

Only rewriting isn't a simple process. (The tools change. The language changes. Yes, Clarion# is different from Clarion Win32 in some significant ways - no WINDOW structure, no ACCEPT loop, zero based array indexes, etc.; the data access layer changes. There is no ABC library. Almost *everything* changes.) You could conceivably spend as much time rewriting your application, recreating it down to the finest detail, as you did developing it in the first place. And that mature Win32 application is the standard your customers will expect you to meet. And having met the standard you're just back where you started, from your customers' perspective. You need to exceed that standard to get paid.

Can you afford ten years of development before you start earning money from that application again? (And who knows what software development will look like in ten years? Probably it's time to rewrite *again*.)

A more sensible role for .NET is new application development. Create something simpler for a new market or a subset of your existing market. (Over time this application may also become mature, but as with your existing application you're going to get paid along the way.)

Mike agreed with Bruce that there really is no migration path to .NET for existing Clarion Win32 applications. I've been holding out hope that refactoring existing code into classes would make it easier to bring some of that code forward into a .NET application, but this discussion took most of the wind out of my sails. I think Bruce and Mike are right - completely new application development is a much safer path forward.

I asked my co-panelists whether the absence of a migration path called into question many of the future-proofing techniques we'd been advocating up till that point. To my mind one of the main reasons for moving code out of embeds and into classes is to increase its portability. And of course the answer is that improving code is always a good thing, and that process can definitely increase the longevity of existing applications, whether or not you're able to move some of that code to .NET.

As with migrating to SQL, consider whether you really need to move to .NET now. The costs are significant; deferring that decision will probably be to your benefit. Make economic decisions.

As Bruce said, don't get bogged down in the technological choices. There are many more important things to think about and work on.

Keep learning. Keep improving. Keep making all aspects of your business better.

ClarionLive video

Video of the conference sessions, including the Saturday panel discussion, is available to attendees [online at the ClarionLive web site](#). Log in to see the links. If you didn't attend, either virtually or in person, you can still buy online attendance. Contact John and Arnold at clarionlive@gmail.com.

David Harms is an independent software developer and the editor and publisher of Clarion Magazine. He is also co-author with Ross Santos of Developing Clarion for Windows Applications, published by SAMS (1995), and has written or co-written several Java books. David is a member of the American Society of Journalists and Authors (ASJA).

Article comments

by George Riddell on November 9 2010 ([comment link](#))

It appears SoftVelocity has run out of innovative gas - it happens. The remaining Clarion developers are obviously at a crossroads and this DevCon and your reports could not be more timely.

I think your suggestion that we refactor and tune our Win32 apps is on the money. With tools like NetTalk, IP Driver and @Thin Clarion developers are in tune with the future. The future is not Windows dot net alone, it will be Windows on some devices, with HTML5 and Java everywhere. Clarion can generate Java code, see @thin.

Our community owes a great deal to Arnold Young and John Hickey, and a few others for organizing this meeting.

by Dave Harms on November 9 2010 ([comment link](#))

George,

As to SV's situation, I really cannot say. I hope I've learned that lesson at least. But the time it has taken to bring C7 (largely a .NET application) to market and the ongoing process of making C7 work as its users expect should be instructive for all of us who think about porting our own Win32 apps.

I agree completely about the thanks we owe Arnold & John & Co., and also CapeSoft, Thin@, Andrew Barnham (for the desktop Java option), and others who create alternatives to fill needs.

And we owe SV thanks as well. Obviously things haven't turned out as many of us had hoped on the .NET side (at least not yet), and there are still C7 issues, but C7 is a viable tool for quite a few Clarion devs and SV is still here and still fixing and improving their product.

by Matjaz Jost on November 10 2010 ([comment link](#))

Hello !

I've heard (read) about C8. What will there be new ? I have an idea: native connection to .NET in a way that .NET modules can be simply called from Win 32 Clarion. What do you think about it ?

Regards. Matjaz

by Dave Harms on November 10 2010 ([comment link](#))

There isn't a whole lot of information yet on what will be new in C8. So far the list I have includes:

- IDE usability improvements - locators, keyboard shortcuts
- IDE visual improvements - cleaner look, better icons
- RTL improvements - more work done on Unicode

As for easier calling of .NET code, that's in the next release of Clarion.NET. You'll be able to export procedures from your .NET DLL and call those in the same way you'd call procedures in a Win32 DLL. Supported data types include:

- long
- short
- ulong
- ushort
- byte
- string
- cstring

This support is not dependent on changes to the Win32 product; as far as I know you could do this kind of interop from C6 if you wanted to.

by Matjaz Jost on November 12 2010 ([comment link](#))

Thank you Dave for the answer. I was basically interested for possibility of calling third party .NET modules from Win32 Clarion (7 or 8) without the need for Clarion.NET. But as you said, could I prepare in Clarion.NET the third party .NET modules for use with Clarion 7 (8)?

Regards. Matjaz

by Dave Harms on November 12 2010 ([comment link](#))

Matjaz,

You can still call .NET code from C6/7/whatever without Clarion.NET, using the process Wade Hatler described in his [series on calling .NET](#). But Clarion.NET (as of the

next release, if I understand correctly) will make it much easier to do procedure calls into .NET code.

 [BACK TO TOP](#)

Copyright © 1999-2010 by CoveComm Inc. All Rights Reserved. Reproduction in any form without the express written consent of CoveComm Inc., except as described in the subscription agreement, is prohibited.

ClarionLive DevCon Day 3: Final Sessions

By Dave Harms

Posted November 16 2010

The Denver DevCon finished up with a half day of sessions on Sunday, October 31.



Superman making sure everything's going smoothly...

Thin@

Marko Golem started things off with a bonus workshop session on using [Thin@](#), a thin client solution for Clarion. Thin@ is similar in concept to [ClarionNet](#), a thin client technology created by Michael Brooks and currently owned by Ivan Mintoff.

Thin@ makes your existing Clarion applications available across the web. Rather than downloading the entire application (and using a secure database connection), your users run the Thin@ client executable, which connects to the Thin@ server application you create from a Clarion app.

Thin@ effectively separates the user interface from the rest of the application. The client app just presents whatever the server tells it to present, so if you change the application on the server all your users are automatically upgraded to that version.

The standard client application is a Win32 executable, but Thin@ also offers a beta version of a Java client, so your Clarion application can run anywhere Java runs.

Marko showed how to turn an existing application into a Thin@ app. He pointed out the need to add equates to menus where needed, and advised against using the DateTimeDisplay extension template. He then added the Thin@ global extension template and showed some of the optimization features.

If you remove the Thin@ global extension template you will need to manually remove the project defines Thin@ has added.

Marko then added the application to the Thin@ server using the configuration tool.

Mark also covered server administration issues such as user licensing and application upgrading.

Thin@ is an interesting new take on Clarion thin client technology; look for more detailed coverage in ClarionMag in the near future.

Software as a Service

Daniel Pavlic continued Marko's theme in a more general vein, discussing rich/thin clients, AJAX versus rich internet apps (RIAs), and Software as a Service (SaaS) in general.

Daniel surveyed the three most popular rich client technologies, including Adobe Air (the front runner), Silverlight and JavaFx.

How about Javascript? Can you build RIAs with Ajax? According to Daniel, yes you can, but an Ajax app isn't automatically a rich internet app. And Flex AMF3 will have better performance.

RIAs have evolved from non-RIA HTML pages through growing use of Ajax, then Ajax on top of libraries, and eventually to standalone RIA clients. In Gartner's analysis, Adobe Air and Microsoft Silverlight are the top choices.

The more complex your user interface requirements, and the greater the need for client computing power, the more you should be leaning toward the RIA side (provided and can live with the platform dependence issues). If you don't need such a rich UI, and you have lots of content (text, pictures), and you need platform independence, Ajax may be a better fit.

SaaS has a number of maturity levels, from each customer having their own instance of the application to large farmed systems supporting many users.

Daniel also covered the platform wars between the various vendors. There's no way to know yet who will be the winner. The four major players are Microsoft, Apple, Oracle and Google. Daniel added in the hardware vendors, the operating systems, programming languages and devices, for a very complex set of variables.

How do you choose a technology? Daniel suggests you consider APaaS, or Application

Platform as a Service. This is what technologies like Thin@ and, for example, Magic Software's UniPaaS provide (Magic is a longtime niche market product similar in some ways to Clarion, although with a larger user base). You can have different UI providers for different platforms, and the application itself is a service in the cloud.

Pierre Tremblay

SoftVelocity's Pierre Tremblay handled the .NET training duties. Although the .NET AppGen isn't out yet, the Clarion# language has been available for several years already and was the focus of the presentation.

Pierre covered .NET language changes and enhancements, such as exception handling, data type changes (including those needed for compatibility with Clarion-specific data types), namespaces, comparison operators, events and delegates, extension methods, properties, and generic types.

Pierre also talked about Language Integrated Query (LINQ), which is used for a variety of query tasks and in a number of libraries. SoftVelocity has implemented a LINQ interface for the file drivers called the LINQ to File provider. You declare IFileQuery instances of the file's record type and issue a LINQ query against that interface to get back file records. Some of the ABC functionality has been ported to the LINQ to File provider. Pierre also showed how to use ADO.NET to connect to data. He covered data grids and data binding, showing how to retrieve data and display it as a browse.

Pierre finished up with a preview of the .NET Application Generator. He fired up the combined Clarion 8/Clarion.NET build 7694 and opened the .NET template registry. The .NET templates are T4 template, but the registry itself is now a DLL. That's because the T4 templates are compiled .NET code, which indicates that at some point it will be possible to debug templates using the .NET debugger. The template editor provides T4 syntax highlighting.

Data access layer generation was pretty much instantaneous, although Pierre declined to press the compile button.

There was a lot of good content in Pierre's session - if you want to get a handle on how SoftVelocity is approaching .NET application development then Pierre's session is well worth your time.

Bob Zauere Q&A

In the closing Q&A panel discussion Bob Zauere (a.k.a. Z) took questions from a panel of speakers as well as the audience.

On the .NET side there were a number of questions about the .NET AppGen. While the high level view of applications is very similar to the current AppGen, the code underneath is

quite different, adopting a more modern layered approach. This led to some discussion of layered architectures as well as the inadvisability of migrating existing Win32 applications to .NET (irrespective of whether they are Clarion apps).

The new template system, according to Z, is designed to be much more open so you can more easily integrate templates into an existing set, or mix and match templates. The entire code generation system is intended to be much more flexible than the existing Win32 AppGen.

There were a number of questions about Win32/.NET interoperability - specifically how to call .NET code from Clarion Win32 apps. The changes to Clarion.NET make that process easier. While there are a lot of different possibilities for interop, including (theoretically) using .NET controls directly inside Clarion Win32, a more likely approach is to open a .NET window and make it a child window of the Win32 app.

There was also some discussion of the community's dissatisfaction over the lack of communication from SV on the .NET AppGen delays.

On the C8 side, in addition to the features already mentioned Z included ongoing work on Unicode, HTML help support and RTL changes. Z specifically asked attendees about their need for a 64 bit version of the Clarion compiler and RTL. About 90% put up their hands in favor of 64 bit, and a few developers indicated they were already experiencing memory problems due to 32 bit constraints on available RAM.

The awards!

At the conclusion of the conference the presenters gave Arnold and John tokens of appreciation, including engraved iPad and iPod covers (sourced by Rick Martin) and bottles of ice wine (brought by Mike Hanson). Then John presented Arnold with the first ever CLDC Arnie Award, which unfortunately broke in his luggage.



From left to right: John Hickey, Rick Martin, Arnold Young, Gordon Holfeder, Mike Hanson, Bob Foreman, Pierre Tremblay, and Daniel Pavlic

Robert Paresi also received an Arnie for his contributions above and beyond, including finding the Inverness Hotel.



Virtual Method awards went to Benjamin Krajmalnik (Master Chauffeur, Internet Connection Specialist, Security Detail), Tom Hebenstreit (Audio/Video Specialist), and John Dunn (Prize Winner Selection Software Modifications). Arnold presented John with the first ever CLDC Superman Founder Award. Heck, even the moderator got one of those. And the presenters all received Virtual Method Awards.

If there's another DevCon next year (and what, after all, does "Arnie" say to all of us other than "I'll be back") I think there should also be an Ironman award in honor of Rick Martin. Rick did two presentations on Saturday afternoon, but Sunday morning he was looking decidedly unwell. When he got home he discovered he had pneumonia. He's much better now, thank you.

Impressions and conclusions

An actual physical gathering of developers is, and I think always will be, a far different experience from an online gathering.

There's an oft-quoted (and oft-debunked) study that showed up to 93% of all communication is non-verbal. Whatever the percentage really is, it's beyond doubt that talking with others in person is a much more complete experience than doing so online, particularly in the newsgroups. After a weekend of meeting with developers from all parts of the globe, I came away with a number of impressions and a feeling of optimism, not so much about future technologies but about the ongoing value of the Clarion community.

I came to Denver looking for more than just programming information. I wanted to get a feel for where Clarion developers were headed. I also wanted to get a better sense of where .NET fits into the grand scheme of things, and I'm grateful to my Saturday night co-panelists, Mike Hanson and Bruce Johnson, for helping me see that subject more clearly.

I came away from Denver with a bunch of bullet points:

- Don't worry about SoftVelocity, just take care of your own business and your own customers.
- Improve your code constantly
- Both object orientation and SQL can have huge payback

- .NET is a factor, but for many not a very big factor
- Forget about porting your existing Win32 applications to .NET
- SoftVelocity's .NET tooling will take time to mature - don't expect too much
- There's lots of life left in the Win32 product.
- Defer major platform shifts if possible
- Make your decisions based on the technology you have before you

There wasn't a lot of product hype at the ClarionLive DevCon, but that didn't detract from the experience at all. This really was a learning event, and the lessons went well beyond just code.

Give Arnold and John a hand

I can't say enough about Arnold and John's efforts. Their vision for the DevCon was to get a bunch of Clarion developers together to learn from each other, and they realized that vision in spades. If you missed the DevCon, you can still [buy access to the conference recordings for \\$395](#). If you don't find more than your money's worth there I'll be astonished. And I think it's worth noting that the recordings were available mere days after the conference ended.



One last look

The downside of attending a busy developer's conference is that you hardly set foot outside the hotel. Which is a shame, if the hotel is the Inverness. Yes, that's a golf course. From the other side of the hotel you can see the Rockies.

