

Centura Pro

Visit us at www.ProPublishing.com!

Hot Ideas for Centura® Developers

CTD 1.5: Beyond Active X

R.J. David Burke

Centura Team Developer (CTD) 1.5 is the latest release of Centura's popular development tool for building Windows-based database applications. Although the most significant area of change in this release revolves around extensive support for ActiveX/COM for SAL applications, other parts of the product have been revised as well. I'll focus on those in this article, then look at the new ActiveX features in CTD 1.5 in an upcoming issue.

SQLWindows: politically correct again

Centura Team Developer encompasses several related tools. Centura Team Builder was the particular tool that replaced 16-bit SQLWindows. Now Centura management has decided that it makes sense to show continuity in that tool, and the new name for Builder is SQLWindows/32.

While ActiveX features steal the show, there's more in Eiger that seasoned developers will welcome (and even welcome back!).

Starter Application Templates

In previous versions of CTD, choosing File | New from the menu system (or clicking on the New button in the toolbar) would clear out any existing outline file (prompting you to save if necessary) and load NEWAPP.APP, or whatever file you specified in Preferences

as your starter application. Now, with CTD 1.5, you can define multiple starter applications and select whichever one is appropriate to the application you want to start writing. In CTD 1.5 these are known as Starter Application Templates. For example, you could develop several different templates with their own library files included or outline items already defined, then just save the template in the templates folder. By default, the templates folder is a subdirectory called "Templates" off of the folder where you installed CTD (typically C:\Centura). You can change the templates folder using the Directories tab of the Preferences dialog as shown in Figure 1.

Once the template is saved to the templates folder, it's easy to access from the File | New... menu (or the Ctrl-N shortcut). In CTD 1.5, The File | New... menu brings up

December 1998

Volume 3, Number 12

- 1 CTD 1.5: Beyond Active X
R.J. David Burke
- 2 ... And How Does That Make You Feel?
Mark Hunter
- 5 A Talk with the Boss
Mark Hunter
- 8 Put NetWare to Work
Joachim Meyer
- 12 Ace! Centura App Detective: Unix, Quest, and More
Ace Centura

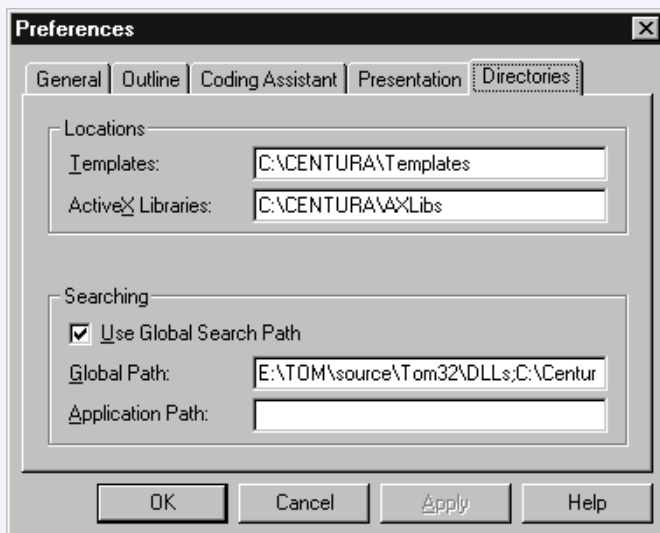


Figure 1. The Directories tab of the Preferences dialog lets you change the templates folder among other settings.

Continues on page 3

... And How Does That Make You Feel?

Mark Hunter

Centura Software was well prepared for the Customer Conference in Orlando. They chose an excellent site, the Grand Cypress Resort. The conference czar, Kerrie Peck of the Genesis Group, kept everything running like a well-oiled machine. The presenters were all completely organized, and none of their demos crashed (well, not often, anyway). There was a lot to see and learn, and ample opportunities to talk to the Centura employees of one's choice.

The only thing they forgot was the therapist.

Centura's CEO, Scott Broomfield, must surely have wondered what was going on. Here were a few hundred customers, fiercely loyal to the products, treating the company management like pariahs. In November, while commenting on some of Centura's new product directions, I asked, "What does this mean for the development tools? Maybe the Orlando conference can shed some light." Apparently, lots of others asked the question, too. The weird part is, they were not really listening to the answer.

To be fair, there was a time, early in Broomfield's tenure, when plans were discussed to dump SQLWindows and CTD. And there were some early actions that seemed to reinforce that, particularly the cancellation of the Tomahawk project. But Broomfield became convinced that the tools were essential to Centura's long-term strategy, and he shifted gears. The first sign of the new direction was the release of Team Developer version 1.5. More recently, Broomfield's

acknowledgement that 1.5 didn't meet its customers' expectations and his promise of a more satisfactory 1.6 in the near future demonstrated his continuing involvement with the tools. At the conference, much attention was given to the feature set of CTD version 2.0, which will provide the ability to generate COM servers from SAL code.

In spite of these tangible accomplishments, many customers kept worrying about the future of the tools—so much so that even one of Centura's Customer Advisory Council members, on stage in the closing session of the conference, pronounced himself only recently convinced

Continues on page 11

Frosty the Editor Mark Hunter, In Her Cap Dian Schaffhauser, In Her Kerchief Shelley Doyle, Little Layout Boy Paul Gould, Mocha the Red-Nosed Doggie

Centura Pro (ISSN: 1093-2100) is published monthly (12 times per year) by Pro Publishing, PO Box 18288, Seattle, WA 98118-0288.

POSTMASTER: Send address changes to Centura Pro, PO Box 18288, Seattle, WA 98118-0288.

Copyright © 1998 by Pro Publishing. All rights reserved. No part of this periodical may be used or reproduced in any fashion whatsoever (except in the case of brief quotations embodied in critical articles and reviews) without the prior written consent of Pro Publishing. Printed in the United States of America.

