

# SPREZZATURA

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

Tel: 020 8912 1010 Fax: 020 8912 1011

info@sprezzatura.com

# SEN

## January 1998

*Sprezzatura's Electronic Newsletter*

*For Revelation developers by Revelation developers*

**MAKING DATABASES HAPPEN**



# SPREZZATURA

## Contents

- Welcome..... 3
- Ramblings..... 3
- Feedback..... 5
- MYKL'S JOKE SECTION..... 7
- Getting a Handle - The FindWindow Function - Carl Pates ..... 8
- Class Names ..... 8
- Using FindWindow..... 9
- Example 1 - Finding a Window by Class Name only.....10
- Example 2 - Finding a Window by Class Name and Title.....10
- Example 3 - Finding a Window by Title Only.....10
- IT'S THAT MAN AGAIN - AARON P KAPLAN ..... 11
- Connections - Your Friends In The Business - Andrew P McAuley ..... 13
- SQL and ODBC ..... 13
- Wombat™ ..... 13
- Overview of Setting Up Wombat™ ..... 14
- ODBC Drivers ..... 14
- Define a Connection Object..... 16
- Define a Dataset Object..... 19
- But first - what is a dataset?..... 19
- Using Client/Server workspace to define a dataset..... 21
- Using a dataset with an entry form - beginner's guide ..... 22
- Conclusion..... 23
- CELEBRITY INTERVIEW - Carl Pates..... 23
- ERIC THE EMU'S AREV TIPS..... 25
- Peripheral Trivia ..... 26
- Peripheral Trivia ..... **Error! Bookmark not defined.**
- COPYRIGHT NOTICE..... 27
- TRADEMARK NOTICE..... 27



# SPREZZATURA

## Welcome

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

Argggghhhhhhh!

Ever had one of those days? The ones which culminate in your being convinced you've saved the changes to the latest SENL and power down quickly so your network consultant can check your card and then you realise that several hours work is down the drain?!!!

Oh well, at least the stress relieving effects of the Australian vacation lasted for a few weeks... On that subject a big shout out to Victor Ratnieks and Leonie and the unforgettable Eric the Emu who spoiled Elkie and I on a brilliant vineyard tour. Cheers guys, it was special!

So back to life, back to reality... (there's a song there somewhere.).

We've bitten the bullet - as of this issue we've gone PDF! So now you need Adobe Acrobat to read this - but you know that already otherwise you couldn't be reading this! The reason was simple - we wanted to be able to include screen shots and other digital stuff! I mean, how can we show you how pretty our software is if all we've got is courier?!

It's nice to be back behind the keyboard - I miss communicating with you all and it seems like months... needless to say we've all been busy here at Sprezz and things are getting better by the day! Feedback on S/List 2.0 is overwhelmingly positive and there is more to come from on that score - see later!

1998 looks like being an exciting year for RevSoft people, and jRev release schedules look achievable. However this section is meant to be generic so I'll drop that for here! Before signing off I'd just like to wish all of our readers all the best for 1998 - may you be successful in your endeavours but never lose sight of the fact that the journey is the point!

Andrew P McAuley  
Publisher

## Ramblings

What a year! 1997 closed out with a flurry of activity and our pride knew no bounds when the Works CD included S/List 2.0, our ground breaking reporting tool for OpenInsight! Whilst this is only a demo version (it truncates at one page and randomly corrupts data) it does have all of the user interface functionality of the full blown version!

For those of you who are still using Arev and haven't made the move to OI, here's a screen shot of what you're missing - a representative picture of a page from the "New Report" S/List Wizard!



# SPREZZATURA



Figure 1

We've been having fun extending the TCL functionality - and we didn't realise just how much we'd missed Eval until we started using it again!

What has been most pleasing is the amount of queries to the effect of "What is S/List written in?". People always seem so surprised when we reply "OpenInsight"! They seem to expect C++ or Delphi! This is leading to quite a few consultancy assignments helping people rethink their user interfaces!

This is always pleasing as we love getting involved helping people to sell more of their own product!

As promised in the last issue we can now provide details of our US Office for those who'd like to deal directly with us without a time zone problem! Sprezzatura Inc. are now established at

Sprezzatura Inc.  
1800 JFK Boulevard  
Suite 300  
Philadelphia  
PA 19103  
USA

Tel: +1 267 238 3814  
Fax: +44 (0) 20 8912 1011

E-mail: [info@sprezzatura.com](mailto:info@sprezzatura.com)

We can provide training and consultancy drawing on both US and UK personnel so whatever your RevSoft needs give us a call and we'll bend over backwards to help!

We're currently working with two companies on technology to extend OpenInsight. Coyote Consultants produce a package which enables Web Publishing via an ActiveX Server. Once we've completed the implementation we'll be able to embed the server in documents and use VBA to access OI and AREV data!

The second company produces an alternative Multi-valued variable length filing architecture called jBase. The intention is to provide an interface to this architecture for those who want multi-million row databases without using SQL Server.

As developments proceed with these products we'll naturally keep you posted!

We're grateful to the lab for putting to rest one niggly little problem that has been bugging us for a while. Essentially when adding edit tables to a form we could never quite get them to align with our 5x5 grid design. As ever the answer is stunningly simple if you know how! Edittables use dialog units



# SPREZZATURA

internally and thus are sized in increments of four pixels. So setting the grid to 4x4 removed this annoying little problem.

By the next issue of SENL we should be able to announce our Tech Bulletin Number 2 - on compiled window structures - along with a useful utility to generate windows on the fly without using a base window to start from - in other words completely soft windows - so no network traffic! Ideal for collector type windows! I know that a lot of you are looking forward to this so we'll post late breaking news about it on our web site as soon as practical!

Finally we're pleased to announce our DSO Tutorial! We've been using dataset objects intensively internally for the best part of a year now, and finding them the easiest way of accessing normalised data in other systems via ODBC links. Whilst the product is great it does have to be said that the documentation is a little lacking. We hope to rectify that! A major vote of thanks to the Lab for their help in ensuring that the DSO tutorial remains accurate - you help is, as ever guys, really appreciated!

Andrew McAuley  
Publisher

## Feedback

N J Wong writes from Multi Value Systems in Singapore

"I've just finished reading SENL 10. Another great issue. Loved the "GUY-JOKE".

"Thought you might like the following article which is currently circulating around the web/email servers here in Singapore. I haven't tried the part about Excel95 as I don't have it installed. Sounds real 'scary' ... "

Okay, that's the first part, now for the second part, for those of you fellas who still have the OLD excel 95 (not office 97) then try this out :

- 1.open a new file
- 2.scroll down until you see row 95
- 3.click on the row 95 button, this highlights the whole row
- 4.press tab, to move to the second column
- 5.now, move your mouse and click on help THEN about microsoft excel
- 6.press ctrl-alt-shift and click on the tech support button simultaneously
- 7.A WINDOW WILL APPEAR, TITLE : THE HALL OF TORTURED SOULS

this is really eerie okay...it has a doom style format and you can walk all around the hall...and on the sides of the walls are the names of the tortured souls...