If there is another ClarionLive DevCon here, I plan to take a little more time after the conference to explore Denver and its surroundings.



David Harms is an independent software developer and the editor and publisher of Clarion Magazine. He is also co-author with Ross Santos of Developing Clarion for Windows Applications, published by SAMS (1995), and has written or co-written several Java books. David is a member of the American Society of Journalists and Authors (ASJA).

Article comments

[↑ BACK TO TOP](#)

Copyright © 1999-2010 by CoveComm Inc. All Rights Reserved. Reproduction in any form without the express written consent of CoveComm Inc., except as described in the subscription agreement, is prohibited.

Fooling the FileManager

By Steven Parker

Posted November 17 2010

There was a recent news group posting, asking:

I need to put a table in the table schematic but I want to handle all access myself.

My procedure scans the customer table and, if certain conditions are met, I want to create a file with a name and path that I set, add the necessary records and close the file.

That is the requirement. The request is can it be done and, if so, how?

The answers are “of course” and “keep reading.” Naturally.

The problem

But, first, I want to explore why someone would want to do this.

In this case, the [interlocutor](#) is clear about what he wants. (I would do this differently; the requirement of the file in the table schematice limits the options. But, options is one of the things we all like so well about Clarion.)

His customer file has a child file and that child uses the `NAME()` attribute to allow him to create many files sharing the same file structure. There are many Clarion Magazine articles showing how to use the `NAME()` attribute to allow a single code base to read and write a number of physically distinct files.

Our interlocutor is also clearly aware that the variable in the `NAME()` *must* be assigned before any attempt to open the file. Failure to do so inevitably causes an ErrorCode 45 (and, note, *any* time you see an ErrorCode 45, there is one and only one cause: you failed to assign a value to a `NAME()` variable before opening a file).

The problem is that, on opening the browse (or form or report or process) the generated code will open this child file. Only there are two big – really big – unknowns. Those unknowns are:

1. Do I need to access any of these child files? (And, this hides the bigger question of “are there

any?")

2. Which one do I use for NAME ()? Which one is the “default” file to be opened at procedure initialization?

Because these child files exist only when certain conditions exist, there may not be any at all when I open the procedure.

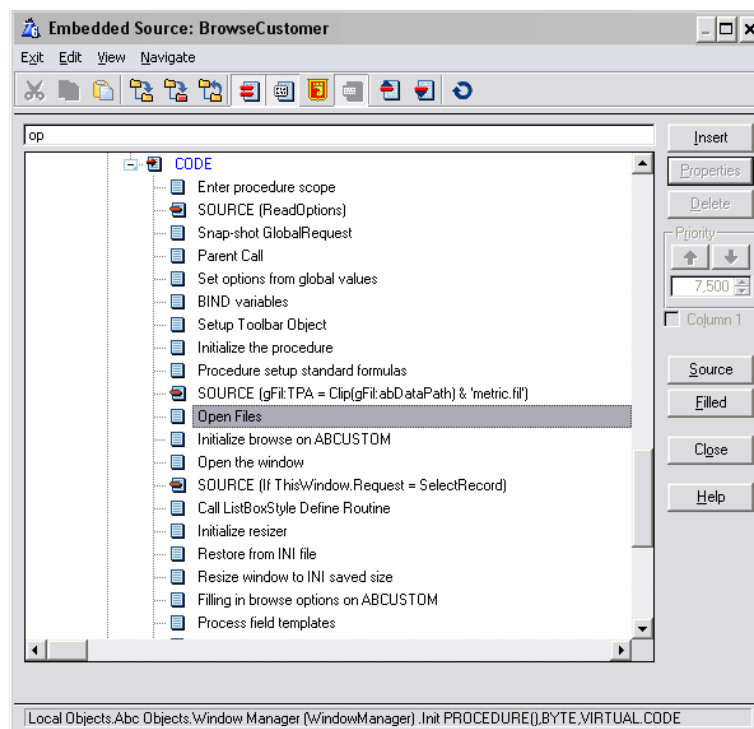
Do I just plug a dummy NAME ()? If I do, then I need to close, rename, re-open when I find that I do need to create and write or open and read one of them.

Our interlocutor is right, it's easier to do an end-run around the FileManager and not let the template's file handling handle this for me.

In general, the need to control the template's file handling is relevant *any* time you may not know the NAME () until a user takes a specific action or any time you don't want to open/close or transmit across a network files with large record structures, large size or large numbers of files.

What do I know

I know, to begin with, that there does exist an embed which looks like a good candidate for what I want:



?

Figure 1. Where the template opens files

Pressing the “Source” button to examine the template-generated code indicates that this is the right place:

```

? Procedure setup standard formulas
IF SELF.Request = SelectRecord
    SELF.AddItem(?Close,RequestCancelled)
ELSE
    SELF.AddItem(?Close,RequestCompleted)
END
? [Priority 7001]

? Open Files
Relate:ABCUSHST.Open
Relate:ABCUSLAB.Open
Relate:ABCUSPLU.Open
Relate:ABCCUSTOM.SetOpenRelated()
Relate:ABCCUSTOM.Open
Relate:ABLAWWAY.SetOpenRelated()
Relate:ABLAWWAY.Open
Relate:ABLEVELS.SetOpenRelated()
Relate:ABLEVELS.Open
Relate:Cus2Lab.Open
Relate:TMPASCI1.Open
SELF.FilesOpened = True
? [Priority 7600]

? Initialize browse on ABCUSTOM
BRW1.Init(?Browse:1,Queue:Browse:1.ViewPosition,B
? [Priority 7900]
    
```

Figure 2. Default template code

(Note: the pictured code is from a real customer browse.)

The “normal” method of dealing with a NAME () that has not yet been assigned is to make the assignment just before opening:

```

gFil:TPA = Clip(gFil:abDataPath) & 'metric.fil'
? Open Files
Relate:ABCUSHST.Open
Relate:ABCUSLAB.Open
Relate:ABCUSPLU.Open
Relate:ABCCUSTOM.SetOpenRelated()
Relate:ABCCUSTOM.Open
Relate:ABLAWWAY.SetOpenRelated()
Relate:ABLAWWAY.Open
Relate:ABLEVELS.SetOpenRelated()
Relate:ABLEVELS.Open
Relate:Cus2Lab.Open
Relate:TMPASCI1.Open
SELF.FilesOpened = True
? [Priority 7600]
    
```

Figure 3. Standard NAMEing

In this case, the variable, gFil : TPA, goes with TMPASCI I (which, unlike the rest of the files in this procedure, is not assigned a value when the program opens).

In the case presented in the newsgroup posting, I do not know what value to put into gFil : TPA. In fact, I do not even know that I will need a TMPASCI I file.

What I want to do is get rid of the

Rel ate: TMPASCI I . Open

line.

If I did not have TMPASCI I in the table schematic, this would not be an issue.

But I do. And the interlocutor wants it there.

I want it there so that the templates will handle the closing for me. There are other reasons why I might want it in the table schematic instead of doing all file handling manually. Those reasons are not germane right now.

The “Ah ha!” moment

It would be nice if I could comment out the one line. Because it is template generated, of course, I cannot.

But I can comment out all of them. Clarion lets me OMIT entire blocks of code, including template generated code:

```
omit('**shp**')
! Open Files
Relate:ABCUSHST.Open
Relate:ABCUSLAB.Open
Relate:ABCUSPLU.Open
Relate:ABCUSTOM.SetOpenRelated()
Relate:ABCUSTOM.Open
Relate:ABLAYWAY.SetOpenRelated()
Relate:ABLAYWAY.Open
Relate:ABLEVELS.SetOpenRelated()
Relate:ABLEVELS.Open
Relate:Cus2Lab.Open
Relate:TMPASCI1.Open
SELF.FilesOpened = True
! [Priority 7600]
! **shp**
```

Figure 4. OMITting all file opening

Naturally, should I be foolish enough to leave the code this way, bad things will happen. Because none of the files required by this procedure are open, I would expect all manner of errors.

The solution

The solution is simplicity itself. “Just” copy the generated code, paste it below the OMIT terminator.

Then, I can delete or comment out the parts of the code I don’t want:


```

omit('**shp**')
! Open Files
Relate:ABCUSHST.Open
Relate:ABCUSLAB.Open
Relate:ABCUSPLU.Open
Relate:ABCUSTOM.SetOpenRelated()
Relate:ABCUSTOM.Open
Relate:ABLAYWAY.SetOpenRelated()
Relate:ABLAYWAY.Open
Relate:ABLEVELS.SetOpenRelated()
Relate:ABLEVELS.Open
Relate:Cus2Lab.Open
Relate:TMPASCII.Open
SELF.FilesOpened = True
! [Priority 7600]
! **shp**
Relate:ABCUSHST.Open
Relate:ABCUSLAB.Open
Relate:ABCUSPLU.Open
Relate:ABCUSTOM.SetOpenRelated()
Relate:ABCUSTOM.Open
Relate:ABLAYWAY.SetOpenRelated()
Relate:ABLAYWAY.Open
Relate:ABLEVELS.SetOpenRelated()
Relate:ABLEVELS.Open
Relate:Cus2Lab.Open
!Relate:TMPASCII.Open
SELF.FilesOpened = True

```

Figure 5. Replacing unwanted template code

There is, however, a very important caveat in doing this. Should I ever add a file to the schematic, or remove one, I have to remember to update my embedded code. I have used OMI Tting like this, twice I think, and have run into the situation where I changed the file schematic and forgot to update my embedded code. Got some seriously strange results when I first ran the procedure.

The point of the exercise

Were it not required that the file be listed in the table schematic, there are easier ways to take control over the creation and opening of multiple child files. That's not the real point.

Were it not a requirement, as it is for our interlocutor, that the file be in the procedure's table schematic, there is a safer alternative. When this file is required:

- Ensure there is a FileManager for the file by Global Properties | Individual File Overrides | double clicking on the file and ticking "Generate file declaration"
- Check the condition that requires a child file
- Set NAME()
- Access: <myFile>. Open
- Do my process
- Access: <myFile>. Close

In other words, there needs to be a mighty good reason that the file *must* be in the schematic.

The real point is that, should it become necessary, even template generate code can be overridden. What I did for file opening could just as easily be done for any template generated code.

We all override template code all the time. Putting a RETURN in TakeNoRecords, Before Parent call overrides template code. Testing a condition in Val i dateRecord and setting ReturnVal ue or RETURNing Record: Fi l l tered overrides template code. Using OMI T and substituting a variation just takes normal (such as it is) Clarion-developer behavior to the next logical level.

Steve Parker started his professional life as a Philosopher but now tries to imitate a Clarion developer. He has been attempting to subdue Clarion since version 2007 (DOS, that is). He reports that, so far, Clarion is winning. Steve has been writing about Clarion since 1993.



Article comments

by Gustavo Schiaffino on November 17 2010 ([comment link](#))

Excellent, Easy and effective solution !!

by Paul Blais on November 28 2010 ([comment link](#))

Learning to work in the embeditor directly makes this type of code flow easier as you learn the total context of the ABC templates. It's a handy technique when you have to do custom population of the browse queue with complex computed columns, custom colors, and icons. Clarion writes the view for you so you can do the rest.

[↑ BACK TO TOP](#)

Copyright © 1999-2010 by CoveComm Inc. All Rights Reserved. Reproduction in any form without the express written consent of CoveComm Inc., except as described in the subscription agreement, is prohibited.

Review: Thin@ Transforms Clarion Desktop Apps Into Software-as-a-Service Internet Apps

By Dave Harms

Posted November 29 2010

Thin@ is a "smart client" technology that lets you run your Clarion application on a server and deliver only the application's user interface to the user across a LAN, an intranet, or the internet. Thin@ is one of two Clarion products in this market space - the other is ClarioNet (which was originally created by Michael Brooks and is currently maintained and sold by Ivan Mintoff).

Your users don't have to have the full Clarion application installed at all - they just install a standard Win32 client application which they use to run any number of Clarion applications remotely. This approach combines the deployment advantages of web applications with the rich user interface of desktop applications. Furthermore, a Java version of the Thin@ client application is now in beta, which will allow you to effectively run Clarion applications remotely on any operating system that supports Java.

Overview and terminology

Any time the topic of servers comes up there's opportunity for confusion, because sometimes a server means a physical piece of hardware and other times it means a program which serves up information (such as a web server or a database server).

So to try to keep things clear I'll propose a few definitions for the following explanation of how Thin@ works (the general principles apply to ClarioNet as well):

- user: the person who is using some software at their own computer
- physical server: a computer which is not the user's computer but which by various means provides information to other computers, usually via one or more server *programs*
- server program: software, usually running on a physical server which provides information to other programs running on the same or other computers
- Thin@ server: a confusing term since there are (or appear to be) two kinds of Thin@ servers, but for the purposes of this discussion this is the part of Thin@ that runs on the physical server.

- Thin@ smart client: a program which the user obtains and runs, and which can be used to execute (or so it seems to the user) a Clarion application running on the physical server.

I'll begin with the Thin@ smart client, in Figure 1.

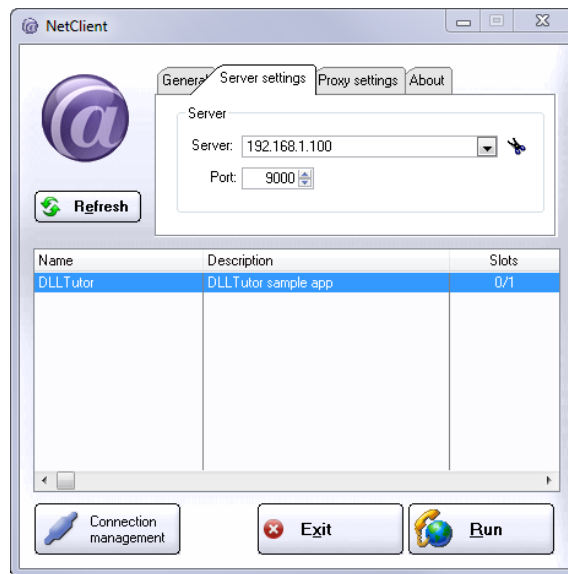


Figure 1. The Thin@ smart client

You use the Thin@ client (which you install on your desktop) to connect to a Thin@ server. The Thin@ client asks the Thin@ server for a list of available applications, and presents that list to you. You select an application and you run it. The application materializes in front of you (Figure 2).

