

SPREZZATURA

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

Tel: 020 8912 1010 Fax: 020 8912 1011

info@sprezzatura.com

SENL

July 2007

Sprezzatura's Electronic Newsletter

For Revelation developers by Revelation developers

MAKING DATABASES HAPPEN

SPREZZATURA

Contents

- Welcome 4
- XML Gateway 4
- Pest Control..... 4
- Open Engine Manager 4
- S/Web 2.0 4
- S/List for U2 5
- S/Encrypt..... 5
- S/Dump 5
- Namespaces 6
- Other business 6
- RTP65 - Making a Hash of It 7
- The Problem..... 7
- The Solution 9
- RTP65 10
- RTP65 Methods..... 11
- RTP65 CreateTable (0) 11
- RTP65 OpenTable (1) 12
- RTP65 ClearTable (3)..... 13
- RTP65 WriteRecord (5) 15
- RTP65 Equates 18
- RTP65 limitations 20
- Revelation EMEA – Annual Seminar – 27th June 2007 21
- Keynote Presentation 21
- Revelation Roadmap..... 21
- Windows Controls and OpenInsight 22
- Drag and Drop – an insider’s view 22
- Lunch..... 23
- Linear Rehashed..... 23
- Web 2.0..... 24
- Closing remarks..... 25

SPREZZATURA

Contents continued

S/Dump – Revisited - Andrew McAuley	26
Overview of the Initial User Interface.....	26
Navigation	32
The Edit Menu.....	34
The Options Menu	34
Advanced Universal Driver Heavy Configuration	40
LH Server Manager Window	40
Registry Entries Used By UDH	43
Email on Error Information	44
Unlocking Other Station’s Locks	45
Peripheral Trivia.....	47
COPYRIGHT NOTICE	47



SPREZZATURA

Welcome to this issue of SENL!

Have you ever wished you could clone your senior members of staff? We here at Sprezz take pride in knowing that our developers are “La crème de la crème” but sometimes that can be a limiting factor! We’ve got SO MANY projects going on internally and so little time to tell you all about them. If only we could clone Carl and Aaron we’d be happy (no point in cloning Andrew – he spends so much time pulling things together that the standard joke is that “Outlook 2007” is his new IDE!)

Still at least these little editorials give us the chance to at least mention fun stuff we’re up to!

XML Gateway

Our XML Gateway project with one of our favourite clients – Social Software – continues to grow in leaps and bounds. This is a truly flexible XML Production/Consumption tool that allows an OI application to produce/consume pretty much any schema you throw at it, and if it can’t cope automatically then we provide user hooks to finesse the data. One day just as a challenge we’ll have to create a schema to produce all the tables in an application in one big XML document!

Pest Control

Our tablet Pest Control app goes from strength to strength. This is a “bolt on” for users of a legacy pest control application to enable their field service personnel to receive routing instructions on a tablet PC, enter job details, obtain client sign off, print the invoice and transmit the invoices back to HQ! It’s going so well that one of our clients just won “Best Pest Control Company” in their prestigious city magazine.

Open Engine Manager

One of our long awaited development projects has finally reached fulfilment. As a lot of you will know, we had to devote so much of our internal resource to working with Revelation on the release of 8.0 that we had to neglect our own projects. Now that 8.0 is out of the door (see the review later in this issue) we can start to provide shoes for the cobbler’s children! We now have our fully functional “OEM” product – the Open Engine Manager product. This is to form the basis of a new range of Enterprise tools we’re working on even as I type. The first successful commercial deployment of this technology is in a hub based gateway system for a large Government project in the UK. At the moment we don’t have marketing materials on OEM so suffice to say it’s like OECGI2 on steroids. It’s a load balancing, self healing, and fully flexible engine management tool.

S/Web 2.0

The next product to be based on it will likely be S/Web 2.0. S/Web has enjoyed considerable success as the only truly scalable web architecture for Rev users out there. From its gestation to now, S/Web has amazed even its author’s by how powerful it is. For the uninitiated, S/Web is a tool that facilitates the production of Enterprise web sites. It provides a framework within which interaction with the web is done via a series of easily to manipulate objects. It further offers full session management facilities and easy implementation of “basic authentication”. Essentially it abstracts

SPREZZATURA

you from the complexities of web programming and allows you to get on with designing your web application without worrying about interacting with the HTTP elements at a low level. The next generation of S/Web based upon OEM looks set to blow S/Web 1.0 away!

In S/Web v2 engines are created and controlled via the ISAPI DLL directly (via the "engine farm" component), so there is no "gateway" application like there was in v1 so there is nothing to go "offline" – it runs as a service. You can however set a timeout value so that you can detect if an engine has hung and it will be terminated and replaced if necessary. This self-healing capability is fundamental to the durability required of a 24/7 web based application

In respect of load-balancing each request is queued and passed to the first available engine. If an engine is not available and the system is not maxed-out an engine will be created to handle the request thus increasing the capacity. You also have the option to retire idle engines after a timeout period to free up resources.

Normally the ISAPI DLL creates the engines locally (out of process to protect the web-server and allow termination) but there is also the option to pass requests onto separate "OE Farm" Services running on remote servers via TCPIP. These can be used exclusively or as a "backup" if the local engine capacity is full, allowing you to spread the load around your network if you wish.

In addition with S/Web 2 we plan to have a desktop UDP/TCP server application that the S/web system can interact with for real time event notifications. You simply tell the S/Web DLL to send notifications to the desktop machine.

S/List for U2

With the increasing number of U2 users we've been motivated to create a special version of S/List specifically for U2 users. This is a self contained application which can be distributed to existing U2 users allowing their client base to create reports quickly and simply. The new version comes with a new pricing model as it deploys on Development copies of OpenInsight – the details will be available on our web site shortly.

S/Encrypt

A product we've been using ourselves for a year or so now providing *commercial-grade* encryption capability for OpenInsight applications (OpenInsight's ENCRYPT_FORMAT uses an algorithm which is widely viewed as not being suitable for new applications.). S/Encrypt provides multiple encryption algorithms to satisfy the most demanding of clients (in our case security aware financial firms and "sensitive" Government Agencies). We've had the documentation in beta for a while but as always development has outstripped deployment ☺. We'll try and get some onto the web site in the next month or so. In the meantime if you app needs COMMERCIAL-GRADE encryption then drop us a line.

