

SPREZZATURA

45 St Mary's Road • Ealing • LONDON W5 5RQ

Tel: 020 8912 1010 Fax: 020 8912 1011

info@sprezzatura.com

SEN

August 1998

Sprezzatura's Electronic Newsletter

For Revelation developers by Revelation developers

MAKING DATABASES HAPPEN

SPREZZATURA

Contents

1. Leap Year	10
2. Index formats	10
3. Accounting months	10
4. Hard Coded 19	10
5. Calculations using a two digit Year Value	11
6. Year Input Validation	11
1. DATE()	11
2. External Interfaces	12
3. Customers and Suppliers.....	12
Step 1 - Creating a Connection Object (XO).....	12
Step 2 - Creating a Query Object (Qry).....	12
Step 3 – Executing the Query	12
Step 4 – Retrieving the Results.....	12
Step 5 – Destroying the Query Instance	12
Step 6 – Destroying the Connection Instance	12
COPYRIGHT NOTICE.....	12
TRADEMARK NOTICE.....	12

SPREZZATURA

Welcome

WELCOME - to this issue of S/ENL, the Sprezzatura Electronic Newsletter.

As promised in the last issue, a little more like on schedule but a little lighter in content!

Big news of this issue is the RevSoft UK "5th Annual Conference" which has been announced for Wednesday, 14th October 1998". To quote from the mailer

"Jim Acquaviva, President and CEO of Revelation Software, will start proceedings with a talk on our strategy for the next couple of years. Cameron Purdy, Principal Software Engineer, will then bring you up to date on our new product, jRev. After coffee, Andrew McAuley, well known for his Revelation expertise, will enthuse about recent OpenInsight developments. Finally, again as last year, we shall have a guest speaker from within our industry but not from within the Revelation community to put the outsider's view".

So if you're planning a holiday to the UK this autumn (whoops, sorry "Fall") then why not arrange to be at the conference – one thing we can guarantee is it'll be good fun – and we promise not to give away insulated coffee mugs this year!

Hectic doesn't begin to describe the last few weeks at Sprezzatura both Inc. and Ltd. It seems that suddenly people are deciding that the time is ripe to convert from AREV to OI. Naturally we agree – OI is now a stunningly mature product, and with the hard work that the lab have been putting in it is now even easier to convert over. We've assisted several companies in this effort recently and the major comment we get is "if I'd have known it was this easy I'd have converted years ago instead of trying to mess around with VB and Access". Why not let us help you to do the same – with our consultancy skills we won't make a drama out of a crisis.

On a much less serious note, I'm absolutely over the moon that one of my favourite artistes has released his first CD in 17 years! Dean Friedman who was the first artiste I ever bought as a teenager, has released a 2 CD set called "Songs for Grownups". I know that my musical tastes are not everybody's so I won't go on too much here, so instead I've had our web mistress knock up a special Dean Friedman page at http://www.sprezzatura.com/dean_friedman.htm. It contains snatches of songs, ordering information and a sad ramble from me. Check it out see if we can agree on something! <g> (BTW all postings are done with Dean Friedman's permission – copyright is taken seriously here at Sprezz! Hell, we're the only people we know with a LICENSED copy of PaintShop Pro and WinZip! OK – after editorial review we now know another!).

A big "congratulations" to Dave Goddard who has been appointed RevSoft's official antipodean representative. We've dealt with Dave in the past and always found him charming and technically competent. Dave Goddard will undoubtedly add much needed impetus and enthusiasm to the Australian RevSoft product market.

It's also nice to see that Dave Goddard's site reflects this appointment. It's always a little disappointing when RevSoft make an exciting announcement about appointing somebody or some organisation to a position, then not to find a mention of this appointment on the corresponding Website. A case in point being one recent announcement where when you went to the Web Site in question explained

SPREZZATURA

that the company were "specialists in converting Advanced Revelation applications to OpenInsight, Visual basic, SQL Server or PowerBuilder".

A big thank you is due to Revelation Software who very generously gave us permission to distribute all of their knowledge base articles with our version of the knowledge base software. This means that if you download our knowledge base software from www.sprezzatura.com/sysknow.zip you'll also get RevSoft's Tech Bulletins, MFS handbook and System Subroutines information. If you've got REVMEDIA FKB this can also be used with the new knowledge base software. We're very pleased with this software as it shows some of the user interface areas in which we as a company excel. Our primary achievement with this window was to enable a slider control so that using the mouse you can drag a divider line and resize the edit boxes on the window – so you call up the window by shift clicking on SYSKNOWLEDGE...



And then simply put the cursor over the vertical bar so that it becomes an arrow, click and drag to get the size you want



In addition the software shows tooltips on a list box and the imaginative use of bitmaps in a hierarchical listbox. As another subscriber mailed us "As a member of the Revelation community, and one whose living pretty much depends 100% on AREV and OI, thank you for your continued efforts to push the envelope with these products. Though we are not (yet) 32-bit, though we don't have the fanciest development toolset, though nobody else in the computer world is manufacturing add-on components as they are for VB and Delphi, we are still holding our own because of people like Sprezzatura. To quote a Brit I know, "Stout work fellas!""

Right, time is tight so time to wrap up. Hopefully by going with HTML only we'll finally silence all of the negative feedback about PDFs we've been getting! (See feedback). By the time we next go to print Carl should be in the 'States on an intercompany transfer to Sprezz Inc. – business is so good in the US that we're having to prove the concept of the global village! Still we always welcome new business opportunities so if you've got any requirements, training, consultancy or even just programming in Rev G, AREV, or OI just drop us an email at sales@sprezzatura.com or call our UK or US office! We'll look forward to hearing from you.

Finally a big shout out to my homey Eric who's now the proud parent of a bouncing baby boy. Congratulations to the loving father and mother, and another large one to the Lab at RevSoft who are so unstintingly helpful with the technical issues we try to address herein. Thanks guys..