Centura Pro is a trademark of Pro Publishing. Other brand and product names are trademarks or registered trademarks of their respective holders.

This publication is intended as a general guide. It covers a highly technical and complex subject and should not be used for making decisions concerning specific products or applications. This publication is sold as is, without warranty of any kind, either express or implied, respecting the contents

of this publication, including but not limited to implied warranties for the publication, performance, quality, merchantability, or fitness for any particular purpose.

Pro Publishing, shall not be liable to the purchaser or any other person or entity with respect to any liability, loss, or damage caused or alleged to be caused directly or indirectly by this publication. Articles published in Centura Pro reflect the views of their authors; they may or may not reflect the view of Pro Publishing. Opinions expressed by Centura Software employees are their own and do not necessarily reflect the views of the company.

Subscription information: To order, call Pro Publishing at 206-722-0406. Cost of domestic subscriptions: 12 issues, \$119; Canada: 12 issues, \$129. Other countries: 12 issues, \$139. Ask about source code disk pricing. Individual issues cost \$15. All funds must be in U.S. currency.

Call Centura Software Corp. at 650-596-3400.

If you have questions, ideas for bribing authors, or would just love to chat about what you're doing with Centura products, contact us via one of the means at right.

Talk to Us

Centura Pro on the Web
<http://www.ProPublishing.com>

Editorial Department

Phone: 818-249-1364
 Fax: 818-246-0487

E-mail: mhunter@sprintmail.com

Subscription Services

Phone: 206-722-0406
 Fax: 206-760-9026

E-mail: dschaffhauser@cwix.com

Mail

Pro Publishing
 PO Box 18288
 Seattle, WA 98118-0288

A Peek into CTD 1.5 ...

Continued from page 1

the Templates dialog, as shown in Figure 2.

With the Templates dialog, you can choose any defined template by double-clicking on it (or clicking on it and then clicking on OK). You can add new templates by clicking on the Edit button to expand the dialog and then using the fields at the bottom to specify the template file and description.

One of the templates is defined as the default template and is indicated by a little checkmark on the icon

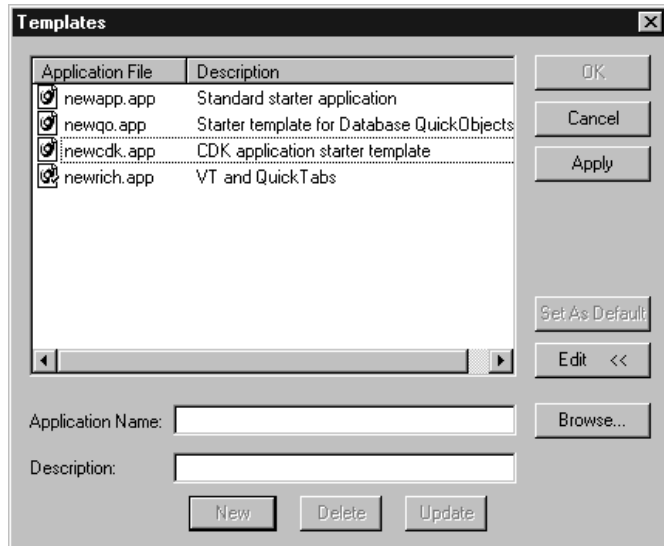


Figure 2. The Templates dialog (shown in expanded form) lets you select a starter application template from the templates you've defined.

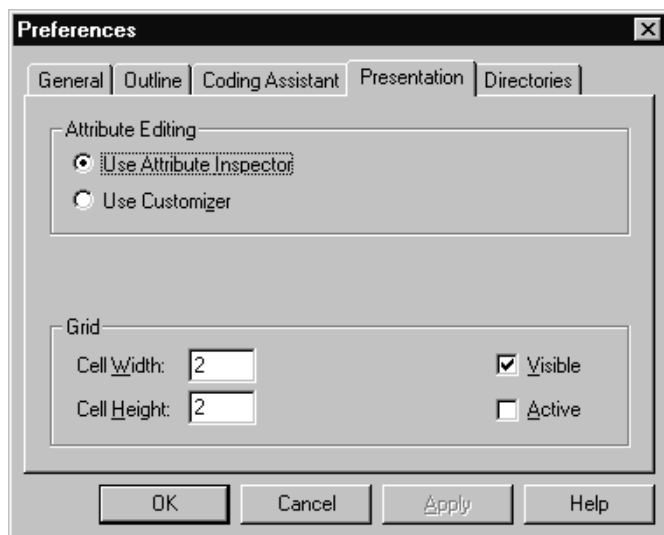


Figure 3. The Presentation tab is used to switch between the Customizer or the Attribute Inspector, among other preferences.

(left of the filename). The default template is the one used when you click the New button on the toolbar.

Application searching

With earlier versions of CTD and SQLWindows, you could define a search path for locating included library files that weren't qualified with a directory path. This capability is still in CTD 1.5 and is called the Global Path. CTD 1.5 also introduces Application Paths, as seen in Figure 1. Application Paths are specific to an .APP file. When you specify a library to be included *without* a directory path, the application path is searched first and then the global path. Additionally, you can disable searching the global path by unchecking the "Use Global Search Path" check box.

The search path is also used to locate other included files. For example, it can be used to locate image files for SAL resources, pictures, and push buttons.

Attribute Inspector or Customizer?

As you can see in Figure 3, the Presentation tab of the Preferences dialog is used to specify whether you want to work with the classic Customizer, or the new Attribute Inspector.