S/Dump

In the last issue we previewed S/Dump and promised beta releases to those who cared to try it. Regretfully our internal alphas were brutal – we're not very forgiving testers and it has undergone a

SPREZZATURA

substantial revamp. So much so that we're RE-reviewing it in this issue and repeating the call for beta testers.

Namespaces

Frequently when we're demonstrating our coding techniques to User Groups and Seminars we're asked why we have such apparently narcissistic naming conventions for our routines, ZZ_ this ZZX_ that. The answer is simple – NameSpaces. When you need to identify anything as belonging to you then you need to create a unique namespace for yourself or there's always the chance that someone else will get there first. Remember when you were in school as a kid and your mum used to sew your name inside your clothes to make sure no one went home with your gear? Well it's the same thing here.

Several of our users have been left wishing they'd sewn their names in to their routines with recent OpenInsight releases. First Editor+ was released with a controlling program of "Editor". Then another conflict was created with another client and a routine called RETSTACK and finally a user struggled for days wondering why his images were behaving strangely when he had one called SEARCH2. All of these were overwritten by Rev routines – and let's face it, it's Rev's environment so they can name as they like.

So a word of advice – if you're developing your own systems then don't forget namespaces! You may live to regret it otherwise.

Other business

This month a senior Sprezzie toured the East Coast chatting to our clients about AREV32 and OI8. This seems like an ideal time to remind you that we're always available for consultancy work – both esoteric and mundane!

Revelation EMEA had an all day seminar recently – we've included a short review of this later in the issue. A good time was had by all and it was great to see so many European countries represented! One of the most interesting presentations was that on security issues facing Web Developers – an area we happen to have quite a lot of experience of courtesy of some of our more demanding clients. The presentation made for quite an unnerving experience if you've considered OI apps safe by virtue of their non-mainstream nature!

Enjoy this SENL we truly hope that you'll find something in it that is of value to you. If there's anything you'd like us to consider for inclusion then feel free to drop us a line!

Regards

Your friends at Sprezz

SPREZZATURA

RTP65 - Making a Hash of It

So, we all know about “hashing” in OI right? After all it’s the principal technology that Revelation databases are built upon – the infamous Linear Hash filing system. Well, what you might not know is that hidden away in the depths of the system is another hashed filing system, but this time it’s an in-memory hash table called the “Fast Filing System” (FFS) and exposed via the system function RTP65.

To illustrate why this might be of interest to you let’s look at an example where you might consider using it.

The Problem

We’re all used to working with dynamic arrays in our Basic+ programs, which are of course just strings with delimiter characters that the system functions recognize. This is a very powerful and easy construct to use for most of our everyday requirements, but this technique can suffer performance problems when used with large sets of data.

For example, say we need to cache 1000+ items of data in memory for subsequent fast access. We can load it into a dynamic array using code similar to the following:

```
$insert logical

common /%%_MEM_CACHE_%%/ idList@, dataList@

idList@ = ""
dataList@ = ""
eof = FALSE$

select hTable ; * // select the records - assume the
                ; * // table has 1000+

loop
  readNext id else eof = TRUE$
until eof
  read rec from hTable, id then
    idList@ := id : @fm
    dataList@ := rec<1> : @fm
  end else
    * // Panic
  end

repeat

idList@[-1,1] = ""
dataList@[-1,1] = ""
```


SPREZZATURA

And then when we need the data we could do something like this:

```
common /%%_MEM_CACHE_%%/ idList@, dataList@

locate id in idList@ using @fm setting pos then
  data = dataList@<pos>
end
```

The big problem with this that our access to the data is sequential – we have to start looking at the beginning of the ID list to find the position of our data, and then the <> operators have to start at the beginning of the data list to extract it. In processing terms this could be quite expensive.

What about dimensioned arrays? Alright, perhaps we could do something like this:

```
$insert logical
common /%%_MEM_CACHE_%%/ idList@, dataArray@(2048)

idList@ = ""
matParse "" into dataArray@

ctr      = 0
eof      = FALSE$

select hTable ; * // select the records - assume the
              ; * // table has 1000+
loop
  readNext id else eof = TRUE$
until eof
  read rec from hTable, id then
    ctr      += 1
    idList@  := id      : @fm
    dataArray@(ctr) = rec<1>
  end else
    * // Panic
  end
repeat
```


SPREZZATURA

```
idList@[-1,1] = ""
```

And to access:

```
common /%%_MEM_CACHE_%%/ idList@, dataArray@(2048)

locate id in idList@ using @fm setting pos then

    data = dataArray@( pos )

end
```

In this case we've removed the need for the second sequential access with the <> operators but we've still got the initial sequential ID lookup and also we're forced to fix the size of the dimensioned array when we create it so we can't grow it at a later point of we wish.

The Solution

What we need is full random access, a requirement which many other languages solve by using an in-memory hash table. For those of you new to hash-tables, Wikipedia defines them as:

"a data structure that associates keys with values. The primary operation it supports efficiently is a lookup: given a key (e.g. a person's name), find the corresponding value (e.g. that person's telephone number). It works by transforming the key using a hash function into a hash, a number that is used to index into an array to locate the desired location ("bucket") where the values should be.

Hash tables support the efficient addition of new entries, and the time spent searching for the required data the lookup time is independent of the number of items stored (i.e. O(1).)"

Sound familiar? It's basically what our Linear Hash filing system does to find records in tables except that its operations are disk-based rather than RAM-based (and there are a few other bells and whistles too). However, implementing this sort of data structure usually involves using pointers and low-level memory allocation routines so it's not something that you would really want to do with Basic+ (You *could* still emulate it using dynamic/and or dimension arrays if you want, but you'd still be taking a big performance hit)

SPREZZATURA

The good news is that OpenInsight actually does implement this sort of data structure at a low level and provides an interface for Basic+ routines to use it. Meet the RTP65 function...

RTP65

The RTP65 function is implemented in the aptly named RTP65.dll module via an exported function called FFSMAIN. This is aliased as RTP65() which we can see by looking at the following DLL prototype record supplied with the OI system (DLL_RTP65 in the SYSPROCS table):

```
RTP65
```

```
VOID OECALL FFSMAIN(VOID) AS RTP65
```

As you can see it's using the OECALL calling convention as opposed to the usual STDCALL convention so it tells us nothing about the arguments that should be passed. When you see this sort of definition it means that the function is "OE-aware" and has the capability to pull the parameters right off the internal OE stack.