# SPREZZATURA

8. NOW WALK UP THE STAIRS AND THEN COME BACK DOWN, FACE THE BLANK WALL AND THEN TYPE IN EXCELKFA

this will open the blank wall to reveal another secret passage, walk through the passage and DO NOT fall off, when you get to the end, you will see something really really eerie...

It goes without saying that we've tried it at Sprezz and it works - so thanks for that! <g>.

Olwen Williams writes from New Zealand

"As a person very experienced in the area of reduction I would like to take exception to your joke about Sarah Fergusson, Oprah Winfrey and Roseanne Arnold who I regard as being light-weights.

Surely you could find more worthy people than these to use in your joke which was extremely offensive to those of us who REALLY need to reduce. Too often we see second-rate examples of the group of fat ladies held up as examples.

Olwen (not singing in Wellington)"

And she isn't joking - she has made major strides in reduction. Congratulations Olwen! Anyone else wanting to check her progress checkout her homepage at <http://homepages.ihug.co.nz/~olwen>.

Someone who shall remain nameless writes "Wow!!! Great SENL - Love all the code - ....." - he's remaining nameless because we suspect he only wrote it to see his name in print! (<g>).

Paul Watson from MCV Global Systems Inc. writes

"I sent an e-mail to you yesterday requesting back issues of your SENL newsletter. You have called me twice (in Los Angeles, California) to tell me that you couldn't deliver the e-mail. That's service! I don't think you could find anyone in the US willing to follow-up on a problem like that. I am truly impressed!"

Thanks Paul - I'm actually reprinting this to pat the responsible Sprezz staff on the head! You both know who you are so well played!

Charles R. Nichols writes from the Vermont Chamber of Commerce,

"I know the following is what you get paid to answer, but at this point I am quite frustrated and hope you can perhaps help me a little. I have used AREV since it first came out and I am just starting to use OI with a very small application (2 windows).

My real questions are:

A definition

- I have a edit table control that is a mv and has 2 rows showing in the window. If I have a value entered and go back to edit it seems that the only option I have is to retype the entire line - I



# SPREZZATURA

can't find a way to place my cursor in the row and edit starting at that place. This would seem to be quite basic and I wouldn't think I would have to program to do that.

- ☑ If I have more than two values entered into the same edit table control with only two rows visible and the 2nd and 3rd rows are showing; when I leave that control the values do not reset so that the 1st and 2nd are the ones visible. Is this the normal way the system operates or am I doing something wrong."

Well Charles, whilst we do offer technical support on a "Pay per call" basis (only \$30 a unit and YES we do take credit cards! <g>) we always welcome enquiries which allow us to fill column inches in SENL!

The thing to remember with the edit table is that it is a third party control (checkout [www.protoview.com](http://www.protoview.com) for more details) and so some keystrokes aren't always self-evident. The edit table emulates a spreadsheet and those of you familiar with Excel will know that pressing F2 takes you into edit mode. Well the same applies to edit tables. The F2 can be set programmatically on a GOTFOCUS and POSCHANGED event using code posted on Revelation Web Site at <http://www.revelation.com/85256311006a3217/2b270df8a7d6fb488525656b005fed64/ca057d5ac3054eae852563bf005a2233?OpenDocument> (I love these snappy names! <g>)

Secondly, silly as it seems, you can guarantee that if it did reset automatically, someone somewhere would complain that it had broken their application! So simply add a quickevent on the LOSTFOCUS event which executes the stored procedure SET\_PROPERTY passing it the following parameters

```
'@SELF', 'TOPPOS', [1,1]
```

The [1,1] is a neat trick to pass a field mark delimited list. See you don't have to cut code!

## MYKL'S JOKE SECTION

IF MICROSOFT MADE CARS:

1. Every time they repainted the lines on the road, you'd have to buy a new car.
2. Occasionally your car would just die on the motorway for no reason, and you'd have to restart it. For some strange reason, you'd just accept this, restart and drive on.
3. Occasionally, executing a manoeuvre would cause your car to stop and fail to restart and you'd have to re-install the engine. For some strange reason, you'd just accept this too.
4. You could only have one person in the car at a time, unless you bought a "Car 95" or a "Car NT". But then you'd have to buy more seats.
5. Macintosh would make a car that was powered by the sun, was twice as reliable, five times as fast, twice as easy to drive - but it would only run on five percent of the roads.
6. The Macintosh car owners would get expensive Microsoft upgrades to their cars which would make their cars go much slower.



# SPREZZATURA

7. The oil, engine, gas and alternator warning lights would be replaced with a single "General Car Fault" warning light.
8. People would get excited about the "new" features in Microsoft cars, forgetting completely that they had been available in other cars for many years.
9. We'd all have to switch to Microsoft gas and all auto fluids but the packaging would be superb.
10. New seats would force everyone to have the same size butt.
11. The airbag system would say "Are you sure?" before going off.
12. If you were involved in a crash, you would have no idea what happened.

## Getting a Handle - The FindWindow Function - Carl Pates

Handles are used an awful lot in Windows programming. When we need to interact with other windows or objects we usually have to refer to them by their Handle, which is a unique number that Windows uses to refer to them itself. In almost all instances of Windows API programming you need to obtain an Handle to the window/object in question first and pass this as part of the function call to let Windows know what you are referring to.

Getting window handles of OpenInsight windows is easy. They all have a HANDLE property which the Presentation Server looks after for you, maintaining the relationship between an OI window name and it's Handle internally. However, this method can't be employed if we want to look outside of OI and interact with a non-OI window, so we have to call a Windows API function to find the non-OI window Handle for us - hence FindWindow function.

### FindWindow

This very useful function allows us to find a window handle by three methods

- 1) By the window's Class Name ( see below ), or
- 2) By the text in the Title bar of the window, or
- 3) By both of the above

It does this by simply comparing the above information against the currently running windows and returning the handle of the window that matches them.

### Class Names

A Class Name is basically a string identifying the type of window to look for. When a window is first created it passes some details about itself to Windows, one of these being it's class, which Windows uses to identify the procedures used to process messages to that window.

Different applications use different Class Names. Eg



# SPREZZATURA

OpenEngine's class name is `Arev`,

OpenInsight's is `RTI\_Oinsight` ,

MSWord's is `OpusApp`

on so on.

Unfortunately if you don't know the Class Name of the window you're interested in, you have to use a tool like Microsoft's Spy++, Borland's Winsight to find it out (or know the window's handle in which case you don't need FindWindow do you?!)