Andrew P McAuley

Publisher

SPREZZATURA

Feedback

With all the brouhaha about Y2K issues at the moment I was interested to receive the following mail –

"To the Management, Sprezzatura

Recently I discovered your web site, <http://www.sprezzatura.com>, and was initially encouraged to see that you had given space to the Year 2000 issue. However, I was disappointed to find the following statement:

"Sprezzatura are pleased to confirm that their systems are all Year 2000 (Y2K) compliant. This is as a direct result of them being developed in Revelation Software's products which are themselves inherently Y2K proof."

The message you appear to give is that because your product is developed in Advanced Revelation then it is inherently Y2K compliant. Unfortunately this is not the case and you may have given other Revelation developers and users cause to take the matter no further. This could have very serious consequences for their business.

Revelation is no different from many other software development languages (DBASE, Paradox, Access, Delphi, Visual Basic etc.) - all of them susceptible to this problem. Although the date handling in Revelation is tidier than in many languages, this does not prevent a developer doing whatever they like in the code.

There are in fact several ways a Revelation application can be non-compliant, as I discovered when recently performing a Y2K assessment on several banking applications. If your company has done no more than assume you are compliant because the Revelation language is compliant and have not checked your applications, you and your customers may also experience problems.

I have developed Revelation Y2K scan software and am currently concentrating on Rev/PICK Y2K work. I can assure you that problems do exist and, if you wish, I can give you the Year 2000 project manager's email address for the bank I recently completed an assessment for. You can then confirm this fact for yourself.

I've listed some of the possible problems below. I realise most of these examples are the result of bad coding technique or lack of forethought, but then so are a large percentage of the global Y2K problems. It is not just COBOL mainframe systems restricted by space that need to be checked. If that were so, all Microsoft products would be Y2K compliant.

I am a subscriber to Peter de Jager's Year 2000 Announcement List. As subscribers, we assist each other with Year 2000 assessment and remediation issues. In addition, we do what we can to raise awareness of the seriousness of this problem within government, business and community. It is instances of unawareness such as that expressed by your web site that is the reason for much of our concern and which makes our task that much harder in getting the Y2K message across. I recommend that you modify your Year 2000 statement to reflect the true nature of the problem which is that that each application will require assessment to ensure that it is problem free, in particular financial and time related applications. Also, you need to state that you have checked your applications for compliance.

SPREZZATURA

Regards

Jocelyn Amon

Computer Consultant"

Well, I'm always a fan of "in your face" advice so I mailed Jocelyn with the suggestion that an article on the subject would be more than welcome. And hurrah – check out later in this issue for an article on Y2K as it impacts AREV. Thanks Jocelyn!

Walter Giesler from Germany writes

"Thank you for the fast shipment of our order for the OpenInsight Works update. We already received it yesterday morning.

One of the first things I do is scanning a CD for textual information. The Intlchar.pdf in "White papers" caught my attention and I printed it out immediately.

I cannot believe what I had to read there: OpenInsight cannot handle text with (amongst other very rarely used characters) the German u-umlaut because Microsoft elected this to be at a character position used by OI for @SVM. And they have an easy workaround: Fold u-umlaut to u and everything is all right !?!

I know that Germany is far away from the United States and that they tend to regard everything outside their borders as jungle, but at least now I know why nobody in Germany uses OpenInsight - one cannot use it.

u-umlaut isn't one of these esoteric typographic glyphs but a common character found in such everyday words that mean door, for, cap etc. Take "Sud" for example which means "decoct, brew" without umlaut and "south" with umlaut. I cannot even do without the u-umlaut in the descriptions of our microcomputer products in an invoice.

The times where the phone company used "ue" instead of u-umlaut now have fortunately passed since 10 or 20 years and n o b o d y here would accept a program that cannot handle the full German character set.

Isn't there a way to convert u-umlaut into something harmless and re-convert it on output? AREV always has been very flexible with things like this.

I spent many weekends since Christmas on the decision where to go from our AREV1.12 and with the new enhancements of OI in the last revisions I now had decided to move to OI - but now I'm again at the beginning!

Best regards

Walter L. Giesler"

Fortunately the last SENL had exactly the answer Walter needed as he subsequently mailed

SPREZZATURA

"Thank you for this recommendation!

I didn't believe it to be as easy. And it wouldn't have been necessary to get as angry as I was yesterday evening....Anyway - I can start now, Thank You very much for the help!"

The only problem is, that as Oystein Reigem pointed out on a recent visit – we reversed the order – the last should be first and the first should be last (may the Lord in his mercy be kind to Belfast (sorry, traditional rhyme breaking through...)).

We've had more feedback about the use of PDFs, all of it negative, typified by this comment from Stephen Bush in Cyprus (this is an island in the Mediterrean by the way, not an urban district in the US of A...

Andrew,

Many thanks for SENL.

However the PDF format stinks (for me) it is very difficult to read as the Adobe reader is clunky.

Emu's code this time shall have to remain untested as I cannot cut and paste from PDF and I am not going to type all those little codes by hand.

I suggest that you use web site HTML - like the 99.99% of the world ;-) HTML does graphics and is easily produced.

Best Regards,"

So enough already, as of this issue we're sticking with HTML...

Meanwhile back at the sales request corner

"Do you know any AREV developers in Costa Rica or any of the central or South American countries? We have our anatomic pathology package converted to Spanish and need a salesperson/developer to market for us."

Anyone interested let us know and we'll pass on your details to our subscriber who asked this.

As regular readers of SENL will know, we never pass a chance to print letters of praise from subscribers so thanks to Stacey Hawes who mailed us "You and Aaron definitely make life easier for those of us with little or no knowledge of AREV."