The Attribute Inspector (see Figure 4) replaces the Customizer for assigning attribute settings to windows, controls, and classes. The Attribute Inspector takes a little getting used to after using the Customizer for years, but I've settled with it. For diehards, you can continue to use the Customizer with this dialog, but be forewarned: You'll need to use the Attribute Inspector to set ActiveX properties.

Another reason to use the Attribute Inspector is that it can be "always there." Unlike the Customizer, the Attribute Inspector can maintain a persistent presence

until you decide you want it out of the way. The Attribute Inspector also supports an Undo feature so you can easily back out any changes you decide you don't want. The Attribute Inspector can be docked inside the SQLWindows/32 design-time environment or can be a free-floating window,

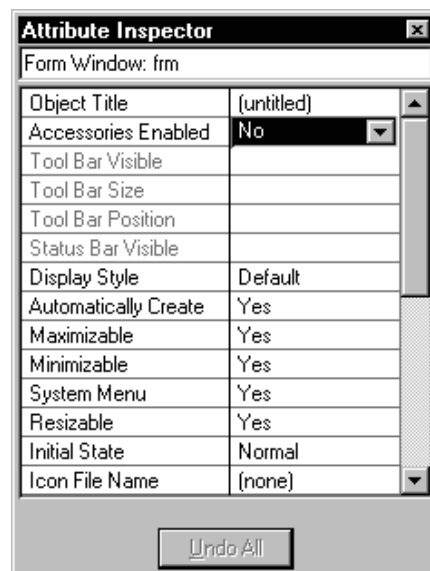


Figure 4. The new Attribute Inspector.

just like the other toolbars. The Attribute Inspector should be comfortable for developers who have worked with other development environments.

An improved Preview Mode

A frequent complaint about Preview Mode in CTD 1.0 and 1.1 was its modality. In CTD 1.5 Preview Mode is much more useful and most of the modality is eliminated.

There's no longer a need for the Preview window; the important tools that you need to work with design windows are all accessible from SQLWindows/32, as shown in Figure 5.

An oft-requested feature is for the ability to select a control in the outline that then selects the corresponding control in the design window and vice versa, just like in SQLWindows 5.0. This feature has returned in Preview Mode with CTD 1.5, and longtime SAL developers everywhere will welcome it back. You can even have multiple Preview windows, but each must be launched from a different outline view.

Customizing the Controls Palette

Another nice feature is the ability to customize the Controls Palette, as shown in Figure 6.

In addition to support for ActiveX controls and containers, you can also add buttons for your own SAL classes to the Controls Palette. For example, if you have a frequently-used child table window class, you can click the usual child table window button, and in the Control Palette's list, you'll see "Standard" plus any child table window classes that have been defined. Use drag and drop to move the desired child table window class to the middle group of buttons on the Controls Palette and it will create a button for your child table window class. Now click the new button whenever you need to drop an instance of that child table window

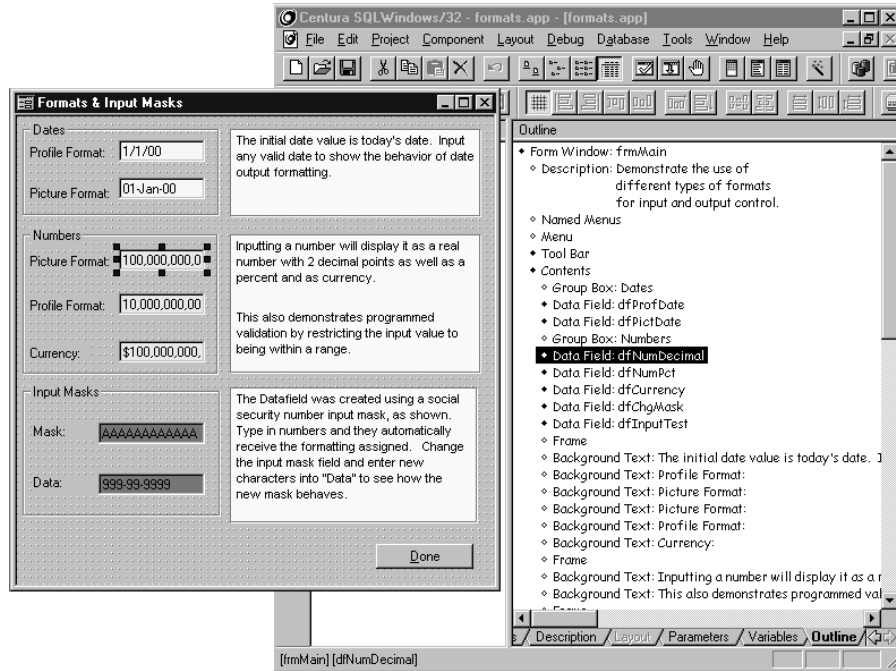


Figure 5. Preview mode in CTD 1.5 loses its modality and restores functionality found in SQLWindows 5.0.

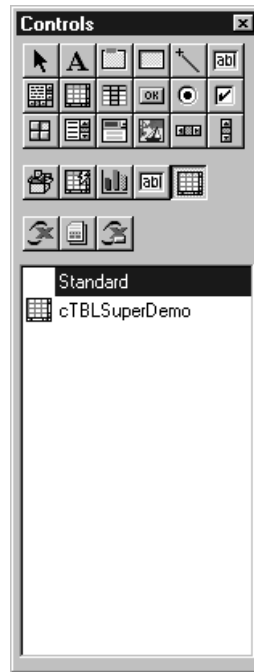


Figure 6. You can now customize the Controls Palette by creating buttons for your own SAL classes. See the middle group of buttons for a data field class button and a child table class button on the right end.

class onto a form. Additionally, any subclasses of the child table window class are listed in the Control Palette's list box so you can easily see which classes are derived from a specified class.