## Using FindWindow

As FindWindow is exported from a Windows module (USER.EXE), it would normally have to be prototyped before it could be used in Basic+ (See Chapter 7 of the Basic+ Programmers Reference Manual for more on DLL Prototyping ), but those nice guys at RevSoft have already done this for us, so we should be able to use this straight away.

FindWindow is defined in the Windows 3.1 SDK as:

```
HWND FindWindow(lpszClassName, lpszWindow)
```

```
LPCSTR lpszClassName; /* address of class-name string */
```

```
LPCSTR lpszWindow; /* address of window-name string */
```

lpszClassName Points to a null-terminated string that contains the window's class name. If this parameter is NULL, all class names match.

lpszWindow Points to a null-terminated string that specifies the window name (the window's title). If this parameter is NULL, all window names match.

This means that we have to pass it two pointers to Char(0)-terminated strings and get back an integer which represents the handle we want. Zero is returned if there are no windows matching the criteria specified.

However, FindWindow has been prototyped by REVSOFT in DLL\_USER as:

```
SHORT PASCAL FindWindow( LPCHAR, LONG ),
```

which means that we can pass a Char(0)-terminated string for the Class name in the first parameter (LPCHAR) and OpenInsight will automatically pass the pointer to this to the FindWindow function for us. The second parameter ,the Title, is expecting us to pass the pointer to the data ourselves (LONG), so we need to do a slight bit of work before we can use it, as you'll see in the examples below:



# SPREZZATURA

## Example 1 - Finding a Window by Class Name only

This is the easiest method and involves simply passing the Class Name:

```
Declare Function FindWindow

/* Finding the handle of OpenEngine */

OEClassName = `Arev' : \00\ ; * Don't forget the Char(0) on the end!!

hwndOE = FindWindow( OEClassName, 0 )
```

## Example 2 - Finding a Window by Class Name and Title

This gets a little more complicated as we have to get the pointer to the Title ourselves. Note the use of the LockVariable statement -Windows likes to move things around in memory, so this is a precaution to make sure that the pointer we pass is still valid when it gets to the FindWindow function.

```
Declare Function FindWindow
/* Finding the handle of OpenEngine */
OEClassName = `Arev' : \00\ ; * Don't forget the Char(0)
* on the end!!
OETitle = `OpenEngine' : \00\
LockVariable OETitle As Char ; * Lock this bit of memory so
* Windows doesn't move it
pOET
tle = GetPointer( OETitle ) ; * Get the pointer
hwndOE = FindWindow( OEClassName, pOETitle )
UnlockVariable OETitle ; * Don't forget to unlock it!!
```

## Example 3 - Finding a Window by Title Only.

It's at this point that we can't use the FindWindow function as prototyped by RevSoft, because to find a window by Title only we need to pass NULL ( 0 ) to FindWindow. As the Class Name parameter is prototyped as LPCHAR OpenInsight will convert whatever you put in there to a string and pass the pointer to it accordingly. So if you try

```
hwndOE = FindWindow( 0 , pOETitle )
```

the 0 will be converted to the string "0", as would "" (an empty string) which would pass a pointer to that string, hence the value received by FindWindow would never have a value of 0. To workaround this we will have to re-prototype the FindWindow function so that OpenInsight passes the first parameter exactly 'as is'. Using LPVOID in place of LPCHAR achieves this for us. Also, we need to alias this 'new' prototype of FindWindow with a different name so we don't overwrite the original definition. ( For more information on Aliasing DLL Functions refer to the document of the same name in the OI Knowledgebase on the RevSoft website

(<http://www.revelation.com/WebSite/knowledge.nsf/88e0581886ff6a0d8525651c004caecf/c1e211ced34167e38525632500745e8e?OpenDocument>)

So, we can redefine it thus:



# SPREZZATURA

SHORT PASCAL FindWindow( LPVOID, LPCHAR ) As FindWindowByTitle

To achieve this open the DLL\_USER record in the SYSPROCS table in the System Editor and add the line definition above to the bottom of the list. Next, save the record, and in the Exec. Line type `RUN DECLARE\_FCNS "DLL\_USER"' and press Enter. The system will now create a new record in SYSOBJ called \$FINDWINDOWBYTITLE that points to the FindWindow function in USER.EXE, and we can now use this function as we would any other in Basic+. (Note that the second parameter has also been changed to LPCHAR. Now OpenInsight will pass the pointer to the WindowTitle for us as in Example 1 above and we won't have to get the pointer ourselves.) So, now to find our window by Title:

```
/* Finding the handle of OpenEngine */
```

```
Declare Function FindWindowByTitle
OETitle = `OpenEngine' : \00\ ; * Don't forget the Char(0)
* on the end!!
hwndOE = FindWindow( 0, OETitle )
```

## IT'S THAT MAN AGAIN - AARON P KAPLAN

As promised, here's more in our ongoing discussion about selection and reduction. This issue, we'll delve deep into the guts of the system, or at least as far as my memory allows, and talk about what the system is doing under the hood.

The first thing the REDUCE does is parse your statement. If you think REDUCE and BTREE.EXTRACT have complex parameters, you should see what the system really requires. Dictionary field references, case sensitivity options, conversions and maybe a few other things all get tacked onto the field names and criteria parameters. After all this is done, the system calls the first of the meta programs to optimize the criteria even further. This meta program, WITH0.RUN, also fully validates the syntax of your query.

Meta is really good for this sort of work. The R/BASIC and Basic+ compilers are implemented in Meta. As I told Joe C. once, if you don't know what it is, don't even think about touching the manual. Basically though, meta is a rules based engine. As such, it's excellent for parsing and validating that it's input passes certain rules. Hence it's major use in the system is the compilers, reduction criteria validation, and automatic code generation (R/LIST and OpenList output, SI.TRANS records, protection code and the like). I once posted a really big block of the stuff on CompuServe a few years back. I can place a good part of my hair loss on this stuff. It's way nasty. Just for reference, all the programs in SYSOBJ that end in RUN and do not start with a \$ are meta programs.

Anyway, after this meta is run, and everything's OK the system runs the two code generating metas. One generates code for non-indexed fields. The other generates code for indexed fields. When all this code is finally assembled, it loaded them into the stack as an RTP20.x program.

The code executed by RTP20 returns a logical TRUE if the criteria is met and a logical false if not. Using the select statement from the previous issue

```
SELECT AREVKB WITH LEVEL = "EXPERT" AND WITH DATE > "01/01/91" BY AUTHOR
```

the appropriate RTP20.x program will contain code similar to



# SPREZZATURA

RETURN {LEVEL} = "EXPERT" AND {DATE} > 8402

For those that don't know about this syntax, R/BASIC and Basic+ allow for sort of syntax. The reason boils back to meta code and how syntax rules are processes and the object code is evaluation.