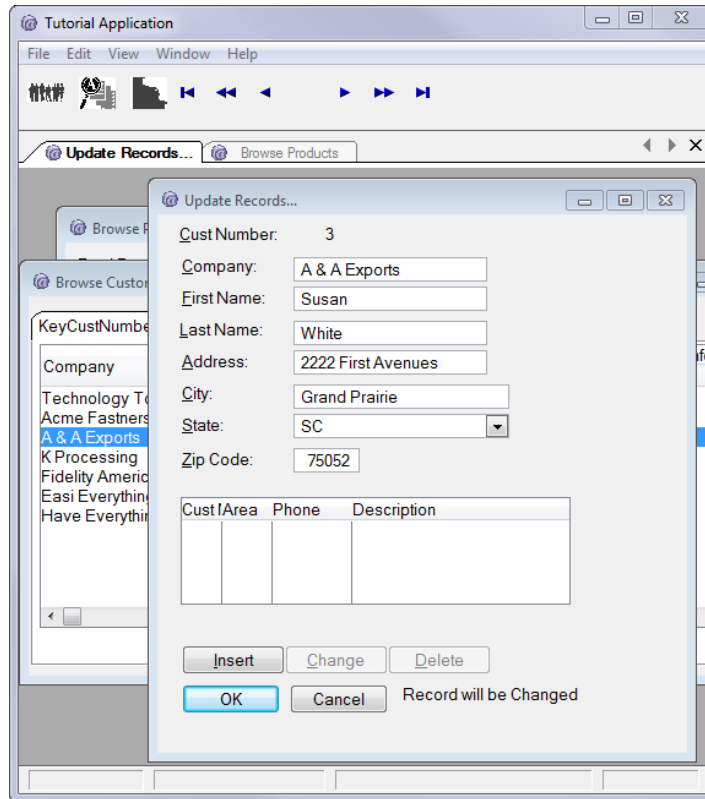


Figure 2. Running an application via the Thin@ smart client

Only it's not the application at all, because that application doesn't exist on your computer; it exists on the physical server. So what's happening?

When you tell the Thin@ smart client you want to run an application, the Thin@ smart client communicates your intent to the Thin@ server. The Thin@ server launches an instance of the application. But it's not just a standard Clarion application that's running; it's a Clarion app that's been modified (via templates) to become a little server app all on its own (in fact, it's possible to run the application locally via the client EXE, without using the Thin@ server at all). When that application starts up it opens a window (e.g. the application frame) and tells the Thin@ server what that window looks like. The Thin@ server passes this information back to the Thin@ smart client running on your computer, which creates the screen.

Now, let's say you click a menu option to open a browse. That event gets sent back to the Thin@ server, which communicates the event back to the running application. The app handles the event and returns its new state back to the Thin@ server, which updates the screen accordingly. And so on and so on. Essentially the user interface part of your application, the skin if you like, is being delivered across the internet while all the code executes on the server.

Installation

You can get a fully functional trial version from the Thin@ web page. Somewhat

confusingly, the downloadable trial is listed on the Purchase page. The download is 30 megabytes, and is available as a setup EXE or as a zip file. You should use the install program the first time; after that you can download the zip file and just copy across the updated files.

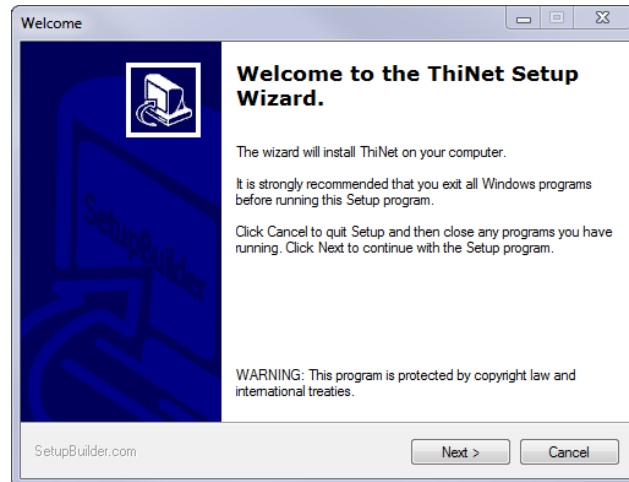


Figure 3. Launching the installation

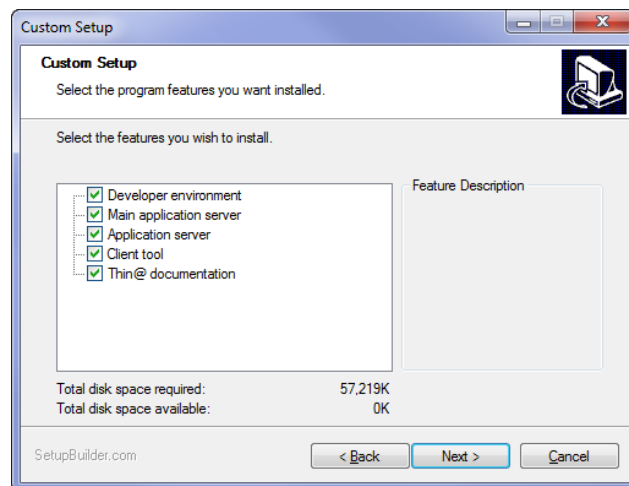


Figure 4. Selecting features to install

The setup contains install files for five major components:

1. The Developer environment, which is all the supporting files needed to add thin client functionality to Clarion desktop applications.
2. The load-balancing server.
3. The non-load balancing server
4. The thin client executable program
5. The documentation

Figure 5. Choosing a target version

I elected to install Thin@ for Clarion 7.2.

I have many versions of Clarion 7 installed; the Thin@ installer defaulted to the most recently installed version.

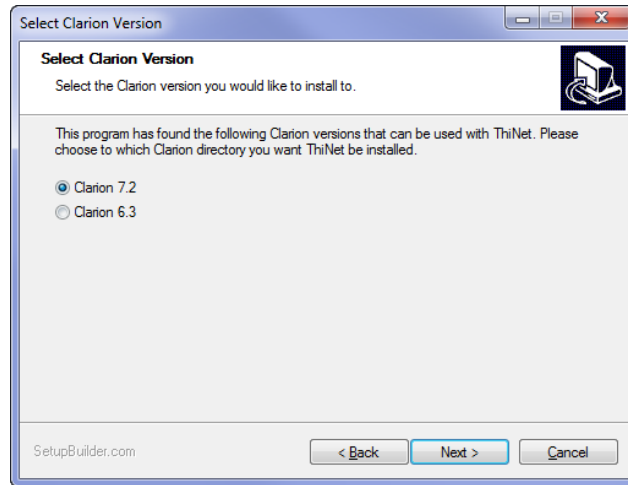


Figure 5. Selecting the Clarion version

The default location for the Thin@ server is C:\ThiNet - I changed this to D:\dev\ThiNet.

I accepted the default value for the location of the Thin@ client application.

Figure 6 shows default setting for starting up the Thin@ server programs. I already have a lot of programs running at startup on this machine, so I deselected this option. I also don't need them running as services since this is a development machine only. But keep in mind that if you don't load the server programs on startup, either via the startup folder or by running them as services, you will have to launch them manually after a reboot.

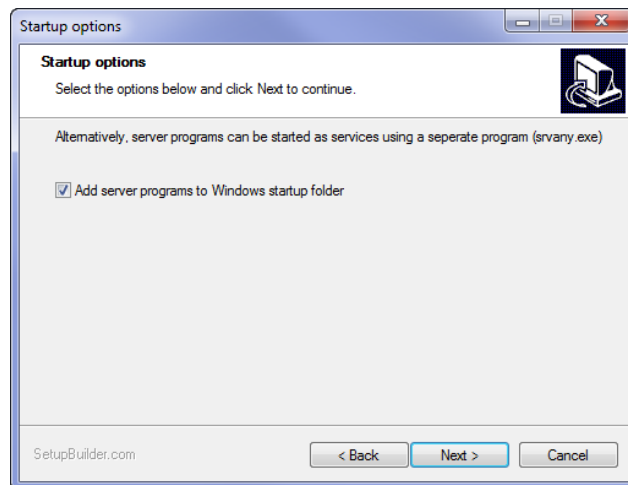


Figure 6. Startup options

On a production server you would almost certainly want the server programs running as a service, since that way they would start up automatically after a reboot without needing anyone to log in. The installer indicates you can use the srvany utility to run these server programs as Windows services, but doesn't provide any further information on how you

might do that. You can find srvany in the [Windows Server 2003 Resource Kit Tools download](#).

Next comes a warning about a Windows heap size limitation:

Microsoft Windows has a desktop heap limitation which prevents you from opening a large numbers of windows or, in some cases, out of memory errors can occur (see: <http://support.microsoft.com/kb/947246/en-us>).

You've probably never run into this limitation in the normal course of using Windows as a single user, but a Thin@ server may be running a large number of application copies at the same time, each with many open windows. The note goes on to explain how to fix the problem and also notes that this same information is contained in the Installation Instructions manual.

Before completing, the installer shows a readme file with instructions for those upgrading from previous versions of Thin@.

The installer registered the templates (after asking me to be sure I'd closed down the Clarion IDE), but apparently template registration doesn't always succeed so you may have to register the template manually.

When the install completes you have the option of launching all necessary admin tools as well as the QuickStartGuide PDF. NetPort, NetListener, and NetServer are all server programs rather than admin tools per se. Since you need all three running to operate Thin@, they should probably be reduced to a single "Launch the Thin@ server" option.

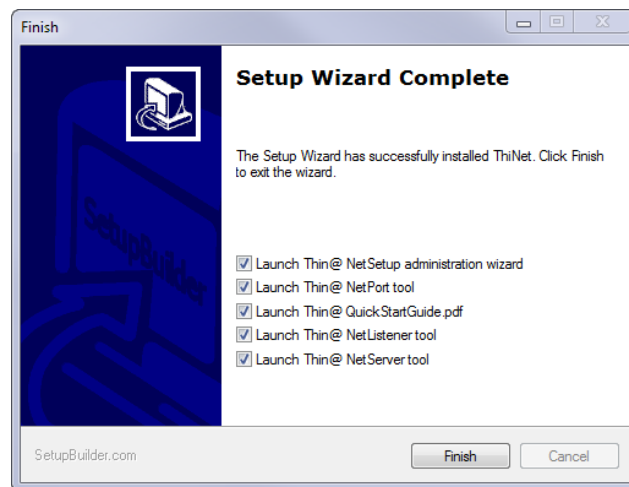


Figure 7. Launch options

After setup completes it launches the Thin@ NetSetup Wizard



Figure 8. Wizard screen #1

Step 1, and your only option if this is a new installation, is to add a new thin@ server.

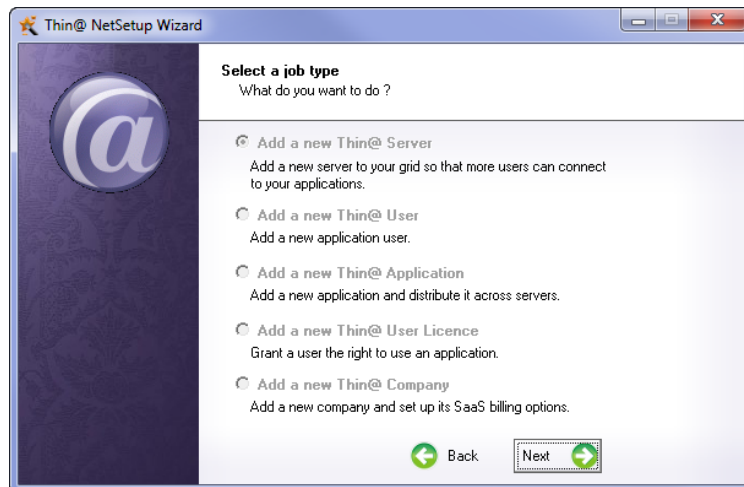


Figure 9. Adding a new server

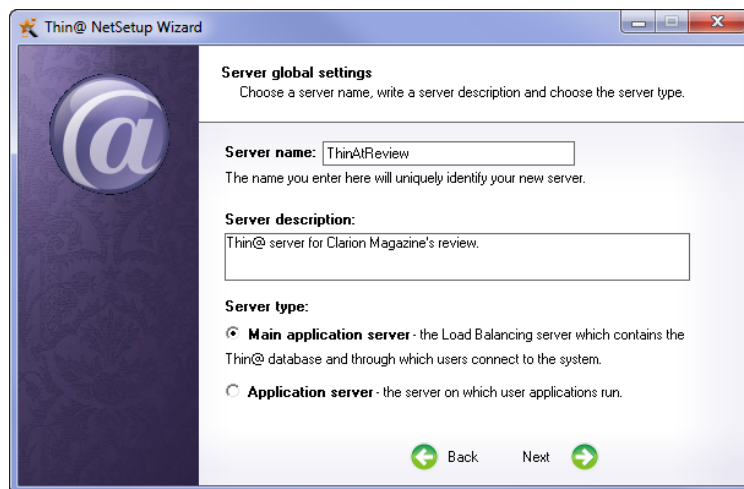


Figure 10. Setting the server name

I was a little confused over the difference between the Main application server and the application server. Was it really just that one has load balancing and the other doesn't? Pressing F1 didn't bring up any help.

I found a link in my Start Menu for the administrator's guide, but the link didn't work.

Using [VoidTools Everything](#) I quickly found all the PDFs in the C:\Program Files (x86)\Thin@\Thin@ Documentation directory. I did find a diagram in the Thin@SystemArchitecture PDF which suggested that the main server is always needed, and that is in fact the default option so I let the default stand and soldiered on.

I later asked Thin@'s Daniel Pavlic for clarification. He replied:

Yes, the main server is the server you must have in every Thin@ farm. It holds the NetServer, NetListener and NetMonitor processes. NetPort is optional. The non-main servers are the additional servers you add to your farm, and they must have the NetListener and NetMonitor services active, NetPort is optional.

When setting up Thin@ for the first time you will only be creating a single server, and if you have no more than 100 or so clients you may not ever need more than that one server (depending on how capable it is and how big your apps are). But when reading the documentation it helps to keep in mind that Thin@ is explicitly designed to scale to many physical servers.

Next came the port settings for the Main application server (Figure 11), followed by settings for the range of application ports to use and the maximum percentage of system memory to assign to Thin@ (Figure 12).

If you exceed this memory percentage Thin@ will refuse new connections with an error message. If you set a value over 95% however Windows performance will degrade.