New functions for picture controls

While there are extensive new functions for ActiveX support, CTD 1.5 also introduces two

new functions for working with picture controls. `SalPicSetImage` and `SalPicGetImage` allow developers to work with just the image content of a picture, eliminating the usual Centura image header. (This eliminates the need to use my technique discussed in "Get That Picture Back" from the May 1998 issue of *Centura Pro*.)

More to come!

Of course the real reason for CTD 1.5 is the new ActiveX support, and that will be the most compelling reason for you to upgrade. But these other new features are a nice bonus and most welcome for improving developer productivity. More later! **CP**

R. J. David Burke is a Senior Consulting Engineer with Centura Software Corp. Reach him through *Centura Pro*.

A Talk with the Boss

Mark Hunter

Is Centura's interest in its development tools decreasing?

The conventional wisdom, when I joined the company, was that you couldn't make money in the tools. But we [do] make money in the tools. Not only are they financially advantageous to us, even though it's not a huge percentage of the company—it's about 25 or 30 percent of revenue...we also find, and I have found, that the tools are strategic. Let me explain that a minute.

Irrespective of what we want to do with the database, which is embedding it and scaling it down...it still requires a development environment to get the data down to the client, even if the client is just a device. If you think you're targeted that way, then you start asking yourself, "What does the tool need to be?"...it needs to be componentized and re-architected to where it can run in a thinner way at runtime...thread-safe and these kinds of things. That's the same thing as requiring a tool for an enterprise solution. So I look at that and I say, you know, we really need to keep developing the tool along with developing the database and do some unique things as we position the company to make sure that we can play in this other area without getting our heads smashed with respect to Microsoft and Oracle.

It's financially advantageous for me to be in this space today. I've stopped the slide on an invoicing basis; that's something as simple as, I released a *product* [Centura Team Developer 1.5]. The message that the company had been sending to its developer community back in 1997, maybe 1996, was, "We're going to get out of the tools business," even if it wasn't formally stated as a policy of the company. And when I came on board, the conventional wisdom of the [board of directors] was that we should get out of the tools business. I evaluated it and said we should take this thing and see if we can make something out of it. I got to that decision by talking to my customers. Even though we don't have market share, we can make some headroom in this area.

Centura Pro caught up with Scott Broomfield, CEO of Centura, just before the start of the 1998 Customer Conference. Like our readership, we wanted to know what—and how—he thinks. Here's the interview, commencing with the question that everyone seemed to have on their minds.

That's looking at it financially and strategically. Let me look at it competitively. I think PowerBuilder shifted its gear. And I think, given the new CTO that I hired [Joe Falcone, from Inprise], Delphi's gonna shift its gear. He knows what their plans are, and he said "I don't want to do this, I want to be [at Centura]". So I keep asking myself, if PowerBuilder is a little bit off center, and if Delphi is going to go the Visigenics way, isn't

there an opportunity, even if we live in a Microsoft world, to be successful, and to maybe even grow a little bit? I think its about simply bringing the product back to parity, on a feature-for-feature basis, and making sure we keep the lead in the programming environment: the SAL language, the outliner, the ability to use the tool in a heavy, mission-critical environment.

[At the conference tomorrow] you'll see a demo that brings together what we call the D3 Framework. That framework is as much about the thinking of what we're trying to do as it is about the products. It's the way we're trying to weave the products together over time. I think the tool is critical to making this be successful. We'll show you what we want to do in 1.6 and 2.0, and then what we call Phase 2 and Phase 3. What I think the company has not done up to this point is to describe where we are really going with this, stuff with respect to CORBA and EJB. Right now my installed base needs the COM stuff, so that was the first selection. The cancellation of the [Tomahawk] Java stuff last year was so that I could get a product out this year. If I had stayed with the Tomahawk project, I would have a product out in mid-1999. I don't think the installed base would have waited that long.

So far, you have been announcing less ambitious plans, and delivering them sooner, than past management. Is that a matter of personal style for you, or just a case-by-case decision?

There are some arguments in the company that I resist tremendously. The industry is known for pre-announcing things. I know that in certain things you have to give directions where you're going. But I resist just throwing stuff out there, just to see, because I'm not comfortable personally unless I know the product is in development, I've seen some examples of it working in a proof-of-concept way, I know how it integrates with the strategy. And then I can stand up there with some credibility, because the thing I had to do this year was to begin to rebuild some credibility. My natural style is to not be too far out ahead of myself.

By the way, there's a constant balancing act here. We did, in my judgement, a little bit of a premature announcement on Smart Cards. We announced it after we had made the decision to do it, but before I had seen the

The thing I had to do this year was to begin to rebuild some credibility.

beta version. We knew the conference was coming up shortly and we wanted to signal to the marketplace that we were doing something in this space as it relates to the database and the application server. It was a seeding mechanism—let's grab the people who are interested in Smart Card development—and that's exactly what it accomplished. You'll see a demo of it tomorrow. We need to understand that other people take a strategic position in their own companies predicated upon a commitment you make.

If you're a quarter or two late, their products don't get to market. We have a little bit of time to rebuild Centura's credibility—maybe a year, two years max. The way to rebuild that credibility is you try to deliver on those commitments.

Is your tools message getting across?

That's a good question. In a partners conference in Australia, I spent an hour and a half in the morning explaining the strategy, and how the tools fit, and what we were doing with 1.5, and what we wanted to do with COM generation, and some of our ideas past 2.0. Still, at the end of the day, during question and answer, several people asked, "What are you going to do with the tools?"

I've told them what in my marketing, I've said it personally. I've actually delivered an update to the tool.

On the Web site I've said it. To my sales force I've said it...We've spent over five million dollars this year on ads alone; on us, on the database, and on the tools. I finally realized at a gut level what the company created historically. I call it the "beat dog syndrome"—I raise my hand, and everybody flinches. I'm hoping that a lot of that noise is gone once and for all after the conference.