If we were to change the code to

```
retVal = {LEVEL} = "EXPERT"
```

{LEVEL} = "EXPERT" gets evaluated to either 1 or 0 and then assigned to retVal.

Try it in your code. You'll see that it works. Just to take this slightly further, for those that are having trouble following,

```
retVal = {LEVEL} = "EXPERT"
```

```
if retVal then
```

```
do something
```

```
end
```

works the same as

```
if {LEVEL} = "EXPERT" then
```

```
do something
```

```
end
```

The IF statements works on the IF TRUE\$ THEN It doesn't care what expression is in the statement, as long as it can evaluate it. So, this means exp1 = exp2 is a valid statement. {LEVEL} = "EXPERT" is

is evaluated for each record, and if true, the ID is placed into the select list.

This information is programmatically stored internally in the same way. Actually, it's stored slightly different for cursor 0 selects and cursor 1-8 selects.

For cursors 1 through 8, selection information is stored in the @CURSORS array. The equates for these (and other selection information) have been slightly exposed. In ARev 3.12 (and maybe older versions) you'll find SELECT.CONSTANTS in the SYSINCLUDE file. Unfortunately this record doesn't seem to be in OpenInsight. The equates we're concerned with here start with CURS. Using these variables, you can see what groups of keys have been returned in this block, the group range the keys are located in, and host of other stuff.

We'll have to leave the exact calculations of this as an exercise for the reader to complete. After all, we can't give away all our secrets, can we?



# SPREZZATURA

There are some fun and games you can play with cursors. The SWAP.CURSORS program, available in OpenInsight and in the 2nd Advanced Revelation Developers Series Disk allows you to move the contents of one cursor to another.

SELECT.SEEK allows you to reposition your location in a select list. This program was documented in the Developer Series manuals for Advanced Revelation and is available in Advanced Revelation and OpenInsight.

And so, until next time....buy bye and buy bonds.

## Connections - Your Friends In The Business - Andrew P McAuley

### SQL and ODBC

For years Revelation Software distinguished itself from the competition by being able to access a disparate range of data sources but using the same front end tool set - using bonding technology. The rest of the world looked on and obviously somewhere an idea was born! As Windows technology became more prevalent, the idea of a universally open standard permitting access to any data source from any environment became more widely accepted and the ODBC (Open DataBase Connectivity) standard was born.

Something which made this sort of interoperativeness even more achievable was the emergence of SQL (Structured Query Language) as the de facto standard database management language. Once this had been established as the standard, you could pretty much guarantee that every database vendor would ensure that their database could talk SQL! Of course, SQL is designed to work off normalised tables, so even though OpenInsight can use SQL it still isn't the most obvious tool of choice for working with Linear Hash tables.

For some time, Revelation lagged behind the marketplace in terms of ease of connectivity to external environments, until in 1995 a new development made access to external data as easy as access to internal data. That development, not being created by a marketing department, lacked a user friendly buzz word name, so we'll have to settle for "The Connection Object/Data Set Object Model" - even better, "The Windows Object for Being Accessible Totally" or Wombat™ for short.

Using Wombat™ it is now possible to get native access to SQL Server and Oracle and ODBC access to pretty much anything with an ODBC driver.

### Wombat™

The theory of operation is simplicity itself - you describe to the system where you'd like it to get its data from (your connection object) and you describe what information from the wealth of data stored at that location you'd like to see (the dataset object). The dataset then becomes the conduit through which your application retrieves and updates data in the target environment.



# SPREZZATURA

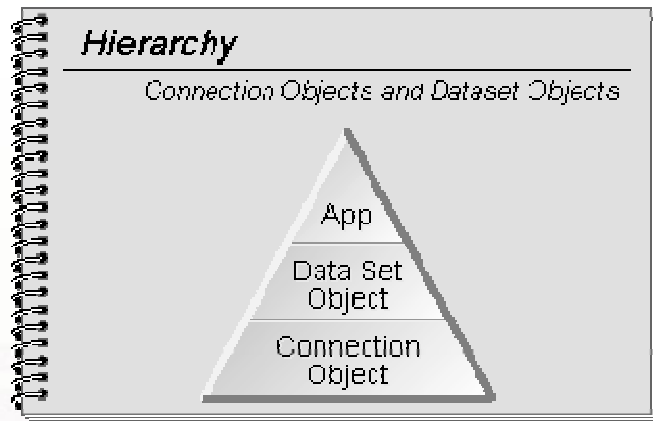


Figure 2

By far and away the easiest way of actually understanding the use of Wombat™ is to actually use the product. With this in mind, in the next issue of SENL we'll be going through a worked example of setting up and using a dataset. To make this possible of course, you'll need to have ODBC drivers loaded on your Windows machine. Microsoft's EULA prevents us from shipping these drivers with SENL so we'll have to tell you where to get them from! Try <http://www.microsoft.com/syspro/technet/technol/odbc/prodfact/odbcqa.htm> as a starting point!

## Overview of Setting Up Wombat™

The steps involved in setting up a working Wombat™ installation on your machine are as follows

Ensure that you have valid 16 bit ODBC drivers installed on your machine

Define a connection object

Define a dataset object

## ODBC Drivers

It doesn't matter how carefully you assemble your connection objects and dataset objects, if OpenInsight can't see a valid ODBC (or SQL Server/Oracle) connection, you won't be able to use them. Essentially this configuration is done outside of OpenInsight at the Windows level. This makes perfect sense if you think about it, the whole point of ODBC is to allow any package to query the information, so the information has to be stored centrally.

With any luck the package to which you wish to connect will come with full installation instructions for the ODBC driver because otherwise it is a nightmare of editing ODBC.INI files and associated components.

The major thing to watch out for is that it is preferable that all drivers are 16 bit. This isn't to say that 32 bit drivers can't be used - just that it is better if they are not! (Microsoft does support the use of 32-bit drivers from 16-bit applications in Windows NT, but specifically does not support it in Windows 95. Most 32-bit drivers do work from 16-bit applications in Windows 95, though) For the examples



# SPREZZATURA

we're going to use I'm deliberately going to flaunt this rule and use my Access 7.0 copy of the Northwinds mdb file. To make OpenInsight see it I went through the following steps.