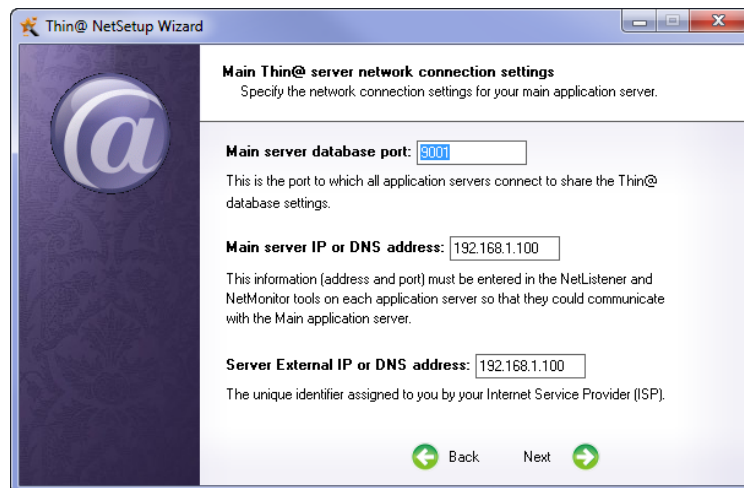


Figure 11. Server ports and addresses

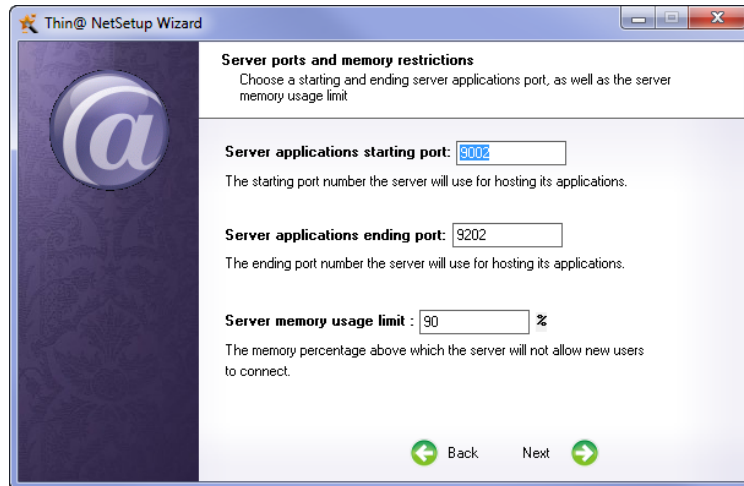


Figure 12. Application ports

I chose to disable authentication (Figure 13) but I was intrigued by the option for external authentication. As it turns out this is a fairly simple process - you create an EXE which receives two parameters and which writes the authentication result (using a defined format) into a text file which is then read by the server. Thin@ has its own authentication system as well.

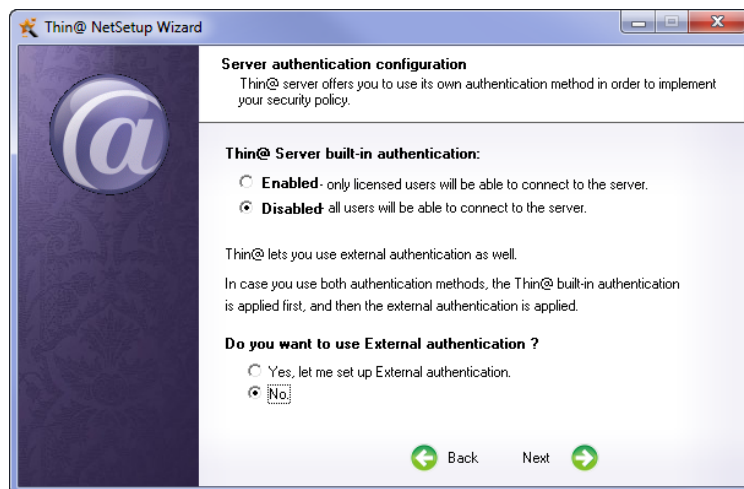


Figure 13. Authentication configuration

In Figure 14 the wizard is asking for a stats path and whether the server should be launched. Looks like the wizard is just about done!

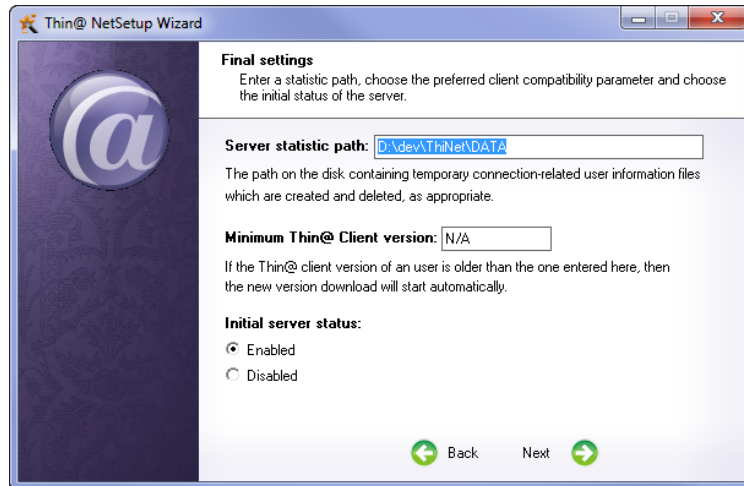


Figure 14. Almost done!

And in fact it is done (Figure 15).

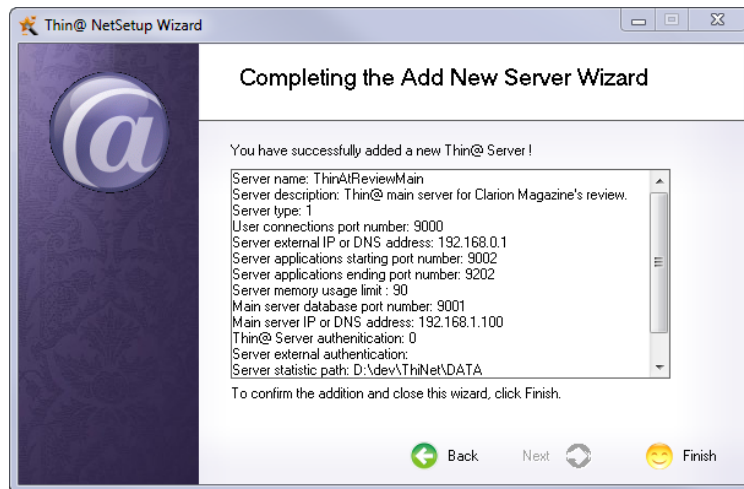


Figure 15. Done installing.

A moment later Windows notified me that NetListener and NetSrv had been blocked by Windows firewall. I authorized both programs by clicking on the Allow access button.

At this point the NetSetup Wizard relaunched and when I clicked on the Next button I saw that the wizard was configured for adding a new Thin@ user (Figure 16).

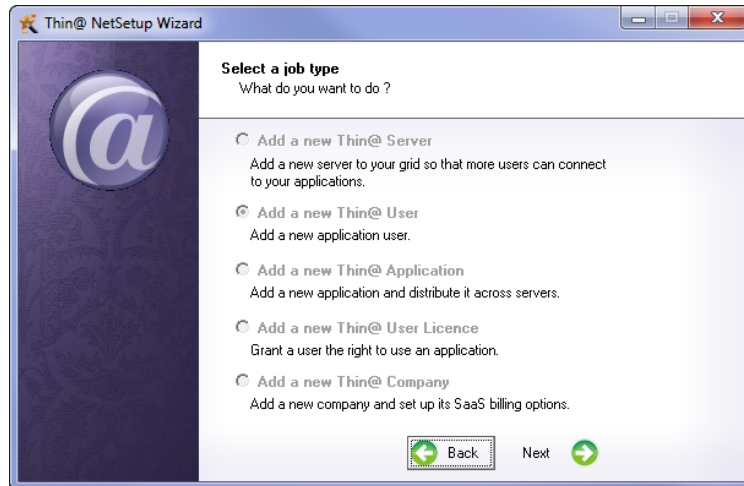


Figure 16. Adding a new user

I specified the username and password (Figure 17) and completed the wizard.

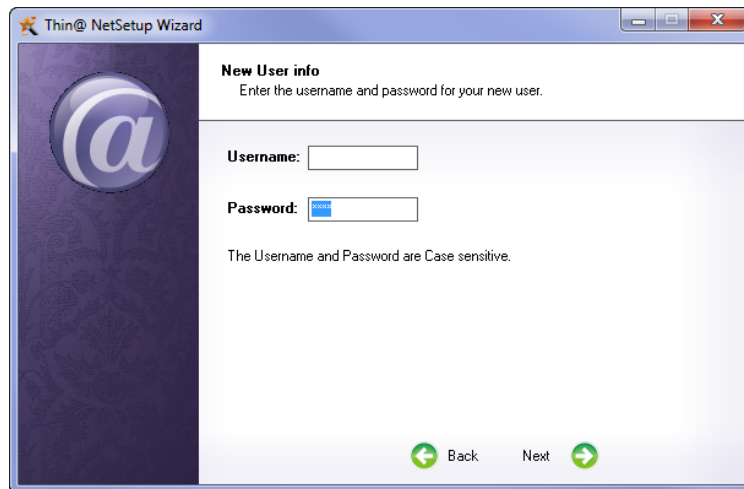


Figure 17. Setting the username and password

The NetSetup wizard relaunched again (I was beginning to see how this thing worked) and now gave me the option of adding a new Thin@ application. I hadn't actually created any apps yet, so I was at a dead end. There wasn't an obvious way to exit the wizard so I just hit the big red X. And the NetSetup admin window appeared (Figure 18).

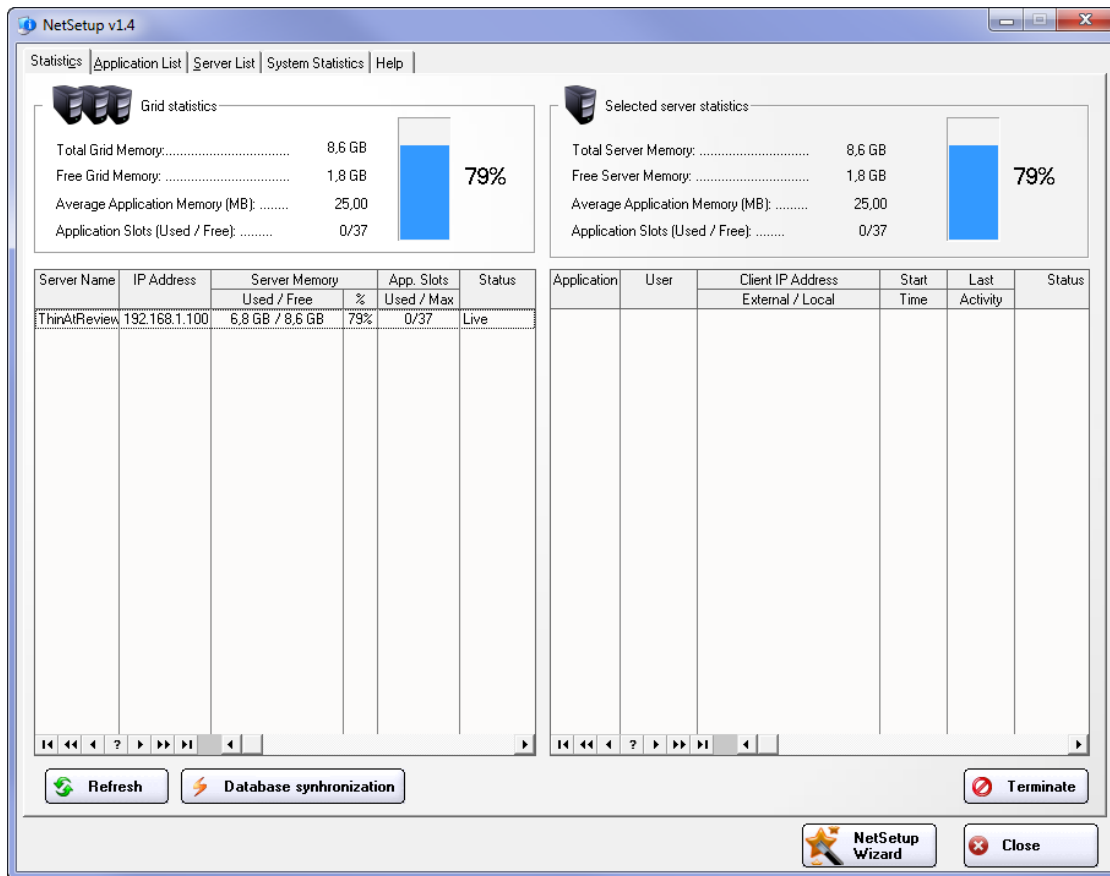


Figure 18. The NetSetup application

Comfortingly, there's a big button at the bottom of that screen to relaunch the wizard. And pay attention to that Database synchronization button - you'll need to press it whenever you make changes to the setup. I'm not sure why it's on the Statistics tab since you don't actually make any changes there.

Before I got too far into server admin I decided I'd better create an application for test purposes. I selected the old standby People app. Converting People into a Thin@ application was as simple as going to the Global Extensions tab and adding the Thin@ extension (Figure 26).

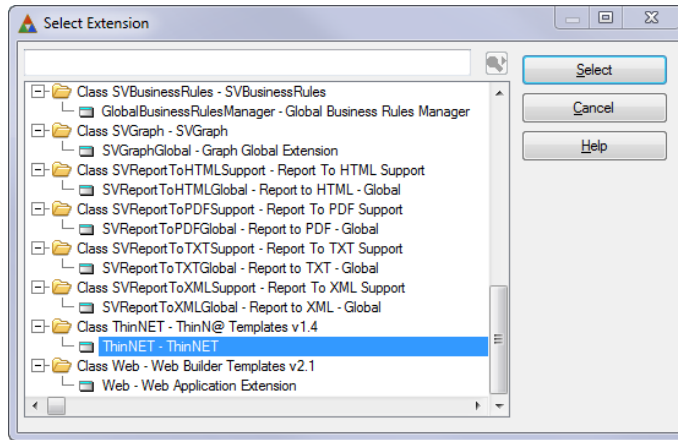


Figure 19. Adding the ThinNet global extension

The EXE compiled; now, how to add it to the Thin@ server? I returned to NetSetup and clicked the NetSetup Wizard button. I chose Add a new Thin@ Application (Figure 20).