Putting the shoe on the over foot for a moment we'd just like to thank Australian's CSSP's very own Michael Moran who continues in his quest to ensure that S/List is bug free. Michael not only locates the problems for us but he also then fixes them thus helping all of our user base. THANK YOU Michael – keep up the good work!

Time for a big "D'oh!"...

SPREZZATURA

"Don't mean to be a pain, but the idea of Gary's column is that it's like a quiz, and the answer that should appear at the end of the newsletter should appear at the end of the newsletter, not right after the article.

C'est la vie

Gary"

Finally my favourite piece of feedback came from Dave Harmacek who wrote "Thank you for placing SENL in A4 format. It is nice to see that you British get a few silly millimeters more per page." And yes Dave, it IS spelt favOUrite! <g>

MYKL'S JOKE SECTION

A week in hell

One day, a guy dies and finds himself in hell. As he is wallowing in despair, he has his first meeting with a demon:

Demon: Why so glum, chum?

guy: What do you think? I'm in hell.

Demon: Hell's not so bad. We actually have a lot of fun down here. You a drinkin' man?

Guy: Sure, I love to drink.

Demon: Well, you're gonna love Mondays then. On Mondays, that's all we do is drink. Whiskey, tequila, guinness, wine coolers, diet Tab. . . we drink till we throw up and then we drink some more.

Guy: Gee, that sounds great.

Demon: You a smoker?

Guy: You better believe it.

Demon: All right! You're gonna love Tuesdays. We get the finest cigars from around the world and smoke our friggin' lungs out.

If you get cancer, it's okay . . . you're already dead.

Guy: Golly!

Demon: I bet you like to gamble.

Guy: Yes, as a matter of fact I do.

SPREZZATURA

Demon: Good, because Wednesday is gambling day. Craps, blackjack, > horse > races, you name it. We even opened up a pai gow poker table.

Guy: Gosh, I never played pai gow before...

Demon: Well now you can. You like to do drugs?

Guy: Yes, I love to do drugs. You don't mean. . .

Demon: That's right! Thursday is drug day. Help yourself to a great big bowl of crack. Smoke a doobie the size of a submarine.

You can do all the drugs you want, and if you overdose, it's okay . . . you're already dead.

Guy: Neat! I never realized that hell was such a swingin' place!

Demon: You gay?

Guy: Uh, no.

Demon: Oooh, you're gonna hate Fridays.

Revelation and the Year 2000 - Jocelyn Amon

There is a common belief that because the PICK language has no problems with handling the Year 2000, then any applications written in languages based on PICK are also free of these problems. My recent work with Year 2000 audits of these languages has shown that this is not the case. Out of curiosity I also checked some financial programs that I had written in the past but which are now, fortunately, obsolete. I had always considered that I was an above average programmer and took pride in the fact that my software had relatively few bugs, so I was shocked to find these types of exceptions in my code. The main exception was a hard-coded 19 in the year for report headings, something that is inexcusable given that a four-digit year is easily obtained by other means. I was relieved to find I was not alone in doing this as other programmers had done the same and a lot worse besides.

It doesn't matter what language it is, the problem is that mostly we programmers have not given any thought to the Year 2000. Many of us are still not aware of our bad coding habits and won't be until it is brought to our attention. I guess what it proves is that contrary to popular opinion, we programmers are just human after all.

Any Year 2000 audit should not be done as a blame laying exercise - this does not achieve anything and can be the cause of additional problems such as cover-ups. The type of problems found in applications need to be documented and noted to prevent them occurring again but nothing is gained by publicising the authors. Code checks, peer review and regular quality meetings are the best means of preventing bugs in code but unfortunately the programming profession very rarely incorporates these practices into application development and the programming psyche doesn't

SPREZZATURA

readily accept such measures. More than likely programmers will be forced to be more disciplined in the next century.

Some of the Year 2000 issues that may affect your Revelation applications are described below.

1. Leap Year

A leap year is determined, in most cases, by checking if the year is evenly divisible by four and if so then the year is assumed to be a leap year. This method will correctly determine that 2000 is a leap year. However, the years 1900 and 2100 are not leap years and this check does not work. Therefore some calculations have had additional accuracy added by determining whether the year is the first year of a century i.e. divisible by 100 and if so, determines the year not to be a leap year, including 2000. The first year in a century is a leap year where it is evenly divisible by 400 - some programs fail to make this extra check. You are particularly vulnerable to this problem if your application holds and calculates with dates pre-1900.

The correct code to determine a leap year is as follows:

```
LEAP_YEAR = "N"

IF MOD(YEAR,4) = 0 THEN

IF MOD(YEAR,100) NE 0 OR MOD(YEAR,400) = 0 THEN LEAP_YEAR = "Y"

END
```

2. Index formats

To save space, indexes which are formed as symbolic fields may use the YY format. This can cause problems with sorting, selecting and any reconstructing of the year back to the YYYY format.

3. Accounting months

It is not uncommon for systems to use the YYMM format to name, index or reference accounting months - calculations, sorting and selection can be affected.

4. Hard Coded 19

Hardcoded '19' is usually found on output reports but can be found in code where more serious problems may occur.

The following is a real example:

```
YEAR = OCONV(DATE(),"D2/E")[7,2]
```

```
MONTH = OCONV(DATE(),"D2/E")[4,2]
```


SPREZZATURA

IF MONTH > 10 THEN YEAR = YEAR + 1

YEAR_ID = "19":YEAR

This code then goes on to YEAR_ID as a key to update a yearly statistics file.

5. Calculations using a two digit Year Value

An instance was found where '1' was added to YY which was subsequently used to build a DDMMYY value. This value was ICONV'ed to get the same day next year. This works with 97, 98 but for 99 and up it will fail. For 99, 100 is returned and for 00, 1 is returned. Neither of these values is suitable for inserting into the YY position without further massaging.

Calculations using a Year value occur when determining: Age, Anniversary, Maintenance/Check-up Due, Investment/Loan/Mortgage Maturity, Payment Due, Last/Next Month, Last/Next Year, Forecasting/Budgets etc.