Go to the Control Panel and launch 32 bit ODBC - this (if ODBC drivers are installed on your machine) will give you a screen similar to below :-

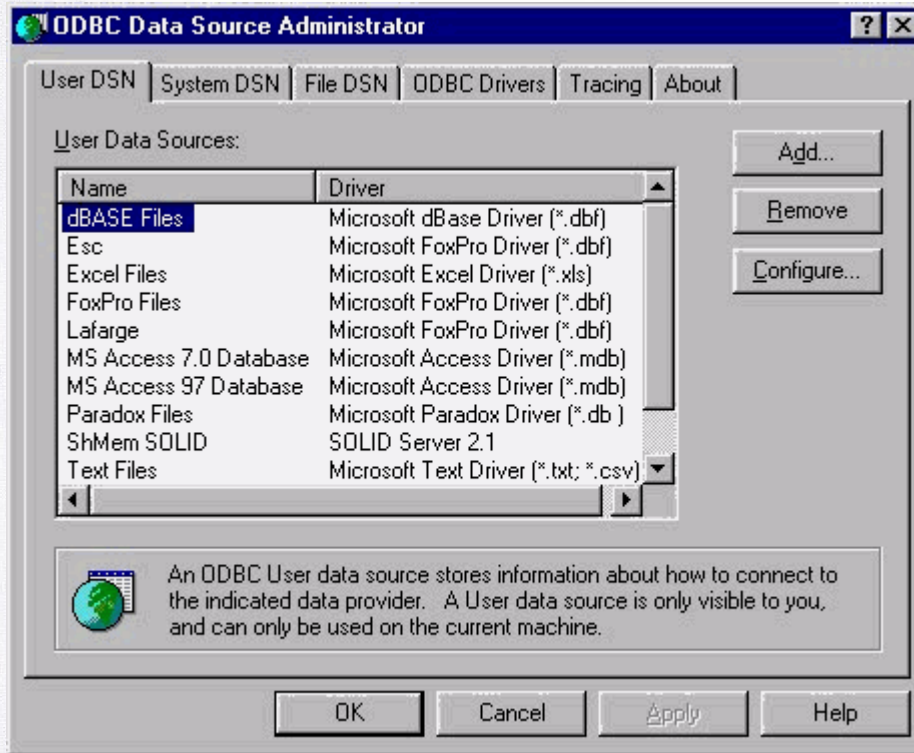


Figure 3

Now double click on MS Access 7.0 Database and fill it in like below :-



# SPREZZATURA

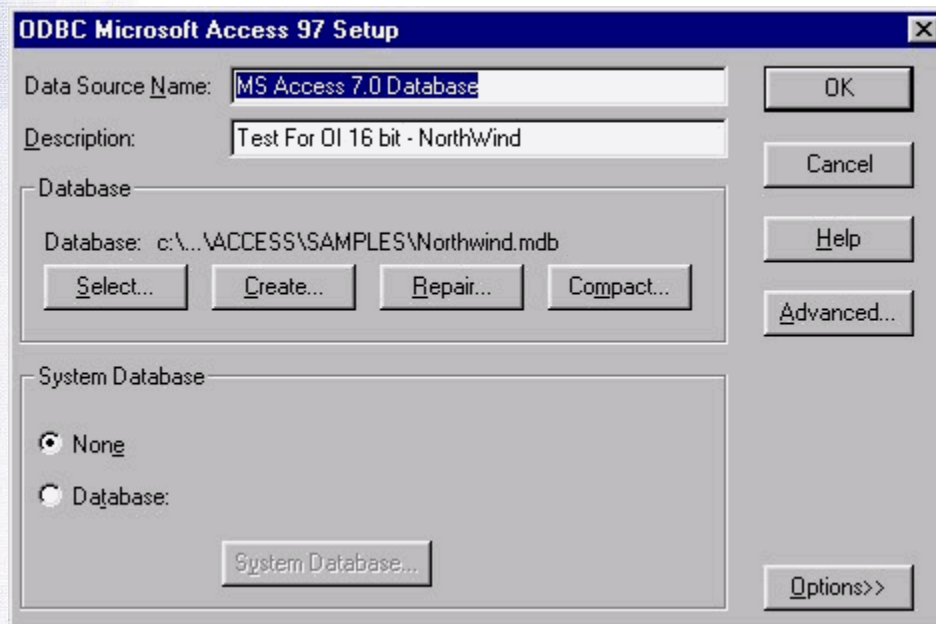


Figure 4

Finally, we need to follow the directions from the White Paper at [www.revelation.com](http://www.revelation.com) called "Seeing 32-bit ODBC Data Sources in OI", and edit our ODBC.INI file so that it looks like this

[ODBC Data Sources]

NorthWinds=Microsoft Access Driver (\*.mdb)

DiningIn=Microsoft Access Driver (\*.mdb)

BANDAG=Microsoft Access Driver (\*.mdb)

MS Access 7.0 Database=Microsoft Access Driver (\*.mdb) (32 bit)

note that final line - copied directly from the [ODBC 32 bit Data Sources] section of ODBC.INI. Note that if you've filled in your details exactly as above, adding that final line should be all you need! Now when we ask OpenInsight to look for ODBC datasources it will see the 32 bit Northwinds database!

## Define a Connection Object

In contrast to the above, nothing could be easier! Simply launch the client server workspace and choose File New. You'll be prompted for whether you want to create a connection object or a dataset object and you'll respond in favour of the former!

Then when presented with the Data Source Type, accept the default of ODBC (or choose it if it is not the default) and double-click on the Value entry for DataSource. You'll see a popup not too dissimilar to that below (Figure 5). Note in mine that I have a 16 bit and a 32 bit copy of Northwinds - the 32 bit copy is called "MS Access 7.0 Database" because that's what I called it in my 32 bit ODBC setup!



# SPREZZATURA

Select the "MS Access 7.0 Database" option and save the connection. If you wish to ape me slavishly you can call yours MSACCESS7 too!

And that's all there is too it! Your first connection object.

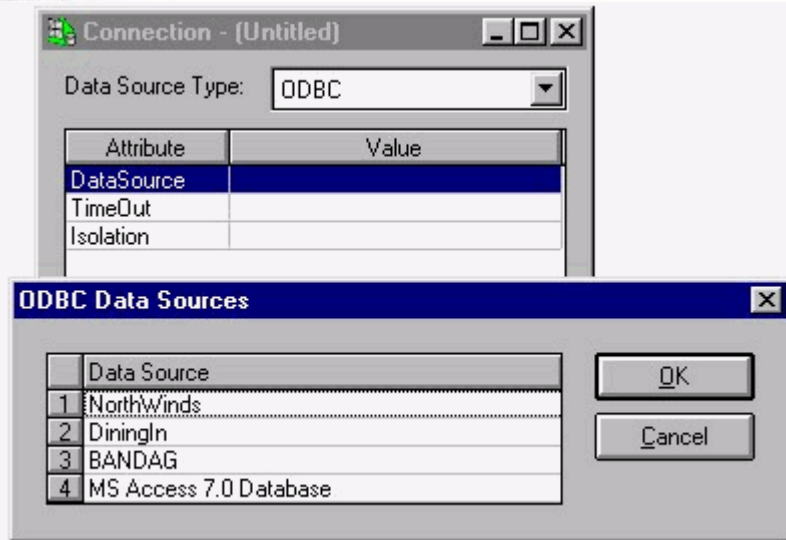


Figure 5

Interlude - Now we've got it what can we do with it?