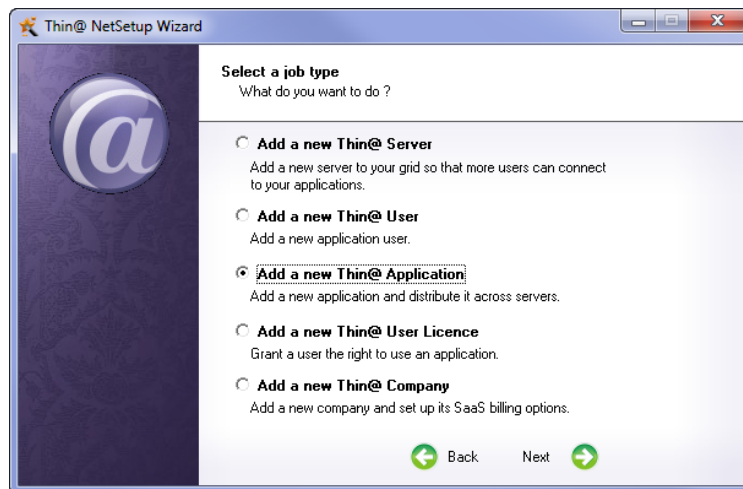


Figure 20. Adding a new application

I filled in the application prompts as in Figure 21.

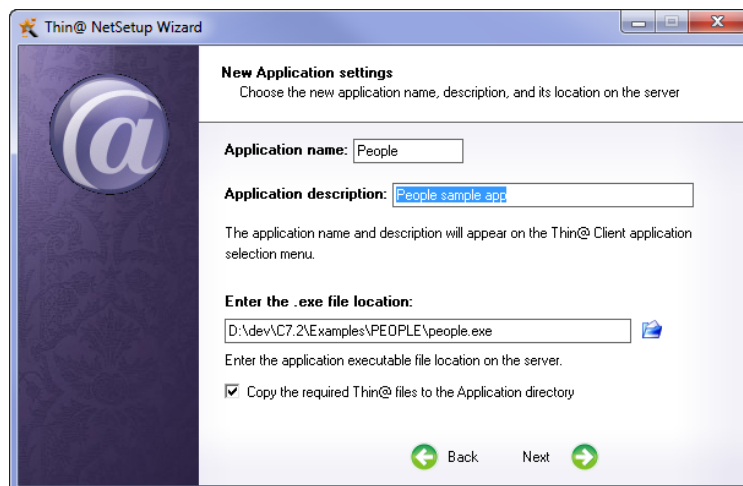


Figure 21. Application settings

I accepted the default inactivity settings (Figure 22).

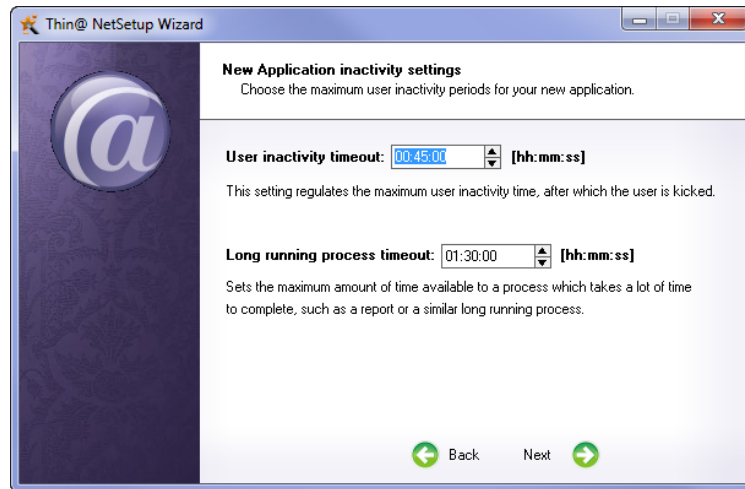


Figure 22. Inactivity settings

The wizard informed me that my application would be available to all servers on the grid. I only have one server, but it's nice to know that changes can easily be propagated across server farms.

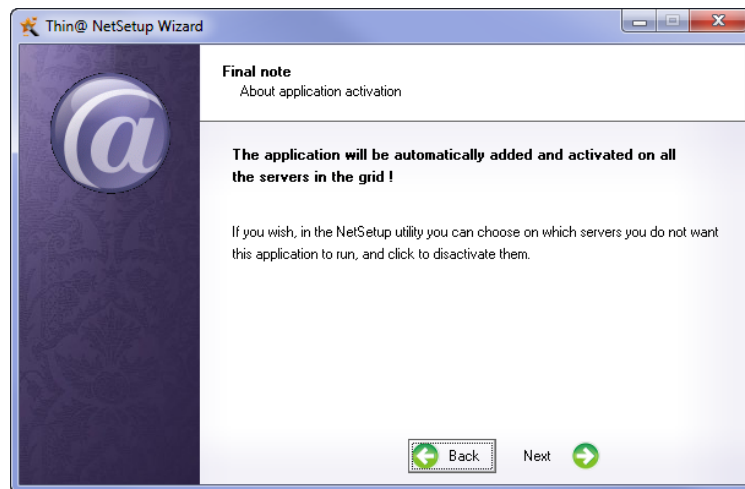


Figure 23. Activating the application

After the wizard completed I was asked if I wanted to commit my changes. I did, so I clicked Yes. That's the equivalent of pressing the Database synchronization button.

Now, how to run the application? Presumably I should be able to run the Thin@ client executable, point it at the server IP address and port number and everything would be good. So that's what I tried to do.

I found the Thin@ 7.1 client in my start menu and ran it. A popup message reminded me this was my first use and told me to check the server address. In NetSetup I located my server address as 192.168.1.100 (a local network address), and my server monitor port as 9000. I

entered these values and clicked on Refresh.

Nothing happened.

Problems

I'd seen Marko Golem viewing server logs in his CLDC presentation, so I went to the system tray and clicked on the icons for NetPort, NetListener and NetServer. All three showed green lights, and none of their message logs indicated any problem. It appeared that the client was asking the server for a list of apps, and the server's response was that it didn't have any apps available.

I discovered that I could get the application to show up in the list if I assigned a user to the app and then logged in with that user. That made sense, but I wanted to know how to get the app to show up without requiring a login.

Thin@ is a complex system with sparse documentation; there is helpful information, just not quite enough of it. Error reporting, particularly when it comes to communication between servers, isn't all that helpful. For the most part you're going to have to manually check the server statuses and the server logs.

Even with full docs and better error reporting, given the complexity of the system you will probably need some tech support assistance. Keep in mind the time differences - Croatia is on Central European Time, which is GMT + 0100. For North Americans (about half the Clarion community), Eastern Time is GMT -0500, so there's a six hour difference on the east coast and a nine hour difference on the west coast. If you're in Los Angeles and you encounter a problem right after you sit down to your desk in the morning, it may be that the Thin@ office has already closed.

I sent an email regarding the login issue around 10 a.m. Central Time; when I didn't get a reply within the hour, I assumed the Thin@ office had closed and I resigned myself to not getting a reply until the next day. But I heard back from Daniel a few hours later, at about 10 p.m. his time. He explained that if I added the "Everyone" user to the app it would become visible to anyone, no login required. That makes complete sense, but I couldn't find any reference to that technique in the administrator's guide.

But even when I added the "Everyone" user to the app I still couldn't see it in the client's app list. There was one other thing I had to do: click on the Database Synchronization button on the Statistics tab in the NetSetup program.

I wasted a bit of time on port 80 as well. By default, the Thin@ server listens on port 9000, but in many networks "security by default" applies, and ports are opened only if necessary. Port 80, the default HTTP port, is almost always available, so Thin@ comes with the NetPort server program, which is basically just a port forwarder. On my desktop port 80 is already in use, so I set up NetPort to use port 81. But really all I needed to do for testing was connect on port 9000.

I tried the DLLTutor application, and was able to turn it into a Thin@ app by applying the

extension to all the DLLs. The only problem I confirmed was that I couldn't exit the application by choosing File|Exit; instead I had to hit the red X. This left the application hanging and using up a connection. I couldn't find a way to terminate the app from inside NetSetup, although I could do so via the client program.

There was one other problem with available memory. To avoid overcommitting memory on my development machine I set the maximum allocated to 50%, which soon resulted in errors on the client side, reporting that no servers were available. This seemed a bit odd since the server programs were using very little memory. Increasing the maximum allocation solved the problem, however.

Performance & reliability

On a local machine I found performance quite snappy, and I would expect similar results across a local area network.

I had some difficulty getting Thin@ set up on the ClarionMag server for test purposes. This is a leased dedicated server, so installation was no problem, but I had a process running on port 9000 which interfered with Thin@'s default port settings. You may want to run `netstat -a` and `netstat -b` at the command prompt before you begin installation to verify you don't have any port conflicts.

After another discussion with Daniel I used NetSetup to alter the NetListener port to 8999 (using NetSetup), after which all my servers ran successfully. I added an app and applied the Everyone license.

After resynchronizing the database I could see the app in my client's app list (on my desktop machine) but on running the app I consistently got the "All servers are full" error message, even though no other instances were running. I changed the allowed number of instances to 10, resynchronized, and tried again. No joy. I shut down all the servers and restarted them. Same problem. So I emailed Daniel again. It was most likely after closing time in Croatia so a few hours later I posted again to the online forum.

Meanwhile I had another look at the demo apps Thin@ has online. You can do the same - just get the client application from the Thin@ site and use port 9000 and either `usa.thindemo.net` or `thin.ris.hr`, whichever is closer. (It would be nice if the server drop-down also set the port to the last-used value.)

Thin@ offers a high degree of performance tuning. All control templates now have an additional two items on the Action prompt. You can disable refresh on both Select and Accept events. Figure 32 shows the Action prompts for a browse box.

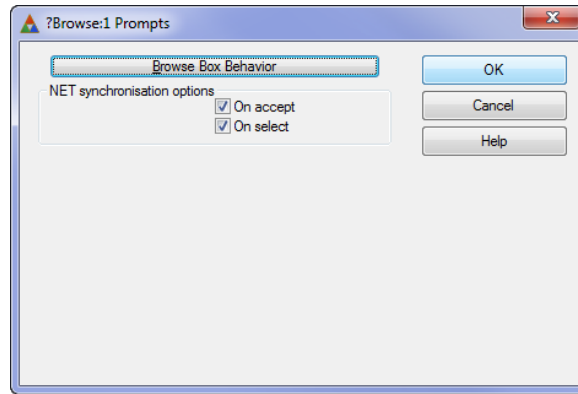


Figure 24. Fine tuning Thin@ apps

For instance, If you're not using hot fields on a browse you will probably want to disable refreshing on select. Leaving select on means that the screen is refreshed each time you click on a browse item.

Thin@ appears quite robust in dealing with network latencies. If you lose the connection entirely the Thin@ client will attempt to reconnect with the server, and when the connection is re-established you can continue on with your session.

Thin@ does *not* use the SoftVelocity AppBroker, but employs its own load balancing technology. The AppBroker's reliability has been something of a sore point for ClarioNet users.

Changes to your applications

You may need to make some changes to your application to get it to work with Thin@. Make sure all your menus have field equates or they won't show up. Disable or remove the date/time display extension template for the main frame; you can leave it there but it will trigger constant refreshing.

Only some third party extensions are supported. See the Thin@ docs for a current listing.

Thin@ works automatically with template-generated code but hand coded controls, such as listboxes you create or modify at runtime, will need some extra work. The programmer's guide explains how to tell Thin@ about your own listboxes.

As of this writing the unsupported features list includes:

- Date and time display on menu frames
- SVGraph template (CapeSoft's InSight Graphing *is* supported)
- Drag & drop functions
- Cell & Column tooltips
- BrowseGrid template
- ASCII viewer
- STD procedure calls (for example STD:CascadeWindow)
- Business rules

- Threaded report calls
- OLE ActiveX controls

MDI *is* supported.

Documentation

Five PDFs make up the Thin@ documentation:

- The Quick Start Guide is just one page and provides a very basic overview of the installation process. You may be able to get up and running after reading this if you don't have any port conflicts.
- The Installation Instructions (nine pages) walk you through the install process. Worth a quick look.
- The Product Activation guide is just one page and explains how to register Thin@.
- The Administrator's Guide is 30 pages and definitely worth reading, particularly for the extensive application usage monitoring and statistics.
- The Programmer's Guide (27 pages) details all the required application modifications (it's possible that none will be needed, but you'd best read this to be sure). This is a must read. Topics include: how to write Thin@-specific code (such as for uploading and downloading files, running server-side processes, making hand-written code Thin@-aware, etc), modifying the Thin@ client app (source code is supplied), and even running Thin@-ready applications locally in test mode *without* using the Thin@ server.
- Software as a Service (ten pages) introduces the concept of SaaS and focuses on economic benefits and business models. Thin@ is clearly designed not simply as a cool tool but as a business solution.
- System Architecture is an eight page document, and contains perhaps the best summary of Thin@'s features and benefits. It's less effective in explaining how the Thin@ system actually works, although it does give a general idea.

Origins

Thin@ is a product of RIS Software, a Croatian company. According to the [web site](#), RIS is an ISO 9001:2000 certified company founded in 1993 and employing more than thirty IT professionals. Daniel tells me RIS has installations with 600 users and five Thin@ servers, and everything works well over a 10Mbit connection.

RIS touts the following advantages of Thin@ to its clients:

- A three-tier architecture
- Higher security, since clients don't access the database connection directly (in other words, the data connection itself is never exposed so it cannot be compromised by a malicious app)
- Database clients (e.g. ODBC drivers) don't have to be installed on each client
- Newer versions are easily deployed

- Intranet and internet deployments work the same way
- Recovery from network errors
- Faster database transactions because the connection between the server and the db is usually much better than the connection between the client and the server
- User activity logging
- Session management
- Deployment to non-Windows computers (Mac, Linux) via the Java client (beta)
- ~100 users per server
- Less expensive than Terminal Server
- More users per server than Terminal Server
- Avoid TPS corruption

The beta Java client

As I mentioned earlier, Thin@ has a beta version of a Java client which will let you run your Clarion apps on Linux, Mac, and any other OS that supports Java. Figure 33 shows the People app with the browse and update form. I didn't try any complex apps, just the People app. The client crashed on me on one occasion, and when I tried to run a report I just got a wait cursor. But clearly it does work for some operations, and it looks very promising.