In this case the only real way to find out the interface is via trial and error, and so without further ado here it is:

RTP65(Method, Handle, Key, Record, Reserved, Status)

Parameter	Description
Method	<p>Defines which RTP65 method to perform. Can be one of the following values:</p> <ul style="list-style-type: none"> 0 – Create Table 1 – Open Table 2 – Close Table 3 – Clear Table 4 – Read Record 5 – Write Record 6 – Delete Record <p>We suggest that you create a set of equates for these values (See below)</p>
Handle	Handle to the hash table. Depends on the Method parameter.
Key	Record Key. Depends on the Method parameter.
Record	Record Data. Depends on the Method parameter.

SPREZZATURA

Reserved	Do not use - Pass null.
Status	Returns the status of the requested RTP65 method. Depends on the Method parameter.

RTP65 Methods

The RTP65 function exposes several Methods as listed above:

RTP65 CreateTable (0)

This method creates an RTP65 table. You must use this method to create an in-memory hash table before you can use it.

Parameter	Description
Method	"0"
Handle	Not used
Key	Name of the RTP65 table to create
Record	Not Used
Reserved	Not Used
Status	Returned status flag. Can be one of the following values: 0 – No error 1 – Table already exists 5 – Maximum number of tables exceeded

e.g.

```

declare function rtp65
tableID = "TEST_RTP65"
htStatus = 0
call rtp65( 0, "", tableID, "", "", htStatus )

```


SPREZZATURA

RTP65 OpenTable (1)

This method opens a previously created RTP65 table. You must use this method to obtain a handle to the table before you can use any of the other RTP65 record methods on it.

Parameter	Description
Method	"1"
Handle	Returns the handle of the opened RTP65 table
Key	Name of the table to open. This table must have been created previously as described above.
Record	Not Used
Reserved	Not Used
Status	Returned status flag. Can be one of the following values: 0 – No error 2 – Table does not exist

e.g.

```

declare function rtp65
tableID   = "TEST_RTP65"

hTable    = 0

htStatus  = 0

call rtp65( 1, hTable, tableID, "", "", htStatus )

```


SPREZZATURA

RTP65 CloseTable (2)

This method closes a previously created RTP65 table. All data is destroyed and the table must be re-created before it can be used again. Any open handles on the table will be invalidated.

Parameter	Description
Method	"2"
Handle	Pass in the handle of an opened RTP65 table
Key	Not Used
Record	Not Used
Reserved	Not Used
Status	Not Used

e.g.

```
declare function rtp65
hTable    = 0
call rtp65( 2, hTable, "", "", "", "" )
```

RTP65 ClearTable (3)

This method simply deletes all data from a previously opened RTP65 table. Any open handles on the table will be still be valid.

Parameter	Description
Method	"3"
Handle	Pass in the handle of an opened RTP65 table
Key	Not Used
Record	Not Used
Reserved	Not Used

SPREZZATURA

Status	Not Used
--------	----------

e.g.

```
declare function rtp65
hTable    = 0
call rtp65( 3, hTable, "", "", "", "" )
```

RTP65 ReadRecord (4)

This method reads a record from an open RTP65 table.

Parameter	Description
Method	"4"
Handle	Pass in the handle of an opened RTP65 table
Key	Pass in the key of the record to read
Record	Returns the record stored in the table.
Reserved	Not Used
Status	Returned status flag. Can be one of the following values: 0 – No error 3 – Bad Table Handle 4 – Table has not been opened or Record does not exist

e.g.

```
declare function rtp65
```


SPREZZATURA

```

* // Open the RTP65 table

tableID = "TEST_RTP65"

hTable = 0

htStatus = 0

call rtp65( 1, hTable, tableID, "", "", htStatus )

* // ... do some processing ...

id = "X2345"

rec = ""

call rtp65( 4, hTable, id, rec, "", htStatus )

```

RTP65 WriteRecord (5)

This method writes a record to an open RTP65 table.

Parameter	Description
Method	"5"
Handle	Pass in the handle of an opened RTP65 table
Key	Pass in the key of the record to read
Record	Pass in the record to store in the table.
Reserved	Not Used
Status	Returned status flag. Can be one of the following values: 0 – No error 3 – Bad Table Handle 4 – Table has not been opened 6 – Memory Error

SPREZZATURA

e.g.

```

declare function rtp65

* // Open the RTP65 table

tableID = "TEST_RTP65"

hTable = 0

htStatus = 0

call rtp65( 1, hTable, tableID, "", "", htStatus )

* // ... do some processing ...

id = "X2345"

rec = "Yadda" : @fm : "Yadda"

* // ... write a record ...

call rtp65( 5, hTable, id, rec, "", htStatus )

```

RTP65 DeleteRecord (6)

This method deletes a record from an open RTP65 table.

Parameter	Description
Method	"6"
Handle	Pass in the handle of an opened RTP65 table
Key	Pass in the key of the record to delete

SPREZZATURA

Record	Not Used
Reserved	Not Used
Status	<p>Returned status flag. Can be one of the following values:</p> <p>0 – No error</p> <p>3 – Bad Table Handle</p> <p>4 – Table has not been opened or Record does not exist</p>

e.g.

```

declare function rtp65

* // Open the RTP65 table

tableID = "TEST_RTP65"
hTable = 0
htStatus = 0

call rtp65( 1, hTable, tableID, "", "", htStatus )

* // ... do some processing ...

id = "X2345"
rec = "Yadda" : @fm : "Yadda"

* // ... delete a record ...

call rtp65( 6, hTable, id, "", "", htStatus )

```


SPREZZATURA

RTP65 Equates

Before we go any further it would probably be handy to create some equates based on the RTP65 interface to make our code a little clearer:

```
compile insert rtp65_Equates

declare subroutine rtp65

* // Method equates

equ RTP65_CREATE$           to 0
equ RTP65_OPEN$            to 1
equ RTP65_CLOSE$           to 2
equ RTP65_CLEAR$           to 3
equ RTP65_READ$            to 4
equ RTP65_WRITE$           to 5
equ RTP65_DELETE$          to 6

* // Status equates

equ RTP65_NO_ERR$           to 0
equ RTP65_TABLE_EXIST_ERR$ to 1
equ RTP65_TABLE_NOT_EXIST_ERR$ to 2
equ RTP65_ERR_BAD_HANDLE$  to 3
equ RTP65_REC_DNE_ERR$     to 4
equ RTP65_TABLE_NO_OPEN$   to 4
equ RTP65_UNKNOWN_CODE$    to 5
equ RTP65_ERR_NO_MEM$      to 6
```