Are the development tools too expensive?

We emphasize the deployment aspects of the database because that's an on-going revenue stream. Right now we tend to make [tool pricing] difficult on the front end, but we don't get anything on the back end. I'd love to find a way to change the business model to make it easier on the front end and share more on the back end.

It's a tough one for me—I *know* it's more expensive than some of the other things that are out there. The percentage of cost of the tool, as it relates to an entire development project, is *tiny*. If you're on Centura License Support, you get CTD 1.5 for 20 percent of full price. But for the first-time user it can be more expensive. So one of the things I'm trying to do is build a starter kit bundle; all the products—here's Blazer, here's net.db, here's SQLBase, here's CTD, etc.

What's "new" about the new Web site?

We've actually improved the way customers interact with one another, with the new chat rooms. They can also begin posting applications on there, so that people can understand, "What have you done with this stuff?" And we've improved the navigation of the site, so that you can get anywhere with just two clicks.

Also, looking at the previous home page as an outsider, it was *hard* for that person to understand who we were. So we did three little things: We gave it a tag that says "embedded databases and application tools," a tag for the D3 framework, and a section for the "evolution of client-server."

What motivates the Centura employees who stayed, and those who left?

Even the people who are now staying were leaving back when I joined the company, because the company was struggling; that's all there was to it. And the job market is competitive in Silicon Valley. When I joined the company there were three things I looked at: its products, the markets it played in, and its employees. Were there enough good employees to build on? The answer was yes. When I framed the strategy for the company, I gave it to the best 15 or 20 employees – the best thinkers. They got a copy of the board presentation. That was back in January and February 1997. Since then, it's been a different mix of departing people. Some people just aren't up to the task—there's pressure in the company, I think, constructive

pressure. Also, there are other opportunities that cause people to leave. We backfill with strong people. I don't think people leave today for bad reasons, thinking we've lost our way; that's all gone.

When I got here I clamped down on spending, and I clamped down on head counts, because I had to; this company had a gun to its head. We had a plan ready to go the day we recapped the company [February 1997], and we turned it on that very day. We've gone from 190 people to 240 people. I'm not doing that for cash flow; I'm doing it to grow.

Your personal reward is heavily tied to the stock price of the company. Is that true for others joining the company as well?

Oh, you bet. Joe Falcone is coming in at a much lower base salary than his previous position, with 160,000 shares, and no bonuses or severance structures in his deal. We have one constituent from the recapitalization who is steadily selling shares, keeping the stock price low at the moment. Everyone else, including management, is holding or buying stock back. All my money, with the exception of my house, an investment property, and some bonds, is in this company.

Falcone must want a technical challenge in addition to the financial opportunity.

Everybody comes to the table with a bias. He believes that the Windows space is going to be viable for many years to come. For companies to be completely throwing that away for the Java/CORBA world is, in his judgement and mine, imprudent. I told him he can help us deliver 1.6 and 2.0. But beyond that, we have some ideas about weaving the products together that could be real cool.

I don't know if he'll bring new customers to the company, but I hope he can round out the rest of the technology team. We're a couple shy on the tools head count right now, but we've budgeted for those people, and he's going to be able to bring them in right away.

The question that I would ask is what would make [a Delphi customer] elect to move my way vs. VB. Because that's the only other [competition] that we see. We don't ever lose against PowerBuilder; we don't lose against Developer 2000. We lose sometimes against Delphi, we win sometimes. We lose to VB not because it's better (although it does COM). We lose a) because it's Microsoft—it's a safe choice—and b) the noise in our own system, that we hadn't been doing development for a long time. And I think we'll bring that back.

The total development tools market, including COBOL and all the 4GLs and Java, is 2.5 billion dollars.

That was a favorite remark of Umang Gupta, when people asked him how the company would grow.

I'm not going to criticize the past too much, but one of the comments I make, generally speaking, is this company really had no business needing to hire me. It should have been a \$250- to \$500-million company. With its lead in tools in 1991 through 1993? It's like VisiCalc before 1-2-3, 1-2-3 before Excel; it shouldn't have happened. I'm having a ball—I'm excited. The things that make me the most excited are the customers, when I talk with them, and then the employees, because there's a lot of energy going on. But I'm still having to prove it to other people.

The foreign distributors have had lots of autonomy and different policies in the past. What does the future hold?

This company technically had real strong leadership and vision. But managerially, the company was five or six or seven totally separate businesses. The guy that ran Germany ran Germany, and you didn't touch Germany. It was the same way in the U.K, in the U.S., and in Asia Pacific. All that's gone now. We're trying to run one company, respecting the fact that the markets have different characteristics. Policy, direction, and customer orientation are getting inculcated as we go along.

Tomorrow I'll be showing my customers how they can win. I don't win unless my customers win. I believe this stuff, all the way down to my bones. **CP**

All my money, with the exception of my house, an investment property, and some bonds, is in this company.

Put NetWare to Work

Joachim Meyer

Computer experts some time ago recognized the importance of sharing data and programs from a centralized machine (server) and enabling other computers to access it. Novell invented one such networking file system and called it NetWare. NetWare enables computers connected by network interface cards and cables to see the server's hard disks as normal drives and read/write to them. The NetWare server can also share its printers with the connected clients.

Clients that want to access the server's resources must authenticate ("login") first. During this process the client enters a username and password, and the NetWare server checks the entries against its list of known users. If the user is unknown or the password is incorrect, the server refuses to give the client access to its resources.