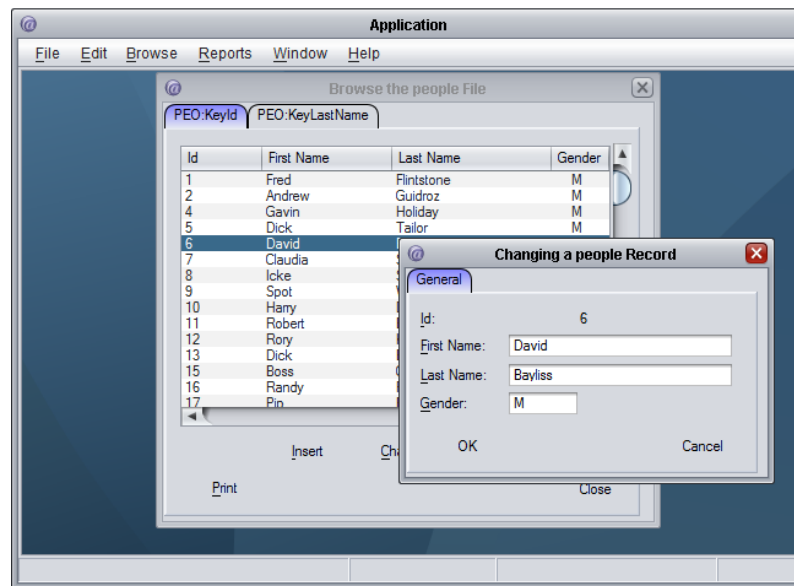


Figure 25. The Java client

The Java client is an applet - just load the page, wait a bit (a "loading..." message would be helpful for the impatient), and you'll see a client similar to the Win32 client. You can use it to load your own Thin@ applications or the demo applications, as you like.

Thin@ vs traditional web apps

For most Clarion developers looking to the web, the two main choices are smart client

technologies like Thin@ and ClarioNet and traditional web development tools such as CapeSoft's NetTalk. These two technologies for the most part serve different purposes. Traditional web development is optimal for public facing sites, where you can have high traffic volumes from users on all sorts of equipment, and where the user interface needs aren't complex. Smart clients are better for smaller, more controlled environments where users are prepared for and need a more complex user interface and where the available equipment (i.e. a computer able to run the smart client program) is a known quantity.

Traditional web app user interfaces have become significantly more complex with the advent of widespread JavaScript support. Better tooling has improved the developer experience significantly, creating some overlap with smart client development. In a Clarion context, the argument for a smart client approach in that overlap zone has a lot to do with ease of development and making the most out of existing applications and existing Clarion programming skills.

Conclusion

The idea behind Thin@ isn't new; credit for the first Clarion smart client goes to Michael Brooks and ClarioNet. But Thin@ takes Clarion smart client technology to a new level by removing the SoftVelocity AppBroker dependency and adding SaaS management tools. If the Java client proves successful, Thin@ will also have extended Clarion's smart client reach onto Linux and Mac computers.

Thin@ is relatively easy to set up and deploy, barring port conflicts. It will take some effort to fully learn all of Thin@'s capabilities (and you may not need the more advanced SaaS features), but even given Thin@'s sketchy documentation (which will be improved, Daniel tells me) that effort is small compared to learning a whole new programming environment. For installations up to hundreds or perhaps even thousands of users, particularly where you have existing applications you'd like to deploy remotely, Thin@ is well worth a look.

Cost: US\$600 for a single server license, US\$1150 for a three license bundle. A [free trial](#) is available.

Rating: ▲▲▲▲▲

Resources

- [Thin@ web site](#)
- [ClarionLive presentation](#)
- For CLDC attendees there are several recorded conference presentations on Thin@ - log in to the [ClarionLive DevCon site](#).

David Harms is an independent software developer and the editor and publisher of Clarion Magazine. He is also co-author with Ross Santos of *Developing Clarion for Windows Applications*, published by SAMS (1995), and has written or co-written several Java books. David is a member of the American Society of Journalists and Authors (ASJA).

Article comments

by Mark Riffey on November 28 2010 ([comment link](#))

I tried the beta java thin client on my MacBook Pro. The apps I toyed with for a short time worked just fine on the mac.

by James S. Smith on November 29 2010 ([comment link](#))

We've developed an application very successfully with Thin@. The original concept was developed in C7 and we intended to use Terminal Server. After testing Thin@ and doing some cost comparisons, we decided to switch to Thin@. There were a few setup issues and a few things we needed were not initially supported, but Marko was very helpful in getting these resolved (calendar lookup and Fomin reports). In the longterm support for additional Third Party templates/extensions would be welcome. Overall the stability of the application and server has been excellent and performance has been very good.

by Brian Reid on November 29 2010 ([comment link](#))

Thanks Dave for a great product review. This product has a lot of potential. My initial experience was very similar to yours. I have a significant investment in a terminal server data centre, and while it is a very robust platform upon which to publish clarion apps, there are administrative challenges (printing comes to mind) not to mention the \$\$\$ for MS CALS, security, etc. I've been working with Thin@ for a few weeks, and compared to the TS/RDP platform, Thin@ looks very promising as an alternative.

The SaaS administration appears to be very robust from our testing, and probably more advanced than most users need. One of the disadvantages of a typical RDP client session, is the lack of control or access to the user's desktop. The ability to move data silently between the server and desktop with a Thin@ client is a significant advantage. It offers an enormous opportunity for integrating the user's desktop to a hosted SaaS application. I have a dozen ideas that I could not get off the ground in a TS/RDP environment. They have grown new wings with the Thin@ technology.

The folks at RIS have been very helpful and receptive to suggestions for their product and have been very accommodating in making the Thin@ product compatible with existing 3rd party templates. That's a big ask considering the number of 3rd party stuff in circulation and to accommodate that, I understand they are developing a strategy for other 3rd party shops to simply include some code that makes their 3rd party templates Thin@ aware. This plug-in for 3rd party developers was in development a couple weeks ago, so it's best to ask either Daniel or Marko about the status.

For us it has already broadened our platform scalability, saved us money, and increased the life span of 12 years of existing clarion code.

I look forward to their future releases that include, the java client, mobile client, activeX control support, advanced load balancing and failover. It feels good to see new innovation happening in the clarion world.

In case it wasn't obvious, mark me down as sold on this technology.

Brian K. Reid Synova Group Canada

by Brian Roche on November 29 2010 ([comment link](#))

Very good review which echoes my experience with the product. I have found Daniel and Marko to be very responsive to support emails, (Even during the Clarion Live event) Thanks guys. I am very excited by the technology and am converting one of my main applications over to Thin@ - As described above the app is not really public facing and has users in the hundreds.

I am a customer already and look forward to developments with the product and the opportunity it presents for my business.

Brian Roche, Solient LLC.

by Dave Harms on November 30 2010 ([comment link](#))

Thanks guys for all your reports from the field!

Dave

by Wolfgang Kirsch on December 1 2010 ([comment link](#))

I have found the guys at Thin@ very helpful when I have run into problems. Overtime the biggest challenge is not so much that the product is not working, besides the known unsupported features/limitations. What time and time catches me out is the fact that ports get block by your firewalls, routers or even Microsoft own security.

One of the things I found very helpfull in this regards is the following two links, which checks if your port is 'really' open from outside your LAN to be accessed by Thin@. Which is rather difficult if you have no-one available to help you to check. Here is the links, thanks to Daniel Pavlic for providing then to me in the first place:

<http://www.yougetsignal.com/tools/open-ports/> (No software needed to download)

<http://portforward.com/help/portcheck.htm> (Software to download, but allows to check even for UDP)

Wolfgang Kirsch intelligent-monitoring.com

 [BACK TO TOP](#)

Copyright © 1999-2010 by CoveComm Inc. All Rights Reserved. Reproduction in any form without the express written consent of CoveComm Inc., except as described in the subscription agreement, is prohibited.

ClarionMag Tip of the Week # 1

By Dave Harms

Posted November 30 2010

Welcome to a new series in Clarion Magazine: The C7 IDE Tip of the Week!

Each week we'll bring you a new tip or trick to make your life in C7 easier. If you have a suggestion for the IDE Tip of the Week, [let us know!](#)

Tip #1: Add your application's dictionary to the solution!

When you open an application in C7, the app shows up in the Solution Explorer, along with all of the generated source file, file drivers, referenced libraries and so forth.

One thing that's conspicuous by its absence is the application's data dictionary. Yes, you can use Open File, specify the Data Dictionary file type, and load the dictionary that way. But it's nice to see the dictionary in the Solution Explorer. Fortunately, that's easy to set up.

Just right-click on the Solution Items entry in the Solution Explorers and choose Add | Add Item (Figure 1).

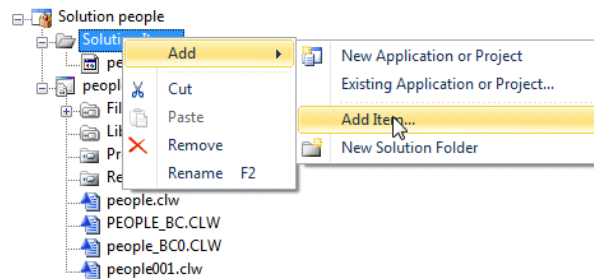


Figure 1. Adding the dictionary to the solution

Pick the dictionary from the file dialog and click Open.

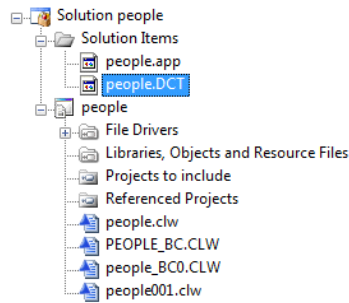


Figure 2. The dictionary in the Solution Explorer

Now your dictionary is listed in the Solution Explorer (Figure 2). To open the dictionary just double-click on it or right-click and choose Open.

David Harms is an independent software developer and the editor and publisher of Clarion Magazine. He is also co-author with Ross Santos of Developing Clarion for Windows Applications, published by SAMS (1995), and has written or co-written several Java books. David is a member of the American Society of Journalists and Authors (ASJA).

Article comments

by Michael Goosen on December 1 2010 ([comment link](#))

Thanks Dave! Great idea!

by William Tetley on December 1 2010 ([comment link](#))

Cool tip, thanks Dave!

by Vince Sorensen on December 1 2010 ([comment link](#))

Nice!

by Dave Harms on December 1 2010 ([comment link](#))

Thanks guys!

 [BACK TO TOP](#)

Copyright © 1999-2010 by CoveComm Inc. All Rights Reserved. Reproduction in any form without the express written consent of CoveComm Inc., except as described in the subscription agreement, is prohibited.

Creating SQL Applications With Firebird

By Dave Harms

Posted November 30 2010

There aren't very many truly free industrial-strength SQL databases. Leaving aside MySQL (which isn't typically free for commercial use), [PostgreSQL](#) appears to be the most popular choice among Clarion developers, with [Firebird](#) running a strong second.

In this article I'll show how to install and configure Firebird, create an ODBC data source and wizard up a Clarion application from the Firebird sample database.

Firebird's origin and features

Firebird's origins go back to Borland's Interbase. In 2000 Borland open sourced Interbase 6.0, and almost immediately the Firebird project appeared on SourceForge. Firebird developers began porting the original C code to C++. Firebird 1.0 came out in 2002, and supported Linux, Windows and OS X (Mac). But the first really stable release is considered to be 1.5, which came out in 2004. The current version is 2.5.

Firebird's features include:

- [ACID](#) compliance
- Referential integrity
- Stored procedures
- Triggers
- External functions
- Incremental backups
- Max table size of 32 terabytes
- Temporary tables
- Native Win32 version
- An embedded single user version is available

A number of versions of Firebird are available for [download](#), including 2.50.2.13, 2.06, 1.5.6, and 1.03. Recent versions are available for Win32, Win64, Linux x86, Linux AMD64, Mac OS X and Solaris. You can also download the source code if you're so inclined.

I downloaded the Win32 executable installer for version 2.5.0.26074. The installer is just 6.4 MB.

Choosing the right architecture

When you install Firebird you have to choose between the Classic and Super Server architectures. The Classic approach is a bit more like a TPS database in that each application can open the database files directly. Well, maybe indirectly: each accessing application invokes its own server process, and multiple server processes contend for access to the data files.

The Super Server is more like a traditional SQL server; there's a single server process that listens for and handles all requests. So you can be forgiven for thinking that Super Server is

the way to go.

Not so fast. The Super Server architecture can only make use of a single processor, so if you intend to scale up your database capabilities with a multi-core CPU you won't be able to reap those benefits. For scalability, go with the Classic architecture.

As of 3.0 you'll have another option: Super Classic. With Super Classic those individual server processes become threads, so you can take full advantage of multiple cores without the extra overhead of having individual processes for each client connection.

For now, think Super Server for small databases (which is a little counter-intuitive), and Classic for big databases. For the future, look to Super Classic. [Read more here.](#)

Installing Firebird

In Figure 1 I've launched the installer. Honestly, does anyone ever close all other applications before installing new software?



Figure 1. Installing Firebird

(This first screen aside, I've skipped some of the non-critical screens shown during the install process.)

You'll be asked to accept the license agreement. Following that you'll see some pre-installation information (Figure 2). Key points include:

- You are advised to uninstall all previous versions of Firebird or Interbase (which makes me wonder how you handle upgrades).
- You are warned that under some circumstances you may encounter problems with the Microsoft Visual C runtime libraries. Firebird will use locally installed versions if present, and otherwise will install the libraries. If you encounter problems you can manually install the VC libraries, following the directions in the document.
- The installer only installs the default instance; there is a zipped install image if you need to install additional named instances.
- The installer cannot reliably detect if a previous version of the Classic server is running.
- If you try to install both 32 and 64 bit versions you're going to have some trouble.
- Under Vista you must install as an administrator. That is, right-click on the EXE and choose Run as administrator.
- If you need to uninstall, do it via the Control Panel.
- On Windows 95 you will need to install the Winsock runtime components.