It's all very well having a connection - but what can it be used for? Remember our earlier hierarchy? It seems from that that we can't actually do anything until we have a dataset level, but that isn't completely true. You see, the connection object (and at this stage it seems appropriate to start referring to it as the XO for reasons that will become abundantly clear when you get round to using the system routines that support XOs) is simply a way of telling OpenInsight the location of a datasource so that OpenInsight can communicate and get the information that the DataSet or any valid SQL query wants.

So yes, if we had a dataset we could start retrieving and updating data but just to get into the mood why don't we play with the connection object using SQL?

Before you panic - we can do simple stuff without even touching SQL because the Client Server workspace has a cool utility built into it called the Query Window. So invoke the Query Window by choosing, surprisingly enough, the Query Window option from the Tools Menu. You'll be asked for (slightly confusingly) a datasource to connect to - and that's our friend the XO. So choose the XO you've already created - mine was called MSACCESS7 remember? Now if you've followed my instructions properly, there won't be any passwords or anything so you can just choose OK, and OpenInsight will establish a dialogue with the Northwinds database before displaying the Query Window.

Now you're in the Query Window, the Query Menu becomes active, so we're going to ask it to show us all of the "objects" in the Northwind database - List\_Volume if you will. To do this, go to the Query Menu and choose "List Objects". You'll see a screen similar to that below



# SPREZZATURA

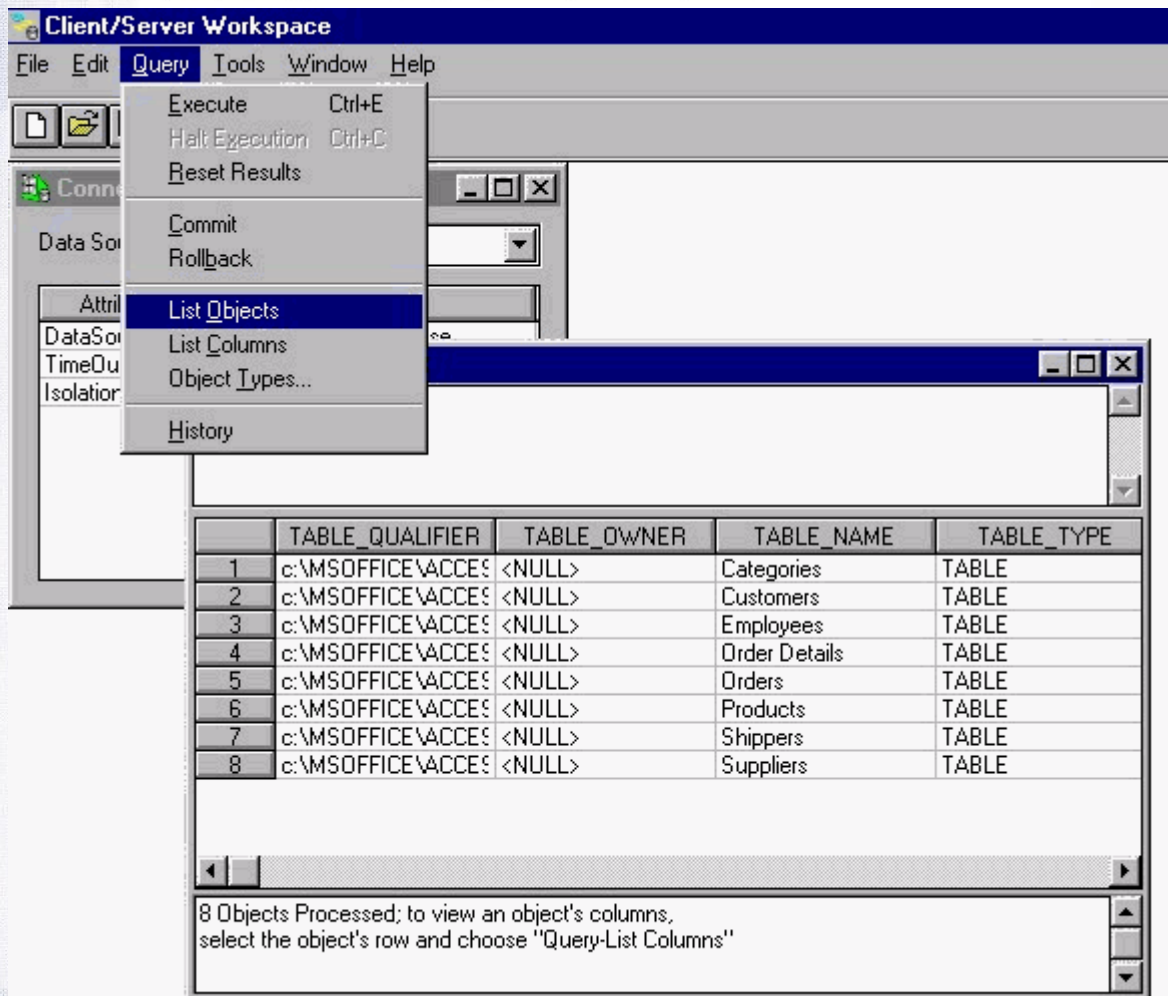


Figure 6

Of course, I cheated slightly, I chose "List Objects", got the list up and then chose the Query Menu again so I could show the screen populated. That's why my "List Columns" option is available and yours wasn't! You see you can't list the columns in a table until you know what tables you have.

Double-clicking on a table will show you the column definitions for that table and doing that gives you the chance to try out your first SQL statement - we're going to see what's in the Suppliers table. When you double click on Suppliers you'll note that the Suppliers table has a column called CompanyName another called ContactName and another called City. So into the top frame of the Query Window, type

Select CompanyName, ContactName, City From Suppliers Order By CompanyName

then press Ctrl-E for Execute!

You should see the screen below!



# SPREZZATURA

Congratulations, you've now used an XO to send a SQL Script to the Access database via the Query Window!



Figure 7

(The complete source for the Query Window tool is shipped in OpenInsight Works 3.5. To edit the source, log into the SYSPROG application; the window name is XO\_QUERY and the procedure name is XOQUERY.)

## Define a Dataset Object

### But first - what is a dataset?

Well, we don't want to smother you with information overload - so take a break, have a coffee, think nice thoughts and then come back....

Are you sitting comfortably?

Then I'll begin...

Firstly I think it's necessary to actually define what a dataset is before actually attempting to create one. We in the Linear Hash world are a little spoiled and hidden away from the complexities of the "normal" world with their normalised databases. You see, out there normalised databases don't exist in quite the discrete self contained manner that our files and records do. One of the whole points of SQL is that it makes it possible for you to have "Views" on a table. In other words to create alias tables which don't contain all of the information the base table contains, or which contains it in a different order.