SQLWindows
Centura

Beneath the appropriate hardware the client needs to have a NetWare Requestor installed. You can think of the requestor as a "driver" that provides a programming interface and takes care of transporting the data to and from the NetWare server. There are several requestors on the market, each of them for a special operating system or server version. Novell no longer supports the older versions of requestors; the latest version is 2.5 (also known as "Client32") and can be downloaded from Novell's Web site at www.novell.com/download. All the APIs I reference in my articles require the latest Novell NetWare requestor, and the samples aren't guaranteed to work if an outdated requestor is installed.

As soon as the NetWare requestor is installed (and working), you have access to a huge number of NetWare APIs. Some of the API functions take lots of parameters, like handles, memory buffers, and C structures. Because these types of parameters are a bit difficult to handle in native SAL code, I wrote several interface classes that keep track of handles and buffers. C structures are internally converted to and from SAL variables.

This is the first in a series of articles that demonstrate how to do advanced programming with Novell's NetWare and SQLWindows/32. Part 1 explains connections to NetWare servers and gathering NetWare-specific information.

The class hierarchy

Every class in the hierarchy is based on a root class called cNWClass (see "The cNWClass class"). This makes it possible to have a single point of implementation for commonly used functions and globally used variables.

The class cNWConnection (see "The cNWConnection class") is derived from cNWClass and implements frequently needed functions about connections, broadcasts, username/servername, drive mapping, and others.

Initialization of the NetWare API

Almost every NetWare Requestor DLL has a pair of initialization and termination calls. These are *Init() and *Term(). The full names can be found in a comment in each library file (for example "NWCallsInit()" and "NWCallsTerm()" in NWBase.APL). Good points to call these functions are the SAM_AppStartup and SAM_AppExit message handlers, or in the SAM_Create/SAM_Destroy message sections of the main form.

If you include any of the NetWare APLs, be sure to call the appropriate initialization and termination functions. Not doing so may result in memory leaks, misbehavior of APIs, and access violation errors.

The cNWClass class

cNWClass is the base class of every other cNW* class. Currently cNWClass contains only a single variable and a single function to return the variable. By convention the variable m_cc is used on every internal API call to keep the API's return-code. This return code can be checked with the function GetLastError().

The cNWConnection class

The cNWConnection class is derived from cNWClass and inherits the GetLastError() function. Internally, cNWConnection writes every API return code to the inherited variable m_cc. cNWConnection takes care of many functions that deal with connection references and connection handles.

A connection handle is a handle to an established server connection; a connection reference is only a reference to that connection handle. This is useful when you need to open and close connections frequently, because the requestor only keeps track of an internal reference counter, and a real connection is only established once for every server. However, this means you might lose your connection when you free the last connection reference.

A connection number is the number that the server uses to enumerate connections. You can view the list of current connections by loading a NetWare Loadable Module (NLM), which is the extension of files that can be executed on a NetWare server on the server console. Simply type "LOAD MONITOR" on the console and select the first menu item. This NLM ships with all versions of NetWare 3 and 4 and can be loaded by anyone who has access to the server console screen.

Connection-related actions

The cNWConnection class covers several aspects of connection-related programming:

- Get several pieces of information, such as server name or username.
- Create and delete drive mappings.
- Send broadcasts to a server or to clients and poll for broadcast messages.
- Use the Transaction Tracking Service for any files on a server.
- Use semaphores.

For an explanation on how to use the functions, see the sample program, NWTest1.apl. It contains a form called frmMain with some sample functions (Test_*).

How to use cNWConnection

The cNWConnection class has two connection-related internal variables: m_ConnHdl and m_ConnOpen. m_ConnOpen indicates whether a connection has been opened or not. If it is open, then m_ConnHdl contains the handle for that connection.

When you want to use the cNWConnection member functions, you have to open the connection first. This can be done in two ways: via connection reference or via servername. If you call OpenByRef, you have to pass a valid connection reference or use "-1" to advise the function to use a default connection reference. If the function returns TRUE, your connection is ready for use and must be closed by a call to Close() when you don't need it any more (see Listing 1). It's always a good idea to

Free Stuff

I encourage you to subscribe to the DeveloperNet (<http://developer.novell.com/>), which is free on the so-called "Electronic Level." The subscription allows you to download System Development Kits (SDK) and tons of sample code along with Technical Information Documents (TID) and miscellaneous stuff.

close a cNWConnection object as soon as possible, because opening a new connection is fast and an open connection consumes resources. But when you call the cNWConnection class functions inside a loop, open the connection prior to the loop and close it after the loop finishes.

A connection to a server can have three states: attached, authenticated, and licensed. A client that's merely attached to a server isn't granted any rights to access server resources. However, limited access to some things, such as the login directory or the ability to scan for other server addresses, is available. After a client has attached to a server, it can authenticate the connection for a specific user. Authentication is the process of securely identifying the user to the server and granting specific rights to resources on the server. Licensed connections enable the use of mapping, file system, and printing functions.

Listing 1. Opening and closing a connection.

```
◆Function: Test
  ◇Description:
  ◇Returns
  ◇Parameters
  ◇Static Variables
  ◆Local variables
    ◇cNWConnection: C
  ◆Actions
    ◆If C.OpenByRef( -1, NWCC_OPEN_UNLICENSED )
      ◇! your code goes here ...
      ◇Call C.Close()
```

Mapping drives

A workstation is able to access a NetWare drive when it has established a "map" to the drive. Mapping is the process of associating a drive letter to a NetWare volume. Mapping a drive with the cNWConnection class is easy. Just open the connection to a server and call MapDrive(). The function takes two parameters: <DriveLetter> is a case-insensitive single character and <Path> is the path, including the volume name but not the server's name. If the letter was already in use by a map, the map is overwritten. Deleting a map explicitly is done by the function UnmapDrive(), which takes the drive letter as a parameter. Information about mapped drive letters can be obtained from GetMapInfo() which returns the status and path of a drive letter.