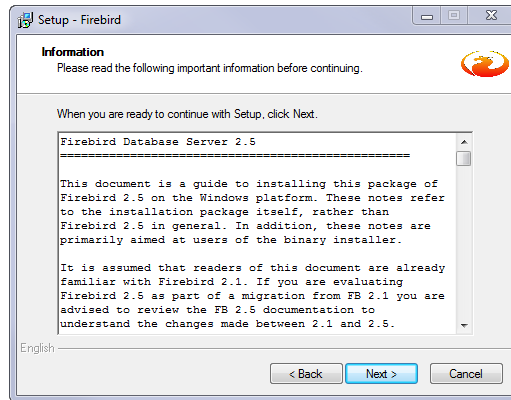


Figure 2. Install information

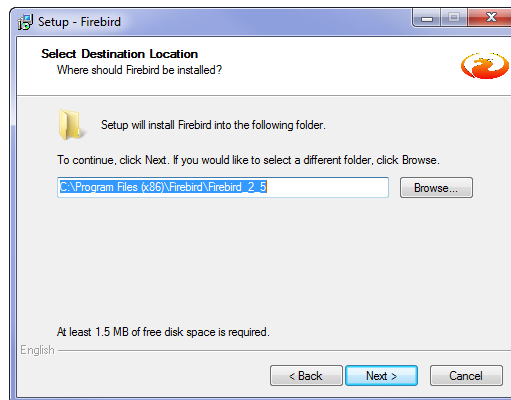


Figure 3. Choosing an install directory

Next you select the server components. I'd like to say that I installed the Super Server binary because I only needed a small database, but in fact I didn't adequately understand the differences between Super and Classic at the time. Not a big deal in this case, but you might as well make the right decision the first time around.

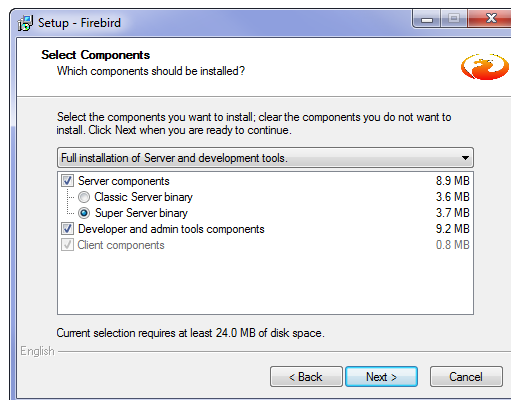


Figure 4. Choosing the components to install

The Firebird Guardian (Figure 5) is only relevant if you are running Firebird as an application - in this case I just accepted the default setting for that checkbox. For a development machine you might choose to run Firebird as an application, since it's only going to be used by you and you have to log in to use it. But on a server you don't want any important processes to be dependent on a user logging in. You want all those processes running as soon as Windows loads whether or not anyone's logged in, and that's what a

service will do.

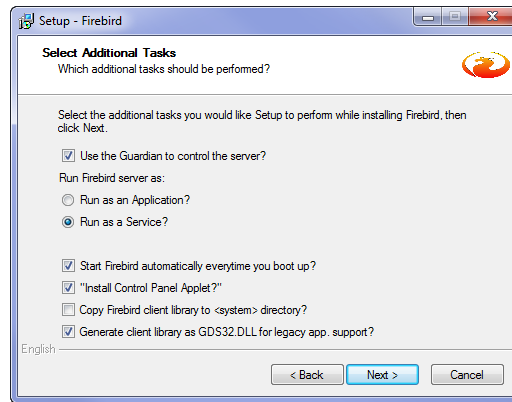


Figure 5. Setting startup options

After installation, if you've checked the After Installation - What Next? option (Figure 6), the installer launches the [After Install page](#) in your browser. This page tells you where to find the Quick Start Guide, which should also be somewhere in your start menu.



Figure 6. Completing installation

Setting up Firebird

You are presented with more information after installing, including:

- A disclaimer that, while Firebird is intended for production use, it is a sharp stick and you should avoid poking your eye out with it. In other words, read the release notes, back up existing data, and do your own testing.
- If you are on Vista and you installed Firebird as a server, you will need to start the service manually, or reboot.
- How to report a bug, or find out if your bug isn't a bug at all. You lamer.

You're done with the install. And no, the docs aren't really snarky. They're just too obviously written by the developers.

Administering Firebird

So, Firebird is running. But how do you use it? There's a sample database, according to the Quick Start. And there's a command line client called Firebird ISQL Tool which you'll see in your start menu. My first step was to try to connect to the database using ISQL.

Command-line SQL utilities can be handy, but they're generally pretty ugly and ISQL is no

exception. And there's another thing about ISQL that's even more annoying than usual: for some reason, although ISQL is running in a command window, right-click doesn't work so you can't paste in text. I fixed that by launching a command window from the Start menu and pasting in

```
"C:\Program Files (x86)\Firebird\Firebird_2_5\bin\isql.exe"
```

The sample database is in C:\Program Files (x86)\Firebird\Firebird_2_5\examples\empbuild. I don't really like to see data under the Programs directory, so I copied that folder to D:\Databases\Firebird.

In ISQL I executed the following command:

```
connect D:\databases\firebird\empbuild\employee.fdb user SYSDBA password masterkey
```

This connected me to the database using the SYSDBA user name (all usernames are in all caps) and the masterkey password.

I then issued a

```
show tables;
```

command and got a listing of tables in the database. And finally I tried

```
select job_code,job_grade,job_title from job;
```

Figure 7 shows the results.

```
G:\>"C:\Program Files (x86)\Firebird\Firebird_2_5\bin\isql.exe"
Use CONNECT or CREATE DATABASE to specify a database
SQL> connect D:\databases\Firebird\empbuild\employee.fdb user SYSDBA p
assword masterkey;
Database: D:\databases\Firebird\empbuild\employee.fdb, User: SYSDBA
SQL> show tables;
  COUNTRY
  CUSTOMER
  DEPARTMENT
  EMPLOYEE
  EMPLOYEE_PROJECT
  JOB
  PROJECT
  PROJ_DEPT_BUDGET
  SALARY_HISTORY
  SALES

SQL> show table job;
JOB_CODE          <JOBCODE> VARCHAR(5) Not Null
                  CHECK (VALUE > '99999')
JOB_GRADE         <JOBGRADE> SMALLINT Not Null
                  CHECK (VALUE BETWEEN 0 AND 6)
JOB_COUNTRY      <COUNTRYNAME> VARCHAR(15) Not Null
JOB_TITLE        <JOBTITLE> VARCHAR(25) Not Null
MIN_SALARY       <MIN_SALARY> NUMERIC(10, 2) Not Null DEFAU
LT 0
MAX_SALARY       <MAX_SALARY> NUMERIC(10, 2) Not Null DEFAU
LT 0
JOB_REQUIREMENT  <JOB_REQUIREMENT> BLOB segment 400, subtype TEXT Nullabl
e
LANGUAGE_REQ    <LANGUAGE_REQ> ARRAY OF 151
                 VARCHAR(15) Nullable
CONSTRAINT INTEG_11:
  Foreign key <JOB_COUNTRY> References COUNTRY <COUNTRY>
CONSTRAINT INTEG_10:
  Primary key <JOB_CODE, JOB_GRADE, JOB_COUNTRY>
CONSTRAINT INTEG_12:
  CHECK (min_salary < max_salary)
SQL> select job_code,job_grade,job_title from job;
JOB_CODE JOB_GRADE JOB_TITLE
=====
CEO      1 Chief Executive Officer
CFO      1 Chief Financial Officer
VP       2 Vice President
Dir      2 Director
Mgr      3 Manager
Mgr      4 Manager
Admin    4 Administrative Assistant
Admin    5 Administrative Assistant
Prel     4 Public Relations Rep.
Mktg     3 Marketing Analyst
Mktg     4 Marketing Analyst
Acct     4 Accountant
Finan    3 Financial Analyst
Eng      2 Engineer
Eng      3 Engineer
Eng      3 Engineer
Eng      4 Engineer
Eng      4 Engineer
Eng      5 Engineer

JOB_CODE JOB_GRADE JOB_TITLE
=====
Doc       3 Technical Writer
Sales     3 Sales Co-ordinator
Sales     3 Sales Co-ordinator
SRep     4 Sales Representative
SRep     4 Sales Representative
SRep     4 Sales Representative
SRep     4 Sales Representative
SRep     4 Sales Representative
SRep     4 Sales Representative
SRep     4 Sales Representative
```

Figure 7. Using ISQL

I'll use a command line tool in a pinch, but I much prefer using a GUI admin tool. Only there is no graphical admin tool included with the install, so you'll have to download one somewhere. Quite a few are listed at the [IBPhoenix site](#).

Using EMS SQL Manager Lite for Firebird

One of the best free admin tools is [EMS's SQL Manager Lite for Firebird](#) (also known as the freeware version). The lite version still has a lot of functionality, as you can see by the [features matrix](#).

The first thing you need to do is connect to the Firebird server; choose Register Host from the Database menu. You'll see the Register Host Wizard (Figure 8).

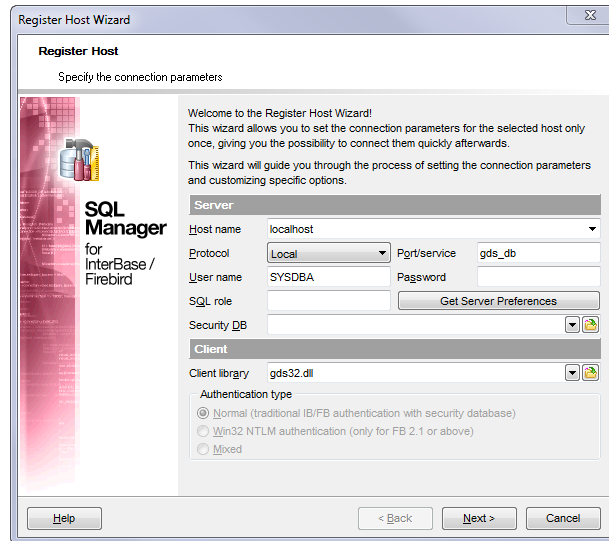


Figure 8. The Register Host wizard

The SYSDBA username will probably already be there; enter the password "masterkey" (without the quotes) and click on the Get Server Preferences button to load up the client library (Figure 9) and the path to the utilities (Figure 10).

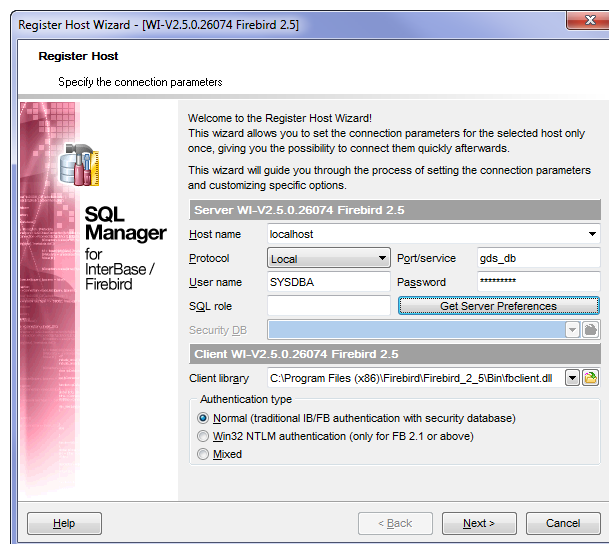


Figure 9. Getting the server preferences

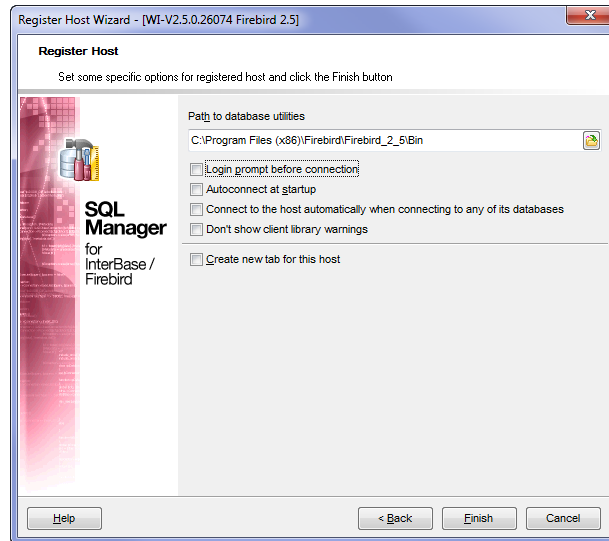


Figure 10. The utilities path has been set

When you've completed adding the host you'll see the localhost listed under Databases. Double-click on it to connect to the server. You'll see two users listed (Figure 11).

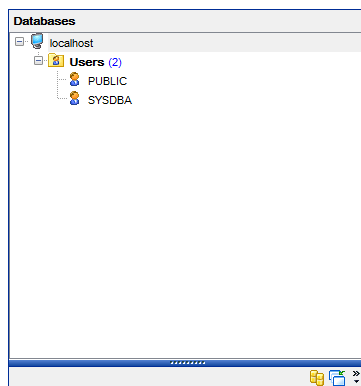


Figure 11. The user list

Now you're ready to connect to the database. Choose Register Database from the Database menu to bring up the wizard (Figure 12).

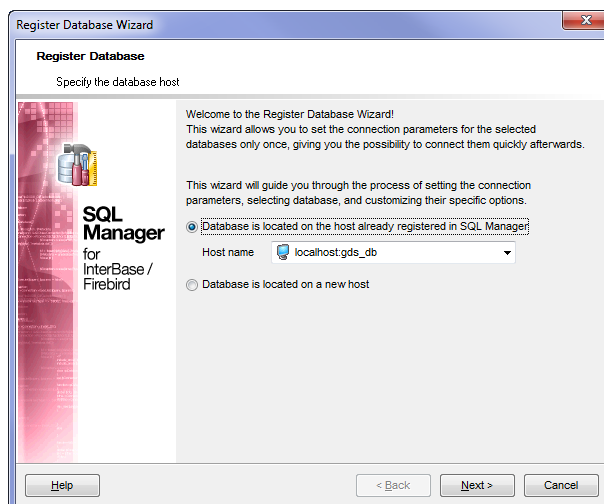


Figure 12. The Register Database wizard

On the second page of the wizard you enter the user name and password (masterkey is the default).

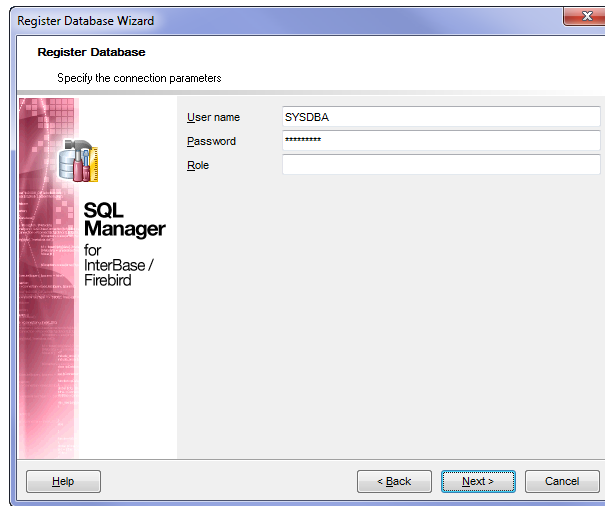


Figure 13. Entering the username and password

Either choose the database from the dropdown list, or if it isn't present in the list click on Browse to navigate to the database file. When you've selected the database click on Test Connection to make sure everything's good.