SPREZZATURA

So looking back to our original problem an RTP65 implementation might look something like this (error checking has been omitted for clarity):

```
$insert rtp65_Equates

$insert logical

tableID = "TEST_RTP65"

hTable = ""

htStatus = 0

* // Create the RTP65 table

call rtp65( RTP65_CREATE$, "", tableID, "", "", htStatus )

* // Then open it ...

call rtp65( RTP65_OPEN$, hTable, tableID, "", "", htStatus )

select hLHTable ; * // select the records - assume the
                ; * // table has 1000+

loop

  readNext id else eof = TRUE$

until eof

  read rec from hLHTable, id then

    call rtp65( RTP65_WRITE$, hTable, id, rec, "", htStatus )

  end else

    * // Panic

  end

repeat
```


SPREZZATURA

And then when we need the data:

```
$insert rtp65_Equates
$insert logical

tableID = "TEST_RTP65"
hTable = ""
htStatus = 0

* // Open the RTP65 table

call rtp65( RTP65_OPEN$, hTable, tableID, "", "", htStatus )

id = "X2345"
rec = ""

call rtp65( RTP65_READ$, hTable, id, rec, "", htStatus )
```

As with any scheme based on hashing algorithms, it would be wise to run tests on your own data before deciding to go down this route, as ultimately the structure of the keys you use will decide the performance.

RTP65 limitations

During the course of writing this article we did notice one obvious limitation with RTP65. As of OI8.0 we are limited to a maximum of 10 hash tables in total. Bear in mind that the system itself uses RTP65 tables in several places (most notably it implements the OI event stack) so you won't be able to use all 10 "slots". You should be able to get away with about 6 or 7 if you need that many - if you need more then we suggest you let Revelation know for future product enhancements.

SPREZZATURA

Revelation EMEA – Annual Seminar – 27th June 2007

As part of their continued efforts to help their EMEA user base stay current with breaking technical trends with the OpenInsight product range, Revelation Software held a conference on Wednesday 27th June at the Chiswick Moran Hotel – a new luxury hotel in the West of London ideally suited for delegates arriving by road air or rail.

The day opened with pastries and coffees and general mingling – Sprezz's delegates included Andrew, Aaron and Carl whilst Revelation were represented by Bill (from Tech Support), Martyn (Master of Ceremonies) and Mike Ruane (CEO of Revelation Software). Delegates from nearly 10 EMEA countries were also present to feast on the wealth of technical information on offer and to network with their fellow developers.

Keynote Presentation

The first presentation was a tag team effort by Ailsa McAlees and Claude Masseron of IBM's U2 division. They outlined the circumstances by which IBM came to be the biggest Multivalue vendor in terms of turnover and explained the reasons for their strategic alliance with Revelation Software. Ailsa underlined the close contacts between Revelation and IBM by referring to her own experience working with Revelation back in the AREV days and by further referring to the close relationship built up since Lotus Notes days.



Revelation Roadmap

Next up came Mike Ruane outlining the changes that had gone into the product for 8.0. He gave an overview of the many new features including the Banded Report Writer, the use of Drag and Drop, improved System Editor features and the many enhancements that have been made to TCL.



He further made the point that this release fixed nearly 100 bugs in the product whilst implementing over 60 user requested features. Needless to say he encouraged all present to make their own suggestions for improvements in the product and, true to form, as suggestions came in he made a note of them – announcing later that several of the things requested had ALREADY made it into 8.01 – taking responsiveness to a new level!

The crowd were thrilled to see AREV32 in action – the idea that a manufacturer would go to such lengths to ensure that people running legacy apps can continue to leverage return from their decade old investment is truly unique in the PC industry!

MAKING DATABASES HAPPEN

SPREZZATURA

Mike finalised with a live demonstration of AREV32 in action – taking an existing AREV app and moving it to ARE32 - ably assisted by Martyn who showed the use of GUI elements from within an AREV32 application before the session was broken for refreshments.

Windows Controls and OpenInsight

Carl Pates then reprised his very successful Seattle presentation on the use of Windows Common Controls within OpenInsight. As outlined in the last SENL he explained that since the release of OI 7.21 it has been possible to include Windows Common Controls such as status lines. Progress bars, Extended Combo Boxes, Tree Views, Tool Bars *et al.*

Then in Carl's inimitable way he took the audience through an explanation of the steps required to implement these controls with OI - ensuring that at every step those listening understood not just WHAT to do but also WHY they had to do it. As an aside he also took the opportunity to provide tangible examples of why respecting namespace conventions and wrapping commonly used functionality in easy to understand "objects" massively increase productivity whilst protecting the developer from subsequent product changes.

After several practical examples (which were all provided on the conference CD along with source code) he went on to examine the changes in 8.0 which allow synchronous responses to Windows messages and explained the impact such an ability can have on OI development. This was illustrated by using synchronous responses to Windows messages to prevent a double click from operating as such on a hierarchical listbox – making it respond in a much more recognisable way for modern users.

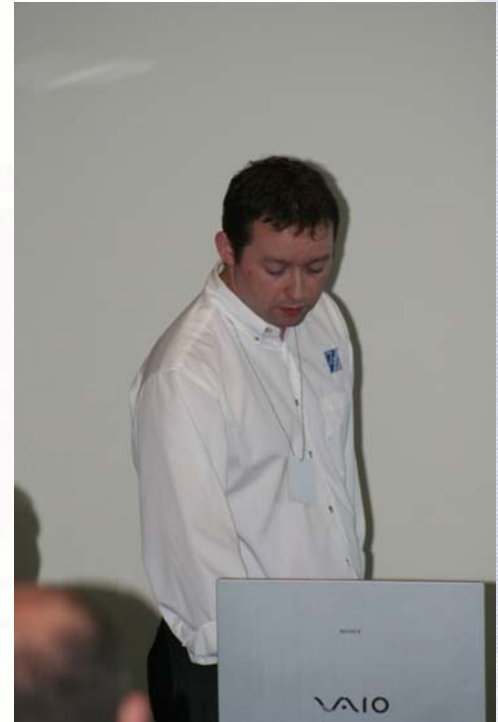


At Seattle the software had been Alpha and thus some features had to be worked around. In the shipping product such things were no longer an issue.