Sending broadcasts

A NetWare client is able to send broadcast messages to server consoles and other clients. `BroadcastToConsole()` sends a given message to the console screen of the server the `cNWConnection` object is pointing to. `SendBroadcastMessage()` accepts an additional parameter, which is an array of connection numbers. Remember: Connection numbers are the numbers you see in the connection list of the Monitor NLM. They have nothing to do with connection references or connection handles.

The Transaction Tracking System

You're probably familiar with the mechanism of transactions. If you execute several related modification statements to a SQL database and execute `COMMIT` as the last statement, then all the modifications are executed. If you close the series of statements with a `ROLLBACK`, none of the modifications are executed. If server or client crash, while the series of statements hasn't been executed completely, the database automatically rolls back on the next connect.

On a NetWare server, the process is similar. You open a transaction, make your changes, and at last commit or rollback. If your workstation loses its connection for some reason, the changes made so far are rolled back. The main difference is that you don't need an explicit action to start a SQL transaction, while a NetWare server needs a transaction start point. Also, make sure that your file is flagged as transactional (using `cNWConnection.SetExtAttr()`) and that TTS (transaction tracking system) is enabled at your server (which is enabled by default).

Using semaphores

Semaphores are named counters residing on a NetWare file server. A client can open a semaphore (identified by a string) and initialize the counter to a value greater than zero. While the semaphore is open, the client calls `WaitSemaphore()` to decrement the counter of a semaphore and `SignalSemaphore()` to increment it. When the semaphore counter is 0, a call to `WaitSemaphore()` will fail (after a specified timeout), until the semaphore receives at least one `SignalSemaphore()`. Also, every semaphore keeps track of how many clients have opened it currently.

OK, that's nice, but what can you use it for? Let me give you an example. Once I had to develop a mechanism to tell a client that a certain record in a SQL database was in use by another user (a.k.a. "locking"). I accomplished this task by defining a special format of semaphore names. Every time a client was editing a record, I opened a semaphore and named it something like, "Application.Table.PrimaryKeyValue." After it was open, I checked the open counter. If it was 1, I was the only one

to edit that record, and if it was greater than 1, somebody else had opened this semaphore previously and I could give my client a message.

This is similar to the ISAM record-locking mechanism. If you want to implement something like that, make sure that all clients agree upon which server to use for tracking the semaphores. The semaphore name can't be more than 128 characters, including the trailing zero byte.

Setting extended attributes

Files on a NetWare file system inherit some extra file attributes. The most important one is the transactional flag (value 0xFF), which must be set when the file participates at the transaction tracking system. Only files with this bit set can roll back transactions. Setting and reading the attributes is done through the `cNWConnection` functions `GetExtAttr()` and `SetExtAttr()`. The filenames to be passed to both of them must have the format "{volume}:{path}". You can't use a drive letter.

An example of dealing with file attributes is given in the `Test_Transactions()` function of the test form.

The test application

My test application contains two forms: `dlgPrompt` and `frmMain`. `dlgPrompt` is only a helper to prompt the user for input whenever needed. `frmMain` is the actual application. It shows a combobox containing all test functions currently available. Each entry in the combobox is associated with a test function. The output of the test functions is directed to the memo field. The names of the test functions correspond with the combobox entries and are preceded by "Test_":

- `Test_WhoAmI` scans all currently active NetWare connections and displays information about each.
- `Test_ShowDriveMappings` loops over all possible drive letters and displays mapping information.
- `Test_BroadcastToConsole` prompts the user for a message and sends this to the console screen of every NetWare server to which the station is connected.
- `Test_BroadcastToConnection` prompts the user for the name of a server, a connection number on this server, and a message text. Then it sends the message to the connection of that server.
- `Test_CurrentListOfSemaphores` prompts the user for the name of a server and displays all currently opened semaphores on that sever.
- `Test_SetASemaphore` prompts the user for the name of a server and a semaphore identifier, sets the

semaphore, displays all currently opened semaphores on that server, and closes the semaphore.

- *Test_Transactions* prompts the user for the name of a server and a valid drive letter. Then it creates a test file and writes to it inside a transaction. After closing the file, it prompts the user whether to rollback or commit the changes. The function also makes sure that the transactional bit of the test file has been set.

What's next?

There are lots of APIs to discover yet. A big part of NetWare programming is the exciting NetWare Directory Services (NDS). It provides a system-wide

implementation of an operating-system database with the ability to store objects and their associated data. My next article deals with NDS and shows how to get and modify NDS information. **CP**

Download NetWare#1.zip from this issue's Table of Contents at www.ProPublishing.com or find it on this month's Companion Disk.

Joachim Meyer is one of the owners of BASYS EDV-Systeme, a networking company in Bremen (Germany). Being a Novell and Microsoft systems specialist he is responsible for software development. He has almost 10 years of experience in programming with SQLWindows, CTD, Delphi, and C. You can contact him at JoeM@basys-bremen.de.

...And How Does That ...

Continued from page 2

of Centura's commitment to the tools. Recently, as in "since lunchtime." Centura's new management team consists of turnaround specialists, and turnaround situations are, by definition, fraught with uncertainty and difficulties. But, as Broomfield noted in the closing session, "It's OK to be skeptical. It's not OK to be cynical."

Has a line been crossed? For many years, Centura customers could portray themselves as the victims of incompetent management. They had bet their future on what was the finest 4GL in the marketplace, only to watch it become a minor player in the hands of people who could not market it effectively. In 1997 the previous management team blew a bundle on marketing a product that wasn't theirs (ForeSite), screwed up a critical merger, and cancelled the customer conference. Let's not even discuss the stock price! So there's ample reason for bad feelings. And when you're a victim, you don't have to examine the moral validity of your position; the victim is always right.