# SPREZZATURA

Your saw an example of this with our Select statement above. If we had have used the opposite of a Select statement - an Insert statement (and before a SQL purist corrects me I use the work "opposite" loosely), to write back CompanyName, ContactName and City only those columns would have been affected. If you set fields 4, 7 and 20 of a record to CompanyName, ContactName and City in a Revelation database and wrote the data back you'd wipe down fields 1-3, 5 and 6 and 8-19. Not so with SQL. (This paragraph is a little hard to understand. -cp)

Now think about that as an idea. On my suppliers table I may have hundreds of different columns, but for basic details I only want to work with say 5 of them. Rather than always working with the hundreds, I could define a "dataset" which just contained the 5 I wanted. I'd need firstly to tell OpenInsight which XO to use for the dataset. Once I've defined the dataset there are a finite amount of things I'd want to do with it. I'd want to look at it, add to it, modify it and remove rows from it.

Now remember that all of this maintenance is going to be done with SQL - regardless of whether we're using an ODBC driver or going native (going native just means that OpenInsight speaks directly to SQL Server or Oracle without going through a slow thinking middleman called ODBC). So what we need is four different SQL statements to do our work for us. SQL aficionados call these "Scripts" - for all the world's a stage and some of us are clowns.

That's really all we need except for one very slight complicating factor. We need OpenInsight to be aware of the rules as to how data should be converted before being written back to the other environment. You see again, we're spoiled by OpenInsight. We don't have to worry about data types and making sure we don't put strings into numeric containers and preallocating memory before copying strings, but SQL databases do. So we need to say to OpenInsight "I know we think everything in the world is a string BUT please change this string back to a number before you try and pass it back to that SQL database or you'll GPF my machine and I'll feel very foolish".

So lets look at that again - what do we need for a dataset to work?

- Define which XO to use for the dataset.
- A definition of what columns are in the dataset
- An SQL script to retrieve the columns from a source
- An SQL script to add new columns to the source
- An SQL script to modify existing columns in the source
- An SQL script to delete rows from the source
- A one to one explanation for every column of what data type it ought to be converted to/from.

Fortunately the Client Server workspace makes it easy for us!



# SPREZZATURA

## Using Client/Server workspace to define a dataset

Well back to some proper work away from all that theory! Bring up your client/server workspace again and choose File New. This time indicate that you'd like to add a new dataset. By now I'm sure you're getting the gist of how this'll work - so give it a title and tell it to use the XO we created before.

Click on the Scripts tab - remember we need to tell the system what columns are in the dataset? Well before that we have to identify the table. So click the "Use Table" Button and choose the Supplier table and columns SupplierId, CompanyName, ContactName, ContactTitle and Phone as follows - then press OK. (Figure 8)

At the moment the system only has enough information to construct the Select script and that's exactly what it does. To go any further we need to define columns, so click on Define Columns and note how the system generates the defaults for you. These are almost always correct so only override if you know what you're doing! The only thing to change is that as we know supplier reference is the unique key, we let OpenInsight know this by setting Key to "Yes" for SupplierId. Just doubleclick and choose from the popup. (Figure 9)

Finally we have enough information to define scripts, so click on the Define Scripts button and let the system finish off generating insert update and delete. Note that if you'll never use the dataset for any of these functions you don't need to generate the script.

You may find it interesting to browse the scripts to see what they're actually going to do. If you don't know SQL it will be a valuable learning experience and if you do you may be horrified by clumsiness of some of the statements. If you know what you're doing you can always override them but for now I'd just leave them as they are.