Note: This type of exception has been found to be very common in financial and insurance applications where anniversary dates and maturity dates are calculated

6. Year Input Validation

Validation of Year may occur either in the entry screen record or within program source code. The acceptable range for input of year may cause an exception i.e. a entry screen range check '(1980,2000)' will pass any Year 2000 testing but fail in 2001. Any value for the top of the year range is possible as a programmer will often arbitrarily pick a value.

Also, where a 2 digit year input is requested, not only should the validation range be checked but also ensure that a hard-coded '19' is not pre-fixed by the processing.

7. DOS filenames

Traditionally DOS filenames have had a limitation of 11 characters - 8 for the prefix, 3 for the suffix. Often the date is required to be incorporated into the DOS filename of data files imported and exported. Sometimes the YY form of the year is used when creating this date and this may be a problem when sorting, selecting and/or archiving these files. Other 'odd' problems could also arise when processing such files.

EXTERNAL ISSUES

While not directly related to your applications, the following issues require consideration.

1. DATE()

The DATE() function accesses the hardware date (BIOS, CMOS etc.). If the hardware is not Year 2000 compliant, data could be scrambled and/or lost on any of the critical Y2K dates. Year 2000 compliance of hardware is a complex issue and needs to be taken into account if you haven't already done so.

SPREZZATURA

2. External Interfaces

Where an application receives in a YY date format or exports YY date formats to external systems, it is recommended to check that the points of interface within those systems have also been detected and checked. A checklist of all points of interface containing YY dates will prove useful to those working on the external systems' Y2K issues.

3. Customers and Suppliers

The viability of your computer system relies on your critical customers and suppliers also being Y2K compliant. If they fail to make it into the next century, your company may no longer be viable. It is therefore in your interest to give every assistance possible in ensuring that suppliers and customers are Y2K compliant, particularly where they interact with or use your company's computer systems.

Jocelyn Amon

Consultant

Financial Solutions Limited

(Marketers of Revelation Year 2000 Scan Software)

finsol@ts.co.nz

Ruminations - Gary Gnu

I was off at my favorite client's the other week. OK, they're not really my favorite client, but it is one of my favorite sites to work. Abe and Izzy make the best prochas this side of Lake Victoria. Abe Achinda and Izzy Ibex run a deli and over the years have moved their inventory control system up from RevD through to OpenInsight. The other week, they wanted me to set up the front-end registers and table services systems to run with the inventory system.

Even though their constant bickering drives me bananas, I love the atmosphere of the place, so I showed up early for a huge helping of lox, sable and Mollie's special herring salad. Mollie's Izzy's wife and also the baker. Just for reference, Abe's wife, Berta, makes the prochas. As I was done with my fressen, Abe strolled over to give me the skinny on what they wanted.

"Gary, boichek!", Abe shouted. Abe's got a huge booming voice, and is constantly laughing. "You ready to work, or you want maybe to sit and have a cup of coffee and I'll tell you what we want instead of you giving us what you think we need."

"Ahh, don't listen him Gary. If Abe has his way, we'd have a drive-thru in the back. He doesn't know what we want. If he did, don't you think he do it himself?"

That was Izzy walking up. He sat down next to me, and started picking at the remnants of the lox on my plate. Not that it was a big deal, they never charge for the grup anyway.

SPREZZATURA

"Listen, cause I only want to say this once," Izzy told me. "I want a simple system. When the girls run the register, I want it to look the same as when they enter stuff in to the back. The scales have been printing barcodes for years, and they can run them right through the front with the rest of the stuff, capiche?". And with that, he walked into the back and started talking at the guys behind the deli counter.

"So, Barry calls his mother in Florida. 'How are you, ma?' he says.

"'Not good,' she says, 'I haven't eaten in 5 days.'"

"'5 days, ma! What's wrong?'"

"'Gd forbid you should call and my mouth be full'"

I groaned. "Please, Abe, that was horrible."

"OK, I'll try and be serious, just for a minute. Izzy's pretty much on the ball. The only thing is, I don't want we should see all the prices and fancy stuff for the girls in the back. Just want them to ring in the order. Izzy's a bit worried you're going to have to write two systems, but I told him not to worry, cause Gary always does right by us, don't you Gary?"

"You know I do Abe," and with that I headed to the back room and started looking over the system.

What they wanted really wasn't all that complex. Hook into the inventory system, display the prices, track what's taxable, and display the subtotals. I'm just glad I didn't have to interface in with the scales. All the fruits and veggies are sold by the piece.

My initial design consisted of an editable. It didn't sit well with me, and Izzy hated it. He was looking for something that looked more like a register tape.

After thinking about it for a while, and over a corned beef on rye, some kasha and a sour tomato, I ended up dropping on a couple of edit boxes. As each item is scanned in, it was just a matter of linking up the second edit box for the price list. Izzy had the great idea of keeping a running total, underneath the prices, in red, of course. This just means I had to replace the control with a rich text control. No biggie, since this wasn't an editable field anyway.

I still had to contend with Abe's desire to have the same window handle both jobs. Pretty easy stuff, making things appear and disappear, but there were enough differences on each screen to make the job a little difficult. First, on the counter side, there was the table number, server number, number of people in the party and the check number to display. At the register, there was tax and subtotal boxes and the slots to enter in the coupons and display the bitmaps showing the current specials (and, of course, pictures of the grandchildren). Lots of things to check.

I was never much for manipulating all those controls and I wanted to find a simpler way to handle all these controls. As I munched on a piece of halavah, Bertha came in, with a plate of her special cookies.

"Have you seen the latest pictures of Jordan?", she asked and without even waiting for an answer she reached into her apron and pulled out two pictures. An adorable little boy, red hair, green eyes.