Now Centura seems to be in the hands of good management. Certainly they've achieved some major success in the refinancing of the company's debt. The third quarter results show that they can also sell product and gain customers; indeed, the tools business has become modestly profitable. The new style of announcing and delivering smaller improvements, faster, looks like it's working so far, although it's quite a contrast to the announced-but-never-delivered grand projects of prior management teams. Ironically, management's failure to announce their five-year plan, sticking instead to more achievable goals, seems to be interpreted by customers as a lack of commitment. That's how we got a reality-based guy like Broomfield standing in front of the closing conference session saying, "I plan to release a 1.6 version. And a 2.0 version. And a 2.5, and a 3.0, and a 3.5, and a 4.0 . . ." We customers have been dealing with old Centura management so long, we aren't satisfied with reality. Tell us a story!

Therapy time

It's obvious how we came by our cynicism, which has caused us to practically accuse current management of being liars. After all, previous management were liars. But as Broomfield and company continue to put together one modest success after another, our cynicism grows more questionable. One must ask, what is the customer's role with the company today? Is it to sit back and wait for today's signs of positive change to double and redouble? What is the management team's role? To offer personal apologies for the betrayal of hopes that happened before they took over?

Are we still victims? Not in December 1998. As with any other unhappy experience in life, we must find our own ways to handle Centura's past mistakes, get over it, and work with the current team to make things better. They have plenty of talent, and so do we. They're doing a lot of things right, and they're willing to listen to input on the things that could be done better. It's our job to help them, for their good and for our own. So talk to them—don't whine to them—and act like your input will make a difference, because it will.

If you missed the conference, you missed a great experience. The evaluation form asked what the highlight of the conference was. Although many of the sessions were very engaging, my highlight was the same as always: personal conversations. I learned so much! I got an opportunity to bounce my viewpoints off my peers, and see what they looked like when they bounced back. I had ideas and opinions that needed to be heard, and Centura's people listened. I had preconceptions that needed to be shattered, and they were. That's what happens at conferences like these. Come to the next one, and let it happen to you.

Scott Broomfield had a lot more to say than just the quotes I mentioned here. Read the interview in this issue for more background on why the tools are strategic. And expect more in-depth conversations with other Centura managers in the months to come. **CP**

Dear Ace,

Is there an easy way to include the ability to dial into a Unix server and connect to a database through a Centura application. (version 1.5 PTF 1) running on Windows 95 and Windows NT 4? I hope you can help!

Ian Ballard

Ian,

It's not clear what you're trying to do. Several products support dial-up connectivity to remote servers. Windows 9x and Windows NT come with dial-up connectivity to remote Windows NT Server (RAS), in fact. Presumably, you want to connect a Centura application running a remote, occasionally connected machine to a database on a server that's part of a corporate network. If the dial-up connectivity software has been configured to support the protocol you need to use to connect with the database (TCP/IP, for example), this should be easily done.

Hope this leads you in the right direction.

Ace

Dear Ace,

I use Quest 3.0.3 ptf4 for reports using SQL Server. What limitations are there when retrieving records? I get the error message, "Asynchronous timeout. Pending results have been cancelled," when I execute a large SQL statement in Quest. There are about 250,000 records that I need to process. Thanks.

Susan Tisinger

Susan,

It's been some time since Ace used Quest. I recall the error you describe as a behavior that can be observed when connecting Quest to a SQL Server 6.x database when using the SQLRouter for SQL Server 4.x (dblib). For SQL Server 6.x and higher, Quest should connect to the database using ODBC connectivity, since Microsoft has stated that ODBC is the "native" connectivity interface for SQL Server 6.x. Hope this helps.

Ace

Dear Ace,

We have a problem with our tour planning system in a SQLBase 7/Netware 4.11 environment. The database connection runs on ODBC. We use the ODBC driver shipped with the SQLBase 7 package, which comes from Intersolv (OEM). Here in our office the application runs properly. At our client, we frequently have "Out of memory" and "Cannot find session" errors. Suggestions?

Sincerely,

Marco Stecher

Marco,

These kinds of problems are usually due to configuration and/or environmental settings, but they can be difficult to track down. Look for differences between your office's setup and your client's setup. Make sure you have the latest PTFs applied as well. Also, you might want to consider a support agreement with Centura Technical Support for a production environment. CTS will be able to assist you in resolving the issues particular to your configuration.

Ace

Dear Ace,

Our 16-bit SQLWindows application uses the SAL function SalHStringToNumber to convert a string handle so it can be passed as the IParam. This works fine in the 16-bit world but causes a crash in the 32-bit world. The help for SalHStringToNumber in Centura Team Builder says to "not use this function for strings passed as parameters in messages posted using SalPostMsg". Is there a workaround to this problem other than using SalSendMsg?

Sincerely,

Tom Schenk

Tom,

The problem with using SalHStringToNumber and SalNumberToHString is that there was never any documented guarantee on how long the HSTRING was valid. Even in 16-bit SQLWindows posting a message with an HSTRING in IParam can be risky. Centura only guarantees the HSTRING is valid during a SalSendMsg and subsequent message processing.

You don't mention what version of CTD you're using. A bug with regards to the life of HSTRINGs was addressed around CTD 1.1.1. If you're using an earlier version, your problem may be solved by upgrading.

If you need another alternative, here are some more ideas: Create a String stack and push your string onto the stack. Then pop the stack in the posted message's handler. Or use the CStruct functions to allocate a small amount of memory buffer for your string, copy the string to the memory buffer, and pass the address of the memory buffer when you call SalPostMsg.

Hope this helps,

Ace

SQLWindows

Centura