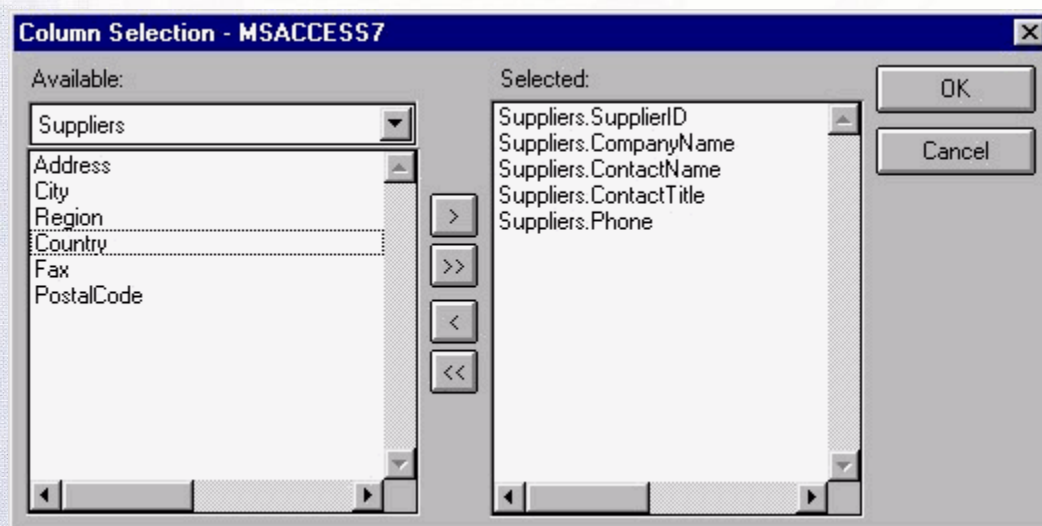


Figure 8



# SPREZZATURA

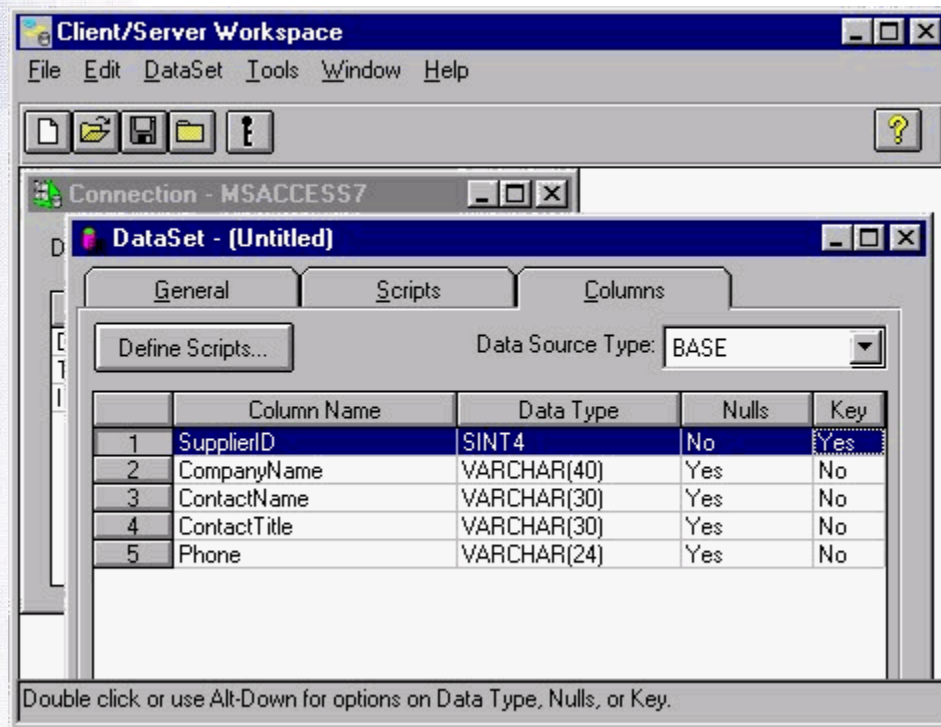


Figure 9

Save your dataset - again I called mine MSACCESS7 and then stretch your hands over your back and pat! Congratulations your first dataset awaits use!

## Using a dataset with an entry form - beginner's guide

If you've never done this before you are in for a VERY pleasant surprise. This makes falling off logs look like brain surgery.

Invoke the Form Designer.

Choose New Form

For DataSource choose DataSets

Choose your Dataset by double clicking on it.

Select all five columns and add them

Click on OK



# SPREZZATURA

Column name	Control type
SupplierID	Edit line
CompanyName	Edit line
ContactName	Edit line
ContactTitle	Edit line
Phone	Edit line

Figure 10

You now have a fully functional entry form for an Access database.

Test run the form and press F5. This will load all the rows into a browse list and you can navigate them using F3 for next and F4 for previous. (Note that under certain circumstances you will get a "Unique Key Validation Error" the first couple of times you press F5. This is a Win 95/32 Bit Driver bug and goes away with other configurations.). Feel free to make changes and when you want to update the database permanently press F9!

## Conclusion

Thus endeth the first lesson. Datasets aren't so scary are they? In next months lesson we'll look at behind the scenes with the dataset spy and start to consider some more complex issues!

## CELEBRITY INTERVIEW - Carl Pates

> Who are you?

The new number 2.

> Who is number 1?

I am not a number I am Carl Pates, engaged, male (YY chromosome), 28, 5 children and very, very tired (but satisfied after all that work :-).

> Who do you work for?

Sprezzatura and the greater good of mankind.



# SPREZZATURA

> What is your role in the RevSoft community?

Consultant and UI smartarse.

> How long have you been involved with RevSoft?

5 years or so (doesn't time fly when you've been having fun?)

> Which products do you use?

OpenInsight, ARev ( and Delphi and C++ when I can't get what I want...<g> )

> What do you most like about RevSoft products?

Basic+ string handling and MV data structures.

> What do you most dislike about RevSoft products?

No support for Doom and Quake (I mean, what sort of software manufacturer is this anyway?)

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

I wouldn't...You can't make me. So there. I'd do something else instead like program games or talk to pot plants ( which is a lot like talking to the others in the office really )

> What are your favourite 3 books and why?

The Mists of Avalon - Marion Zimmer Bradley. A alternative spin on the Arthurian legend.

The Dragons of Heorot - Pournelle and Niven. 'Aliens' for the thinking man (i.e. without the marines)

The Lord of the Rings - Nuff said.

(The office obviously has some strange affinity for sad bastards - Ed)

> What are your favourite 3 CDs/Albums and why?

Floodland - Sisters of Mercy. Beecos it is gud. Well it just sounds perfect. Every track. That's it really.

Heaven & Hell - Black Sabbath. Hey this has Neon Knights, Children of the Sea, Die Young - all this on one album and you have to ask why!

Love - The Cult. This was the first stuff I learned to play on guitar because it's so easy!

> What are your three favourite films and why?

Star Wars - Hey, do you really need to ask why for this too? Some things are obvious, yes?







# SPREZZATURA

```
A := /13BA03002EFF1E0A002E880E1300B42B/
A := /CD212E8A2E11002E8A0E12002E8A3613/
A := /00B200B42DCD21CB/
if len(A) = 104 then
open 'BP' to bp.file then
write A on bp.file,'$TIMESET' else
call msg('Modify BP in code to a file you have',' ',' ',' ')
end
end else
call msg('SENL corruption error - Spam filter',' ',' ',' ')
end
End
End
```

SUBROUTINES DATESET(YYYY,MM,DD) and for good measure, TIMESET(HH,MM,SS) are useful if you want to write a global test feature for inclusion in your AREV/REVG payroll/HR/accounting/manufacturing/whatever application to test for Y2K.

But let's suppose you have the Y2K firmware problem, how the hell are you going to fix it? After all, a crook BIOS can't be patched, right? Well, fortunately DOS only has two time and two date functions (one to set and one to get), so if you were to run a TSR which intercepted all such calls, and accepted and fed back the correct year when asked for, then all would be well, at least on a temporary basis. Then there's the problem that every time you cross out 1900, and replace it with 2000, the DOS dir /OD command would store in the FAT the wrong value anyway. File writes and creates would need to be intercepted. Backups by date would fail, and goodness knows what else in the operating system would be faulty, what utilities will fail, and what your next restore will actually do.....

Well, just to wish you Happy New Year, ol' Eric will source a couple of lousy machines over the next few weeks, and next issue, attempt to produce the TSR which will give you the chance to share a hangover on Monday 3rd January, 2000. Don't bother calling - Australia is ahead of most places (except New Zealand, and no-one's told them yet) so Eric will have celebrated long before most of you pop the first cork....and if it is the second coming, I'll email you, because after all, it's probably in the Book of Revelation....

## Peripheral Trivia

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

- TV : TeleTubbies
- Book:The Moral Intelligence of Children - Robert Coles
- CD: "A Young Person's Guide To Compact" - The Compact Organisation
- WEB: www.hotbot.com

Join us :

Send Mail to [Admin@Sprezzatura.com](mailto:Admin@Sprezzatura.com) with subject SUBSCRIBE SENL or complete our [online registration form](#).



# SPREZZATURA

Leave Us:	Send Mail to <a href="mailto:Admin@Sprezzatura.com">Admin@Sprezzatura.com</a> with subject UNSUBSCRIBE SENL
Change of Address:	Leave at the old address & join at the new one
Web Info:	<a href="http://www.sprezzatura.com/">http://www.sprezzatura.com/</a>
Tell us what you'd like to see in S/ENL:	<a href="mailto:info@sprezzatura.com">info@sprezzatura.com</a>

## 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 2 Issue 1, 7<sup>th</sup> January 1998.

Please encourage your correspondents to send e-mail to [admin@sprezzatura.com](mailto: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.