SPREZZATURA

"Who's son is he?" I asked? There were just so many of them I could never keep it straight.

As Bertha went on about the family tree, the names started blending into one. All I heard was parent, child, parent, child, then it hit me. I knew how I can manipulate the screens and play with a really cool feature of OI.

So, do you know what the gnu knew? The answer is at the end of the issue.

Connections - The Query Interface (or SQL programming for ReAL men!) - C Pates

No more Wombats!Ô

That's right! This time we're going to look at manipulating an SQL datasource directly via the Query Interface, allowing you to build and execute your SQL statements at runtime, without any of the work associated with pre-defining dataset scripts and columns (Look Ma, no hands!). As wonderful as the Dataset Object is, there are times when all you want to do is run one simple measly SQL command without any hassle, so lets take a look at the steps involved:

- ☑ Create an Connection Object (XO) Instance
- ☑ Create a Query Object (Qry) Instance
- ☑ Execute the SQL statement via the Qry Instance
- ☑ Retrieve the results (if any) from the Qry Instance
- ☑ Destroy the Qry Instance
- ☑ Destroy the XO Instance

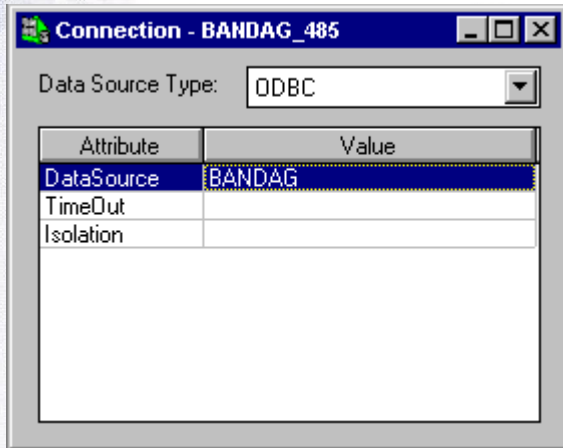
Note that all of the XO and Qry function declarations and constants we'll be using for this article are contained in the Insert record XO_EQUATES, supplied with OI, so make sure you include this in any XO/Qry programming you do.

Government Warning

Although using the Query Interface provides a nice quick and direct way to get to your data, make sure you've defined your datasource (ODBC connection etc.) properly in the C/S Workspace before you rush into the System Editor, or you won't be going anywhere. See page 15 of SENL v2i1 for more information on doing this (and shame on you if you've forgot!).

SPREZZATURA

Step 1 - Creating a Connection Object (XO)



As with all C/S access in OI, an XO (or more precisely an XO instance) must be created first so that OI can communicate with the datasource. These instances are always created based on a pre-defined datasource definition that is stored in the repository like so:

Each of these datasource definitions contains the basic information necessary for OI to connect to several different DBMS's (currently ODBC, SQL Server, and Oracle). You simply specify the type when you create the connection instance- Very handy when working with similar data on different back-ends!

In previous issues of SENiLe we have connected to our datasource via the C/S Workspace Query window and via a form and dataset. In both of these cases the connection object was created by the system, but if you are interested in manipulating SQL data directly you will probably have to create your own connection at some point .

A connection is created via the XOInstance function, which is defined as follows:

```
hXO = XOInstance (SourceName, SourceType, LoginID, Password, Timeout, Options, Scope)
```

The arguments are:

SourceName This is the name of the XO definition (as saved in the Connection Designer above) that you want to create an instance of.

SourceType- This is the type of datasource you wish to connect to. Each XO definition allows you to define separate parameters for each datasource type so you specify which one you want to use here. Possible values are:

ODBC

SQLServer

Oracle

SPREZZATURA

LoginID- The login name to use when attempting to connect to the datasource. This may be left null, in which case the system may attempt to connect using a default login, or ask the user depending on what options are set in the Options parameter below.

Password- Corresponding password for the LoginID.

TimeOut The number of seconds to wait for a connection before returning. This field can be left null in which case a default value is applied.

Options- This is a bit-masked number that specifies several options that apply to the XO instance. See below for more details on these options.

Scope- If the XO instance is being shared by using a scope identifier (XO_SCOPEDSHARE\$ is specified in Options) then this parameter contains the string that identifies which scope to use.

Brief interlude - Connection Options

When OpenInsight attempts to connect to the datasource several options can be specified that affect how the connection behaves. These options are specified in the Options parameter as a bit-masked integer in much the same way the Windows stores window style information.

The options cover four different areas:

Transaction Processing.

This option specifies if Transaction Processing should be used by the XO. The default is to use Transactions. Specify XO_NOTRANS\$ if you want Transaction Processing turned off.

Connection Sharing

When opening a connection it is possible to ask the system to attempt to use a previously opened one instead of creating a new one. However, doing this may negatively impact performance if Transaction Processing is being used, so sharing connections is best done when you want to perform mainly read-only operations.

Sharing has three possible options

You can set the connection to allow no sharing (the default)

You can attempt to share by using a scope, which basically means that you look for another connection that matches a specify string you supply (in the Scope parameter) and use that connection before attempting to create a new one. Specify XO_SCOPEDSHARE\$ for this option.

Finally you can attempt a Global share which basically means that you use any other global connection you can before creating a new one. Specify XO_GLOBALSHARE\$ to use this

Login Options

SPREZZATURA

When logging in to the datasource OpenInsight can use a login dialog box (XO_LOGIN) to get parameters from the user. There are three choices that control the behaviour of this dialog:

You can set the dialog to show only if the connection fails (the default)

You can make sure the dialog never shows and XOInstance just returns an error if the connection fails by specifying XO_NODIALOG\$

You can make sure the dialog is always used by specifying XO_ONLYDIALOG\$

Resolve Parameter options

If you leave some parameters blank or unspecified OpenInsight can attempt to resolve them when connecting. Use XO_NORESOLVE\$ if you want to avoid this behaviour.