This presentation was so avidly received that the decision was taken to allow it to encroach into the time allocated for the next presentation, namely :-

Drag and Drop – an insider's view

Again invoking Seattle, Andrew's presentation was only slightly different than his original presentation.



SPREZZATURA

In the first instance Andrew explained how simple “Drop Files” functionality worked – this is the form of Drag and Drop that has been in Windows since 3.1. Starting simple, he showed dragging bitmaps onto OI forms and off OI forms before making the situation more complex by allowing file lists to be dragged into Edit Tables. This provided him the excuse to show off some undocumented calls designed to make the OI programmer’s life easier. These features were added after the Seattle show in response to developer requests for simplifying some processes.

After simple drag and drop Andrew moved onto more complex OLE drag and drop and showed text being dragged from Word docs into OI Edit controls and from OI screens into external Window applications.

Finally Andrew gave those present some feel for just how granular developers could make OI’s control over drag and drop by showing how to STOP specific data types being dragged into specific controls and further how to manipulate the data before it was dropped to change the contents of the data to be dropped.

Lunch

At this point the conference broke for lunch and the ensemble tramped down to the Napa Grill for a tasty three course lunch. The food was well presented and the service was excellent. The only downer was having to choose only one starter/main course/dessert!

After lunch the sessions reopened with

Linear Rehashed

Aaron spent a little time ensuring that everyone was acquainted with filing system fundamentals – exactly how hashing worked, how overflow worked – the pace was gentle – not unlike the way a roller coaster gently inches its way towards the apex of a particularly vertiginous drop...

From hereon in it got pretty technical. The exact structure of LH frames, how overflow actually works and more complexly – how resizing works! Perhaps one of the most interesting assertions was that the new file architecture with the UD3 allows a maximum file size of 16 **exabytes** and as Aaron pointed out Wikipedia estimated that the TOTAL human media output to date globally is around 12 exabytes – meaning we’re not going to run out of space any time real soon!

In conclusion Aaron demonstrated Sprezzatura’s new S/Dump facility as a way to show how this information all fitted together from within OpenInsight. This preview of the new S/Dump interface was well received by the audience. Since the last S/ENL this interface has undergone a fairly radical re-tweaking.

After the talk there were several questions from the audience –more so than in any other presentation and Aaron fielded these skilfully before handing over to :-



SPREZZATURA

Web 2.0

This presentation – which was a late addition primarily added at user demand – was a whirlwind tour through the addition of OECGI2.EXE to the product through to an overview of just some of the security issues that fact OI Web developers.

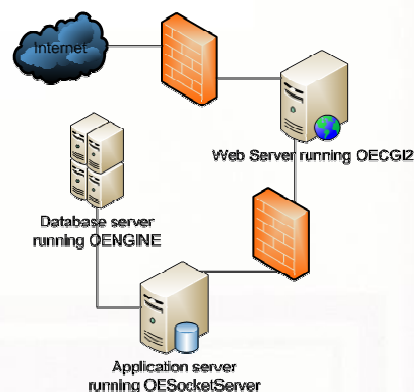
The first part of the session was handled by Andrew who provided an overview of why OECGI2 had been provided and what extra facilities it offered to those who were previously using OECGI. This first necessitated a discussion of what the OESocketServer technology actually constituted and why this opened the product for external access to all components understanding the the JD3 protocol.

Subsequently there were a number of practical demonstrations of using REGEDIT to configure OECGI2 correctly, of setting up IIS to utilise OECGI2 and of accessing multiple applications from one Web Server utilising OECGI2, Virtual Sub-directories and sub keys in the registry. In addition a practical illustration of using customised error pages when unable to contact the OESocket server was provided.



Fascinating as this was it was as naught compared to Carl's mastery of the second half of the session. Starting out from the basic premise that ALL web applications are inherently unsafe at the hands of determined hackers Carl explained why we, as OI developers have nothing to be complacent about. Easy as it is to fall into the trap of believing that only mainstream databases can be hacked Carl set out in easily understood terms how OI Web based apps are just as open as any others to

- Information leakage – providing useful information to potential hackers
 - At each stage Carl explained potential areas of leakage whilst identifying best practice to deal with it.
- Input Validation Bypass
 - By showing how easily hackers can amend HTTP streams Carl was able to show how relying on Client Side validation is a recipe for disaster. Some basic "safe" rules were outlined but the nature of the presentation was such that time was too tight to provide detailed guidelines
- Cross Site Scripting
 - This flaw which is present in any unprotected site that allows free text input then republishes this input on the web provided an ideal opportunity to discuss just how easy it is to allow hackers a portal to hijack other users' machines. Carl pointed out how apparently safe ways of looking for dangerous strings ARE NOT SAFE by reference to several examples.
- R/List injection vulnerabilities



SPREZZATURA

- Here at least there was SOME good news. Whilst these vulnerabilities can allow a denial of service attack, they cannot allow corruption of databases – unless you are running a backend that allows the creation of in line formulae in a query statement.
- ...

Carl went on to discuss many other vulnerabilities than an OpenInsight based web app might be prone to. The consensus was that perhaps being complacent about the security of OI based web apps was a little naive!

Closing remarks

Martyn closed the conference by thanking all the delegates for their attendance and by reminding them that next year saw the return of the International Revelation Conference to Vegas – full details at <http://www.revelationconference.com>.

The remaining delegates who chose to do so then departed in taxis to the Fullers Brewery in Chiswick to see how London's most popular beer is brewed, followed by eating and drinking in the adjoining pub!

All in all the day was a stunning success. All of the delegates spoken to felt that the day had been well worth the money and that the international conference seemed more attractive as a result.

We look forward to seeing even more people at the next EMEA event!