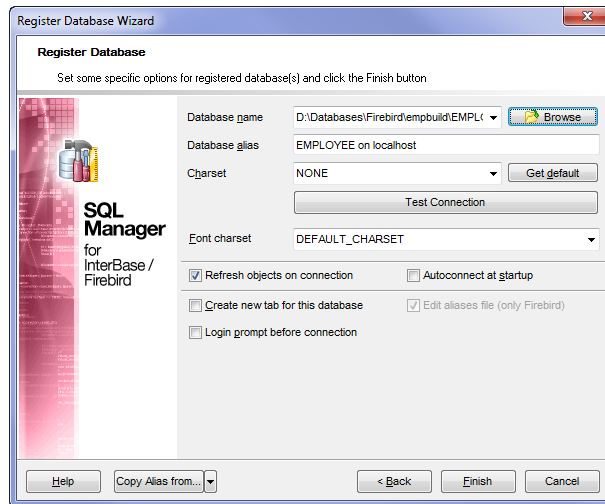


Figure 14. Completing database registration

Click on Finish and you should now see the database listed as in Figure 15.

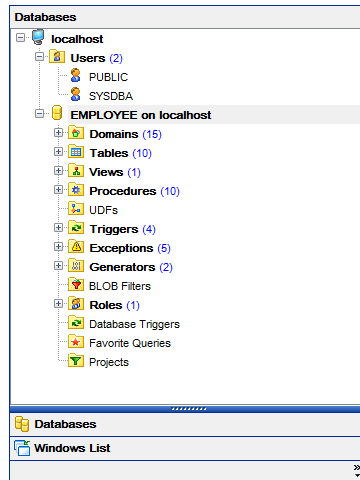


Figure 15. The registered database

At this point you can browse and modify the database. You're doing it as an administrator, and while that's fine for testing you might also want to set up a non-admin user. In Figure 16 I'm adding a new user.

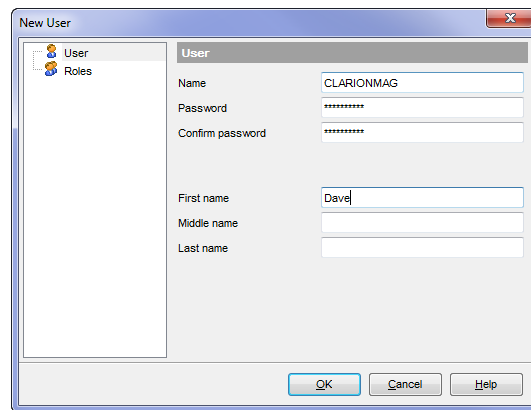


Figure 16. Adding a new user

Figure 17 I shows the Role Manager (available from the Tools menu).

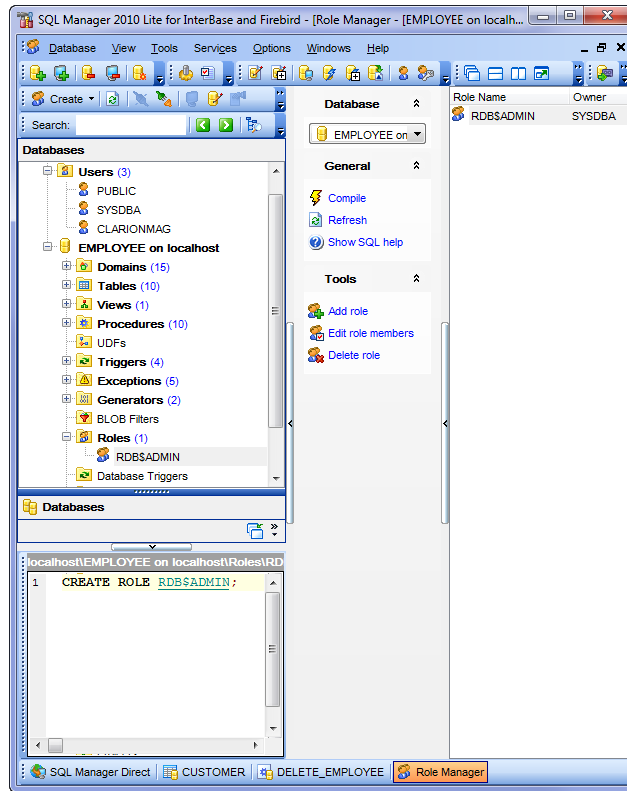


Figure 17. The role manager

Click on Edit role members to get the window in Figure 18.

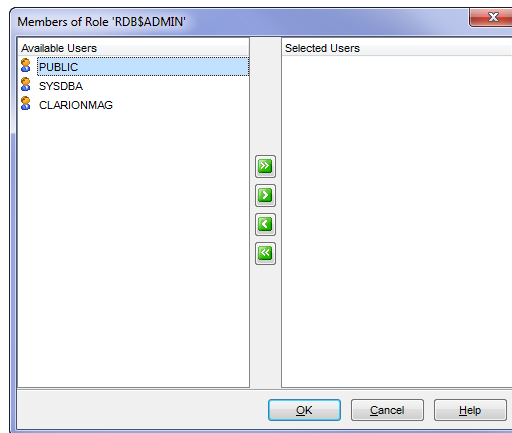


Figure 18. Adding a user to a role

Move the CLARIONMAG user over to the right side to add it to the RDB\$ADMIN role and clicked OK. RDB\$ADMIN is the admin role associated with this database, so this gives the CLARIONMAG user all of the admin rights.

I didn't see an easy way to actually administer role privileges, however. Perhaps that's buried deeper inside the admin utility. Certainly you can do this manually as well - see the GRANT and REVOKE topics in the Firebird help.

Finally, it's time to install the ODBC driver and then get on with creating the application.

The ODBC driver

I installed the 32 bit [IBPhoenix ODBC driver](#) which I downloaded from the FirebirdSQL site.

Installation is pretty straightforward, so I won't cover that here. Once you've completed installation you'll need to set up a data source.

In the administrator click on Add to create a new data source. Choose the Firebird/Interbase ODBC driver and click on Finish (which is of course just the beginning).

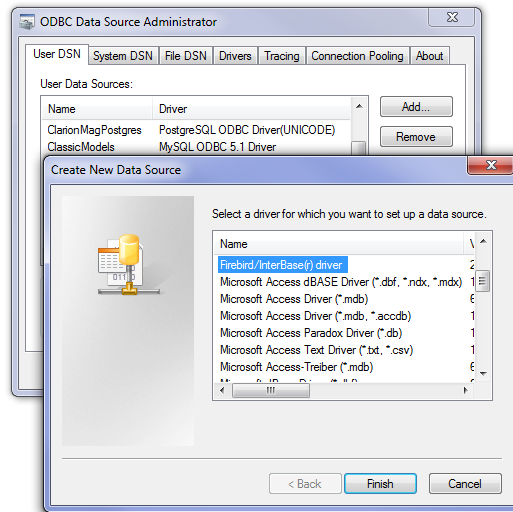


Figure 19. Adding a new data source

Figure 20 shows the setup window for the Firebird ODBC driver.

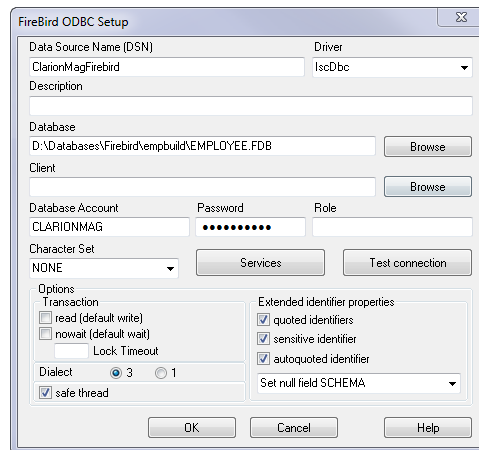


Figure 20. Firebird ODBC setup

If you click on Help you'll get a help file which contains a note on using the driver with Clarion (Figure 21). Note that in Figure 20 I've checked all the extended properties.

Using the Firebird ODBC driver with Clarion

Clarion users can work with mixed case object names in Firebird.

1. Create your database in Firebird. You can use table names like "Pending_Invoices" and fields like "Order_Number".
2. Create the DSN for the Database, making sure to check all options in "Extended Identifier Properties"
3. Open your dictionary, and import multiple tables as normal from the odbc source. It will work, but do not try to browse or use the files in an application yet.
4. For every field, type in the 'External Name' the name of the field surrounded by quotes (for example, type "Order_Number" in the external name).
5. That's it! Now use your dictionary with Mixed_Case_Identifier without problems. But remember - you must use double quotes around object names in all sql statements from inside Clarion.

Thanks to Jorge Andres Brugger, Vernon Godwin and Vladimir Tsvigun for the info contained in this document.

Figure 21. Clarion notes on the driver

Test the connection, save your changes and you're done!

Creating the application

Now you're ready to create a Clarion application. This is pretty straightforward. In the DCT explorer find the Import Tables menu option (Figure 22).

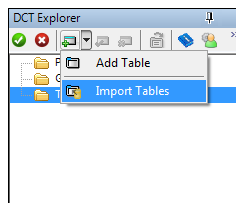


Figure 22. Importing tables

This brings up the table import wizard (Figure 23). Set the Server to ODBC.

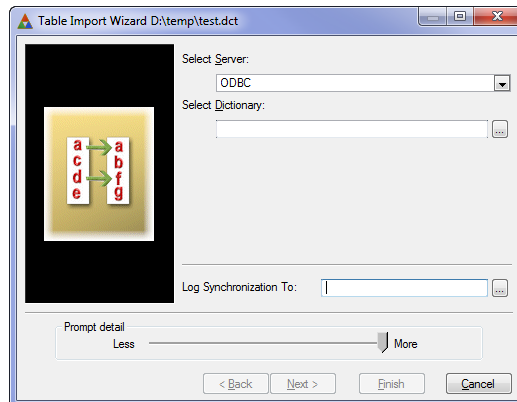


Figure 23. The table import wizard

On the next screen set the Datasource to the ODBC data source.

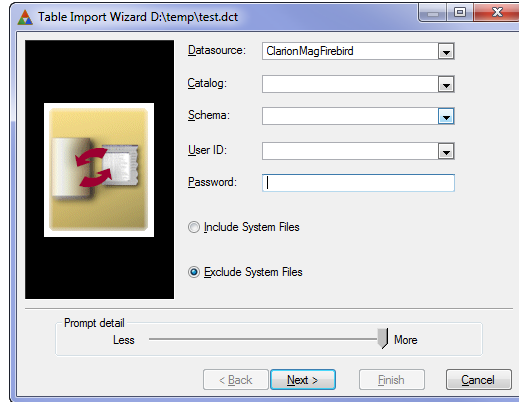


Figure 24. Setting the Datasource

And on the last screen add all the tables to the right side.

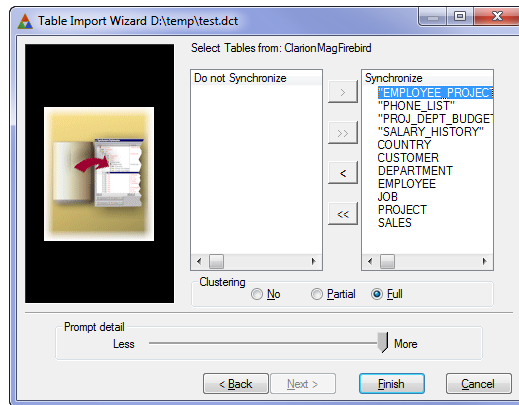


Figure 25. Adding the tables

Click on Finish, save the dictionary, and wizard up a new app. Figure 26 shows the finished running application.

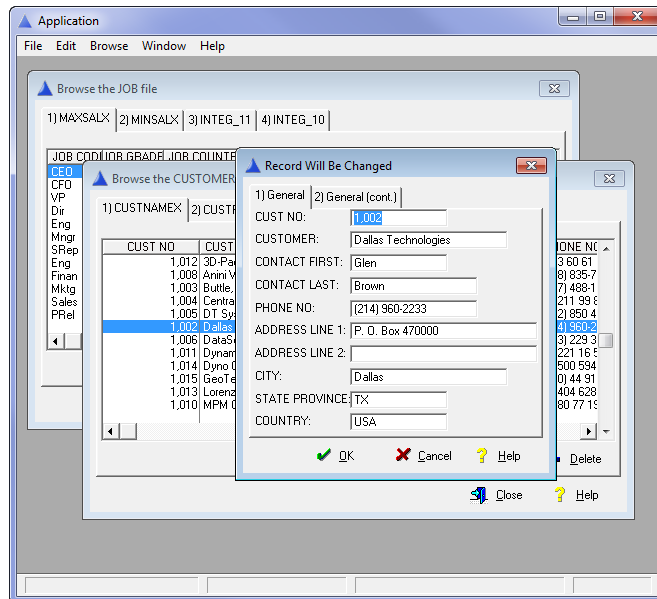


Figure 26. The wizarded application

Summary

Firebird is one of only a small number of free SQL servers, and a popular choice for Clarion developers. It's easy to install, and although it comes with only the most rudimentary admin tools there are a number of free and paid products that fill the void. Database importing, in my tests at least, worked smoothly, and the free ODBC driver also gets the job done when it comes to handling data.

[Download the demo app](#)

Resources

- [Firebird Home](#)
- [Firebird Portal](#)
- [IBExpert free edition](#)
- [EMS SQL Manager Lite for Firebird](#)

David Harms is an independent software developer and the editor and publisher of Clarion Magazine. He is also co-author with Ross Santos of *Developing Clarion for Windows Applications*, published by SAMS (1995), and has written or co-written several Java books. David is a member of the American Society of Journalists and Authors (ASJA).

Article comments

[↑ BACK TO TOP](#)

Copyright © 1999-2010 by CoveComm Inc. All Rights Reserved. Reproduction in any form without the express written consent of CoveComm Inc., except as described in the subscription agreement, is prohibited.