OK, back to the plot...

So, to connection to the Access database shown in Figure 3 above we could use:

```
hXO = XOInstance( "BANDAG", "ODBC", "admin", "", "", "", "" )
```

but if I wanted to be awkward and always use a login dialog and attempt to share the connection with a scope of "AWKWARD_WAZZOCK" then I could use:

```
Options = XO_ONLYDIALOG$ + XO_SCOPEDSHARE$
```

```
hXO = XOInstance( "BANDAG", "ODBC", "admin", "", "", Options, "AWKWARD_WAZZOCK" )
```

If successful we should now get back an XO handle which we can then use to open a Query. If we get back 0 then we've got an error so we'll need to find out exactly what has happened (See Retrieving Errors below).

Note that the convention of returning FALSE (or 0) for an error applies to all of OpenInsight's C/S Workspace functions).

Step 2 - Creating a Query Object (Qry)

Well, now we've connected to our database we need to create a Query object which we can use to execute and retrieve the results of an SQL statement. This is achieved via the QryInstance function which is defined as:

```
hQry = QryInstance( hXO )
```

It takes one parameter, which is the handle to the Connection Object that you wish to use for the Query, and returns a Query handle for use in subsequent Query functions. If the function returns 0 then an error has occurred and we'll have to use the XO error functions to find out what has happened (See Retrieving Errors below).

SPREZZATURA

Step 3 – Executing the Query

At this stage we should have a functioning connection and a valid Query object, so next we need to pass the SQL statement to the Query object and execute it. This is accomplished by the QryMethod function, defined as:

```
flag = QryMethod (hQry, method, arg1, arg2, arg3, arg4, arg5)
```

This function takes as parameters the handle to the Query that we created in Step 2, and a value specifying the method we wish to use. There are several different methods that may be invoked by this function, and the values of the argument parameters (arg1...arg5) may change with each method. For now we are interested in the QRY_EXECUTE\$ method which needs three arguments:-

arg1 SQL Script

arg2 Execution Type - at the time of going to press this can take two values. Setting to True\$ (the default if nothing or null is passed) will return results, False\$ will discard results.

arg3 A boolean flag. This is set to TRUE\$ if any results are available as a result of the Query after execution, FALSE\$ otherwise. Note that a True\$ does not necessarily imply the existence of results – a result set of zero rows might still be valid if that was the result of the script.

A boolean flag is returned by this function, TRUE for success, FALSE for an error (See Retrieving Errors below).

So, to execute a script use:

```
RetVal = QryMethod( hQry, QRY_EXECUTE$, SQLScript, "", bResultsThere )
```

Step 4 – Retrieving the Results

If you've executed a SELECT statement then you'll need to find out what you've selected, so the next step is to get back the results. For this we again use the QryMethod function, but this time we specify a different method, QRY_GETROW\$, and a different set of arguments:

arg1 Result Row (@Fm delimited like a normal LH record)

arg2 Fetch direction. The default is Forwards, but you could trawl backwards too! (If your database supports it of course <g>).

arg3 Conversion Type. When OpenInsight retrieves data from the datasource it is converted from its natural state to corresponding Basic+ types using a set of default output conversions. This parameter allows these default conversions to be overridden if desired.

This argument may be left null if desired but if it is used it will be one of the following – (taken from DSXO_API)

Conv_None\$ (0) Pure data – the fastest way to get data in. Returns data in internal DataSet structure.

SPREZZATURA

Conv_ODBC\$ (1) Returns data in the ODBC binary standard format

Conv_OI\$ (2) Returns in standard OI datatypes

Conv_ASCII\$ (3) Performs a simple conversion using Windows defined local values. This is what the Query tool uses.

We need to execute this method once for each row of data returned, in a similar manner to ReadNexting in standard Basic+. The function returns a boolean flag denoting if there are any more rows to be returned.

```
If QryMethod( hQry, QRY_EXECUTE$, SQLScript, "", bResultsThere ) Then
  If bResultsThere Then
```

```
    Loop
```

```
    MoreRows = QryMethod( hQry, QRY_GETROW$, ResultRow, "", "" )
```

```
    While MoreRows
```

```
      * Process ResultRow
```

```
    Repeat
```

```
    End
```

```
  End Else
```

```
  * Error
```

```
End
```

Step 5 - Destroying the Query Instance

Now you've got your data and have finished playing with it, it's time to put your toys back in the box for next time. Creating a Query object, and indeed a Connection object, uses up valuable Windows resources and memory, so we have to release these when we're done with them.

First off, you should cancel the query calling Qry_Method in the following manner:

```
bOK = QryMethod( hQry, QRY_CANCEL$ )
```

This good behaviour, but if you forget to do it the system will do it for you.

We then release a Query Object by calling the QryMethod yet again (useful function yes?), with yet another method, QRY_DESTROY\$. This takes no extra arguments.

SPREZZATURA

```
bOK = QryMethod( hQry, QRY_DESTROY$ )
```

Hey presto! The Query is toast! (and you'll need to create a new Query object if you want to perform any more Query functions now!)

Step 6 - Destroying the Connection Instance

The next step is to release the Connection. Before you do this however, there are a couple of points to take into account:

If you have updated the database in some way you should commit the data before you close the connection, as Commit and Rollback transactions are methods associated with a Connection, not a Query. You'll lose your updates if you close the connection before you commit!

Committing a transaction is handled by the XOMethod function, which is very similar in operation and design to the QryMethod function we've been using. However, at this point all we want to do is commit a transaction and we do this like so:

```
bOK = XOMethod( hXO, XO_COMMITTRAN$ )
```

Likewise, if we wanted to rollback we'd use

```
bOK = XOMethod( hXO, XO_ROLLBACKTRAN$ )
```

Both of these methods return TRUE for success and FALSE for failure.