SPREZZATURA

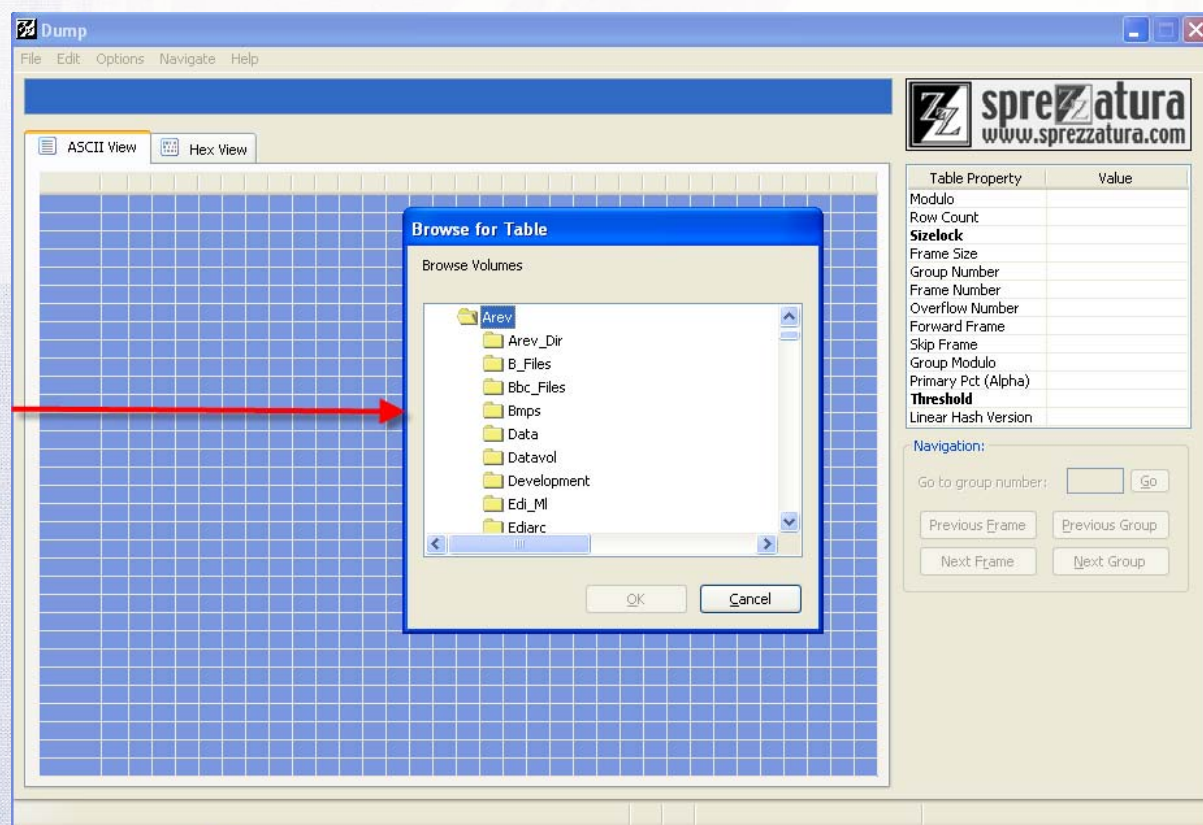
S/Dump – Revisited - Andrew McAuley

As hinted at in the Editorial the release of S/Dump was delayed as we tested and re-tested internally. One of the issues that we considered needed to be addressed was one of UI. We ultimately took the decision that we were allowing form to triumph over substance. Now whilst this seems to be almost *de rigeur* in a Windows environment we couldn't bring ourselves to ship something which looked fantastic but which lacked needed functionality. So it was back to the drawing board for the intrepid design team and poor Aaron was forced to work through countless internal Alpha cycles.

We're pleased that we've taken this step as the result is a finely honed piece of software that we're pleased to release to our beta testers!

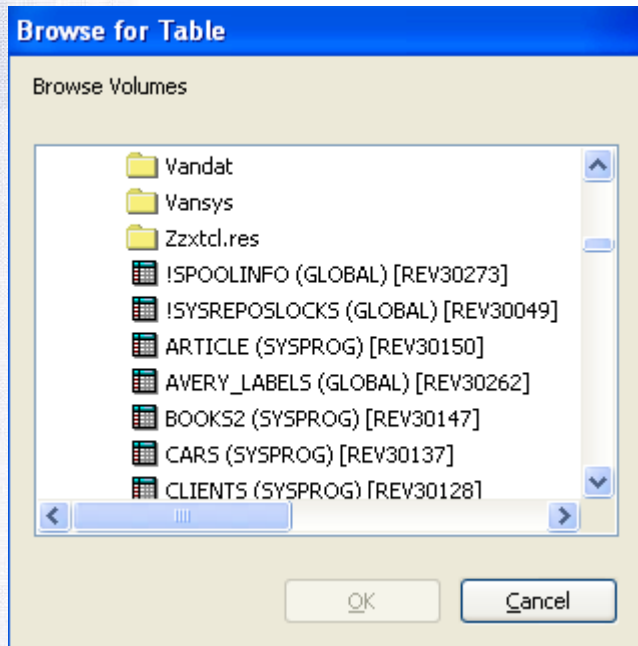
Overview of the Initial User Interface

Launching Dump (by shift clicking on ZZ_TCL_UI_DUMP or EXEC'ing it from the system monitor) displays the DUMP window and as the Window itself is pointless without a file to work on it displays Sprezzatura's "Browse for table" dialog.

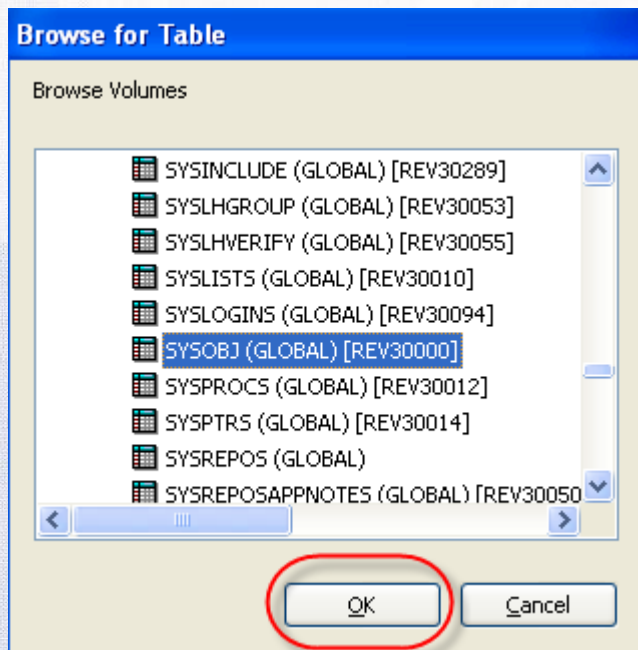


This is a little bit different than a normal "Open File" dialog as it understands OpenInsight files and lists them accordingly :-

SPREZZATURA

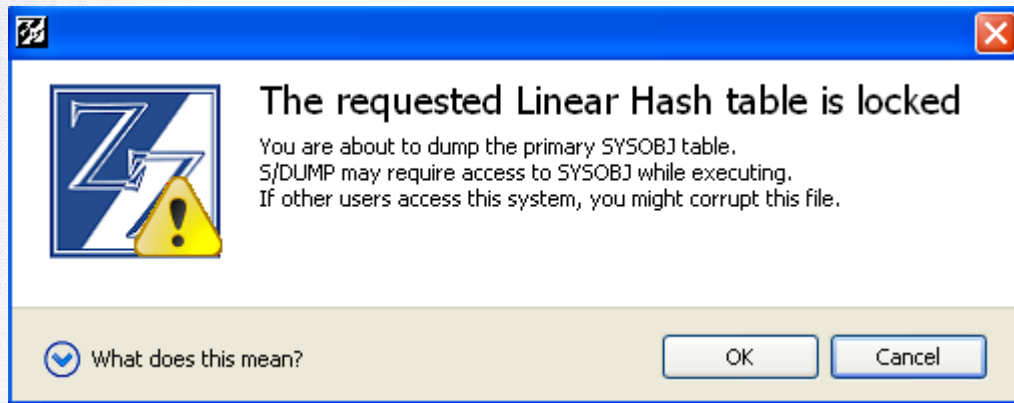


So selecting a table enables the OK button

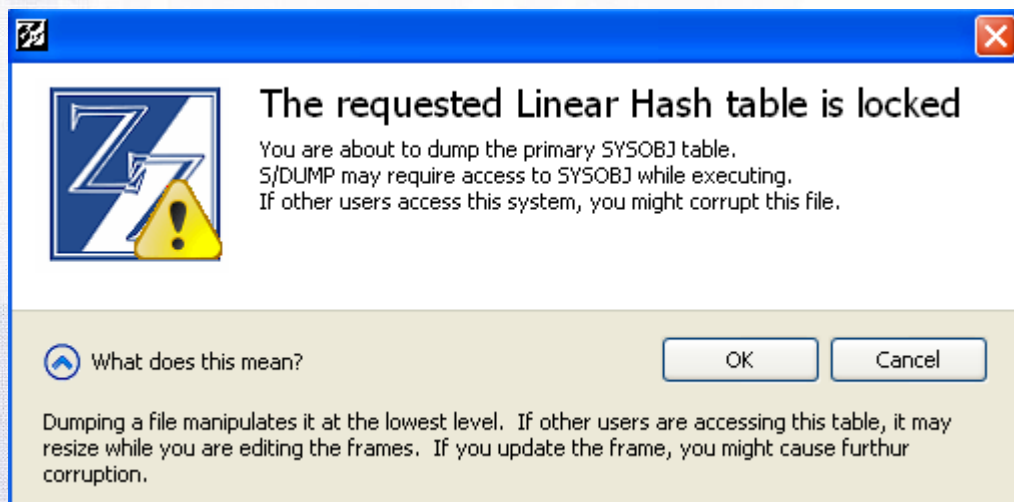


SPREZZATURA

Dumping SYSOBJ (and other system files) can lead to problems so the new “Intelligent” Dump acknowledges this and warns you accordingly :-



If that information isn't enough for you then you can always click on the down arrow to get more information :-



As we opt to continue the main S/Dump Window opens and rather than slavishly following the AREV Dump Window layout it provides a more modernised cleaner UI.

SPREZZATURA

The screenshot shows the S/Dump application window for SYSOBJ (REV30000.LK). The main area displays a hex dump with columns numbered 1 to 32. The first few lines of the dump are:

```

1  * * * * *
33  X G E T P R O P E R T Y
65  E R T Y * * O B J X A R R A Y , - 1 , 1 * * P R O P A R R A Y ,
97  - 1 , 2 * * D A T A A R R A Y , - 1 , 3 * * E R R O R A R R A Y
129 , - 1 , 4 * * O B J X P O S , - 1 , 5 * * P R O P O S , - 1 ,
161 6 * * E O F , - 1 , 7 * * T H I S O B J E C T , - 1 , 8 * * T H I
193 I S P R O P , - 1 , 9 * * P R E V P R O P , - 1 , 1 0 * * T H I
225 5 D A T A , - 1 , 1 1 * * E R R O R T X T , - 1 , 1 2 * * # #
257 # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
289 # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
321 # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
353 # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
385 # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
417 5 * * # # # # # # # # # # # # # # # # # # # # # # # # # # # #
449 * * * * * 1/2 * * : j * * s * * N * * > * * e * * # * * * * *
481 H * d * h * # * * * * * I * * u * * * * * * * * * * * M * * i * * v * * # * *
513 5 * # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
545 x * - * x * < * # * * # * * # * * # * * # * * # * * # * * # * *
577 * * t * * * * * # # # # # # # # # # # # # # # # # # # # # # # # #
609 3 * * * * * M * * * * * A * * * * * M * * * * * O * * # * * # * *
641 A P P S * E D I * P R E X L A T E * I N V * T I O L T D * # * E D I
673 E D I * P R E X L A T E * I N V * T I O L T D * # * E * * * * *
705 * * * * * # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
737 * * * * * # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
769 * * * * * # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
801 * * * * * # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
833 * * * * * # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
865 * * * * * # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
897 * * * * * # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
929 * * * * * # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
961 * * * * * # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
993 * * * * * # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
  
```

On the right side, there is a 'Table Property' panel with the following data:

Table Property	Value
Modulo	8286
Row Count	20,681
SizeLock	0
Frame Size	1024
Group Number	1
Frame Number	0
Overflow Number	0
Forward Frame	0
Skip Frame	0
Group Modulo	8286
Primary Pct (Alpha)	79.68
Threshold	80
Linear Hash Version	Standard Linear Hash

Below the table is a 'Navigation' section with buttons for 'Go to group number:', 'Previous Frame', 'Previous Group', 'Next Frame', and 'Next Group'.