Releasing a Connection Object is again performed via the XOMethod function which takes a connection handle and a method specifier.

```
bOK = XOMethod( hXO, XO_DESTROY$ )
```

A boolean flag is returned, which returns TRUE if the Connection is terminated successfully, or FALSE if an error has occurred.

Note that when you make a connection via XOInstance the system keeps a reference count of how many 'users' that connection has, incrementing it each time (so shared connections will typically have a reference count of greater than 1). Destroying the connection via XO_DESTROY\$ essentially decrements this count by 1. When the count reaches 0 the connection is truly destroyed.

Retrieving Errors

If a XO or Query error occurs at some point in your code you can find out more information by using the XOMethod and QryMethod functions respectively.

To retrieve an XO error use the XOMethod function passing it the XO handle, the XO_GETERROR\$ method specifier, and another 5 arguments which are:

arg1 - Local Error List (@Vm delimited for multiple errors)

SPREZZATURA

arg2 - Local Severity List

arg3 - Native Error List

arg4 - Native Severity List

arg5 - Error Text List

So, to use:

```
bOK = XOMethod( hXO, XO_GETERROR$, LocalErr, LocalSev, NativeErr,
NativeSev, ErrorText)
```

One other point to note: If you get an error while trying to create an XO instance (via the XOInstance function) you can still obtain the error by passing 0 in place of the hXO parameter.

To retrieve Query errors you use a similar approach to the above, only you use QryMethod with the Query handle and the QRY_GETERROR\$ method specifier. The other 5 arguments perform the same function.

```
bOK = QryMethod( hXO, QRY_GETERROR$, LocalErr, LocalSev, NativeErr,
NativeSev, ErrorText)
```

And here endeth the lesson gentle reader. As a final point, note that the Error Text List is going to be the only stuff you can use in the above error details – unless you're a walking encyclopaedia of native DBMS error codes! Look forward to more WombatÔ -clubbing next time

Bugs - we gottem - Aaron Kaplan

In recent months, there has been much alluding to a select bug in Advanced Revelation and it's appearance in any questionable action by the product. I'd like to spend my column inches here to try and explain this bug and how it can or cannot affect your system.

This bug in question exists in the C code of the Linear Hash drivers and appears in all versions of Advanced Revelation from ARev 1.14 through 3.111. It does not appear in prior versions of Revelation since the LH drivers were written in ASM in those versions. The bug was fixed for ARev 3.12

The bug itself is rather simple and very well defined. While executing a select on a large file with large keys and under low memory conditions, the readnext pointer could be come corrupted. Keep in mind, it takes all of these conditions for this to happen. Now, I can't remember exactly where we hit a large file, but in general it is in the hundred thousands. Large keys are four or more parts, each part about five characters.

When using Reduce, the system will store all the information in @CURSORS. In @CURSORS(X, 4) the current group information is stored. The system uses this value to read in all the keys in the current group, which get stored in @CURSORS(X, 2). Current group is probably misleading, since there can be a range of sequential groups stored.

SPREZZATURA

As is, RTP11 (the R/BASIC) code performs simple manipulation to move (or let RTP57A move) the group pointer up (if an ascending select) or down (if a descending select). By simple manipulation I mean $\text{Pointer} += 1$ or $\text{Pointer} -= 1$.

Now, here's where the bug comes in. There is the possibility that when the proper circumstances arise, RTP57A will corrupt the group pointer. When this happens, when RTP11 gets control, the increment or decrement will result in an improper value.

Now, what does this mean to you, the developer, and what are it's effects?

First of, in the vast majority of cases, the pointer gets corrupted to a lower value. This means that some groups are going to get processed twice, which in turn means some records are going to get processed twice.

It's real effects on your system are something I can't answer. You have to look through your code and see what will happen if a record is processed twice. For processes involving sums, the results can be devastating. For a process like an index rebuild, the only result is a waste of time. This is based on the way indexes work. Since the system stores a value and a key, if the same value has the same key twice, the system will not re-add it to the list. It's just the way it works. Your code might be the same.

So, where can this bug affect you? Well, for starters, only during RTP57A readnext calls. BTREE.EXTRACT does not make RTP57A readnext calls, so you can't have the bug there.

There are a few ways to fix this. It's fixed in ARev 3.12. It's also fixed with all the new Revelation LH products, meaning the NT Service, NLM and NPP. If you are using any of these products, you cannot achieve the bug.

Other things to consider are that memory constraints are very important. This means that all other things being equal, you are more likely to see the bug in 3.11 than you will in 1.15, since 3.11 uses much more memory.

Also, it is possible, though not probable, that the pointer will increase in value instead of decreasing. This means that you might have groups of records that are not processed.

It is also possible that you will never see it. When I was at RTI, we had reports of the bug for a few years, but we could not duplicate until we seriously crippled memory and used a 13 million record file. Even then, it only happened in less than 7% of the selects.

Suffice to say, this is a serious problem and needs to be taken seriously, but it is not the solution to every unexpected result returned by ARev

CELEBRITY INTERVIEW - Dave Goddard

Who are you? (Name, married, single, male, age etc. etc.)

David Goddard, 31, Married with 3 children.

Who do you work for?

SPREZZATURA

I am the representative for Revelation Software in Australia and New Zealand

What is your role in the RevSoft community?

As Above

How long have you been involved with RevSoft?

11 years. 9 years with XLTECH (Australian distributor) in support and sales and 2 years on the other side of the phone as a developer.

Which products do you use?

All of them. (well maybe not RevG any more.)

What do you most like about RevSoft products?

The Linear Hash filing system. It's just so flexible.

What do you most dislike about RevSoft products?

RevG is not certified for Windows 98.

If you HAD to use another database what would it be and why?

I recently did some work with MS SQL Server and it seemed to perform ok.

What are your favourite 3 books and why?

Almost anything by Douglas Adams - He's just plain funny.

Test Cricket Lists - I love cricket and stats.

The Concise Oxford Dictionary - I can't spell to save my life.

What are your favourite 3 CDs/Albums and why?

The New Album - Tina Arena - That's my fav right now.

Billy Joel Greatest Hits - not a bad song to be found

Simply Red Greatest Hits - as above.

What are your three favourite films and why?

Star Trek First Contact - My fav Star Trek movie.

Rope - Hitchcock at his best.

SPREZZATURA

The Sound of Music - wholesome family viewing.

What event in history would you most like to have been present at?

The Roswell Incident.

Who is the best president/prime minister we never had and why?

Paul Keating - Question time was never boring.

And finally

Your motto/witty aphorism

Empty vessels make the best beer mugs.

ERIC THE EMU'S AREV TIPS

The feathered flapper is delighted this month to announce the addition of a new emu chick, Erastus Emu, to the clutch. Erastus is the firstborn son of the firstborn son, and so in the name of family tradition inherits the keys to the refrigerator. Some have questioned the wisdom of the name Erastus, thinking it too unusual.

What did you expect? "Bill Smith" or something?

This month Eric's attention turns to things unique about your box.

From time to time, there is the need to get a unique ID for the PC you are on. One such case is on NT where, under REVG, @STATION does not resolve uniquely, but you want to uniquely create/name a data file. Another example is where you only want to know when your application has switched to an unlicensed machine, then quietly and discretely format C:, or even ask for a new registration ID or somesuch.

The emu has always pondered about the disk ID that appears on floppies, and on other disk media - the one under a directory listing that looks like A6B0-F430. It actually is a hash of the date and time the hard drive was formatted. This turns out to be a pretty reasonable unique ID for the physical hardware.

Under Windows 95, unlike DOS, you have to lock the disk to read it, but the ID's still present just the same.

You can also get an indication if the user has replaced their hardware, their hard drive, or even their underwear (though there may be other tell-tale signs).

The DISKID routine converts the number to a decimal, for ease of use (actually ease of writing) - a returned zero means that the drive is absent, or empty.

```
a = \02000000000000000000E9ED03030000\
```


SPREZZATURA

```

for i = 1 to 62
a := str(char(0),16)
next i

a := \00000000000000000000000002E803E050001\
a := \7560B8000DBA01002EFF1E0A00FCF8B8\
a := \00008A042C40724A1E0E1F2EA21100B8\
a := \0D44B94A082E8A1E1100B700BA0400CD\
a := \21F8BA1600B469B0002E8A1E1100CD21\
a := \558BEA3E8B46022EA314003E8B46042E\
a := \A312005DB80D44B96A082E8A1E1100CD\
a := \211F33D22E8B0E12002E8B361400B800\
a := \142EFF1E0A00CB\

open 'BP' to bp.file then
write a on bp.file,'$DISKID'
end else

call msg('Modify BP in code to an existing library name','','','')

end

* Syntax:

* DECLARE FUNCTION DISKID

* RESULT = DISKID('C:')

*** Remember to always include the drive letter AND colon...

```

On a final note, if you think that websites on the whole are bland, check out the original layout for the Sprezzatura website at <http://www.wwwvoice.com/bud/bud.html>

What the Gnu Knew

The magic minds who wrote OpenInsight anticipated a lazy gnu in their midst. So, they implemented a slight form of "inheritance" under group boxes. When you modify the enabled or disabled property of a group box, OpenInsight checks to see if any control exists completely inside the group box. If so,

SPREZZATURA

then OpenInsight matches the property of that control to the group box. So, if you set the group box disabled, all the controls inside will become disabled as well. Quick, simple and only one call.

Try it yourself and see!

Peripheral Trivia

As this issue of S/ENL was put to bed we fed the inner man with:

- ☑ TV : South Park – still!
- ☑ Book: "Pictures from the Water Trade : An Englishman in Japan" – John David Morley
- ☑ CD: Songs for Grownups – Dean Friedman
- ☑ WEB: http://www2.homefair.com/calc/salcalc.html?NETSCAPE_LIVEWIRE.src=homefair

Join us :	Send Mail to Admin@Sprezzatura.com with subject SUBSCRIBE SENL or complete our online registration form .
Leave Us:	Send Mail to Admin@Sprezzatura.com with subject UNSUBSCRIBE SENL
Change of Address:	Leave at the old address & join at the new one
Web Info:	http://www.sprezzatura.com/
Tell us what you'd like to see in S/ENL:	info@sprezzatura.com

COPYRIGHT NOTICE

S/ENL - ©2009 Sprezzatura Ltd. All rights reserved.. Portions copyright Microsoft Corporation Inc. Portions copyright Revelation Technologies, Inc. No portion of this journal (other than code segments) may be reproduced by any means, be it photocopied, digitised, transcribed, transmitted, reduced to any electronic medium or machine readable form, nor translated into any other language without the prior written consent of Sprezzatura Ltd or Sprezzatura, Inc. The moral rights of the authors have been asserted.

Disclaimer - Whilst every effort is made to ensure the accuracy of the information contained herein, neither Sprezzatura Ltd nor Sprezzatura Inc. can accept liability for the failure of anything documented herein to work nor for damage resulting from the application of methods/techniques learned herein.

TRADEMARK NOTICE

OpenInsight is a trademark of Revelation Technologies Inc. trading as Revelation Software. Microsoft, Windows™, and MS-DOS are registered trademarks of Microsoft Corporation. All other product names are trademarks or registered trademarks of their respective owners. Printed in the United Kingdom.

S/ENL Volume 1 Issue 8, 27th August 1997.

SPREZZATURA

Please encourage your correspondents to send e-mail to admin@sprezzatura.com with SUBSCRIBE SENL in the subject line to get their own free subscription. Everyone is welcome! Tell your friends about S/ENL.



MAKING DATABASES HAPPEN