The caption identifies the tale being worked on



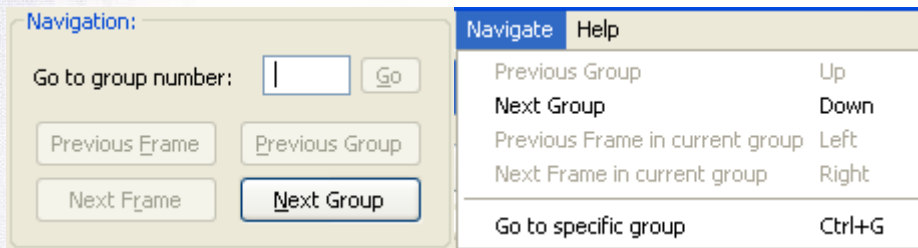
While the property panel identifies the header details, with changeable values being indicated by emboldening. Attempts to edit non-bold cells fail – non-bold cells cannot get focus.

Table Property	Value
Modulo	8286
Row Count	20,681
SizeLock	0
Frame Size	1024
Group Number	1
Frame Number	0
Overflow Number	0
Forward Frame	0
Skip Frame	0
Group Modulo	8286
Primary Pct (Alpha)	79.68
Threshold	80
Linear Hash Version	Standard Linear Hash

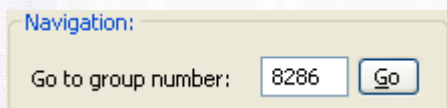
The system also identifies the type of Linear Hash file, AREV 1, standard or UD3 format.

SPREZZATURA

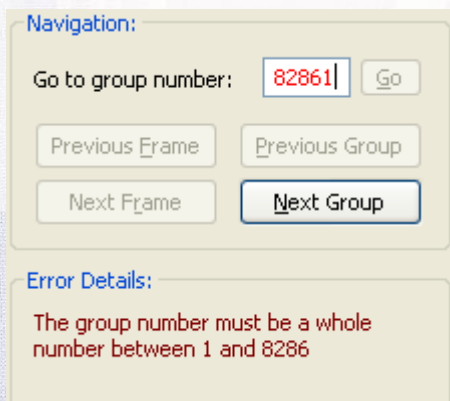
Naturally the navigation panel is intelligent and on screen and on menu only allows the selection of options that make sense in the current context. So on the very first frame of the file without overflows you can only go to the next group OR go to a specific group.



You'll note from the modulo that the highest modulo is 8286, so I can type that in to the edit line and the Go button will enable



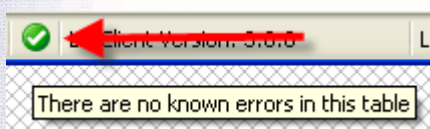
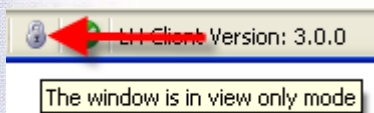
But the moment I step outside the allowable bounds the Go button is disabled and an error is displayed.



The status line displays further useful information, the file name, the LH Client version and the LH Service running.

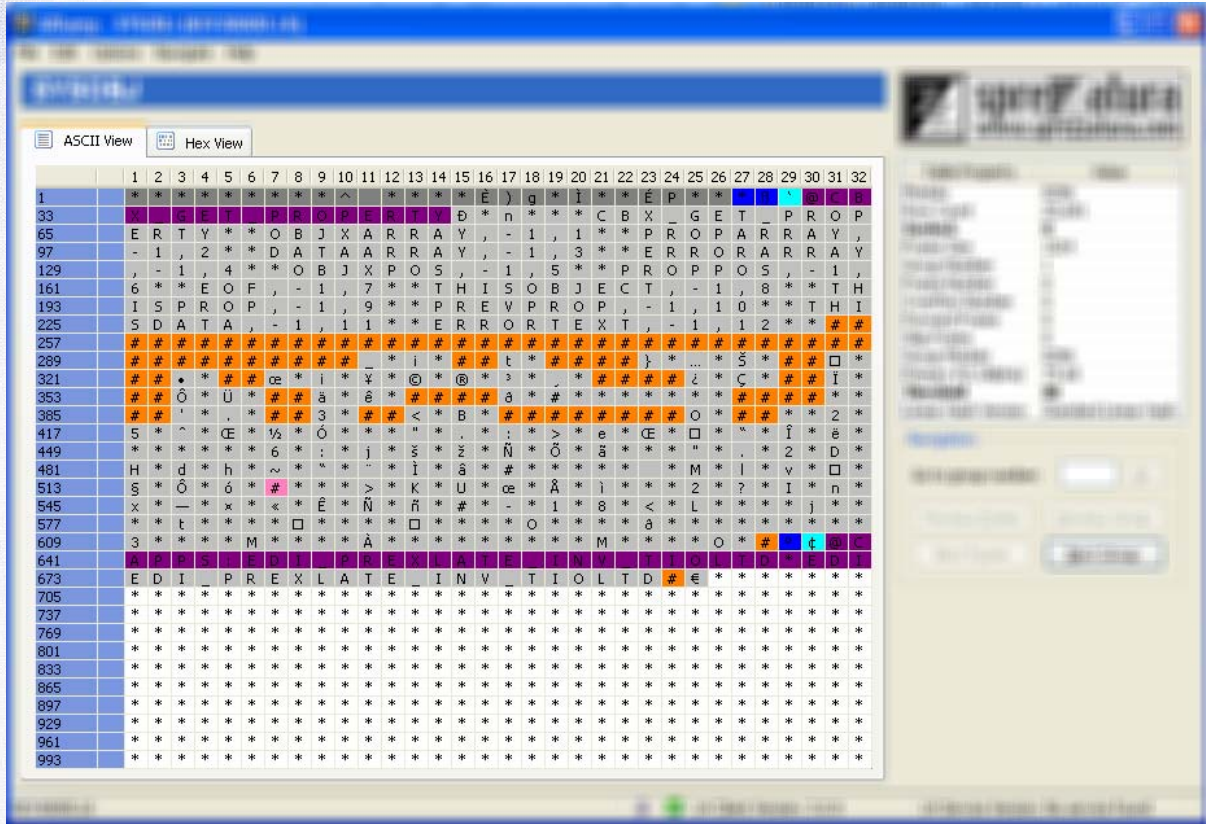


In addition there are two little icons providing further information with their very own tool tips :-

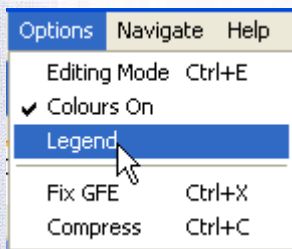


SPREZZATURA

The main body of the Window is given over to the display of the frame contents :-



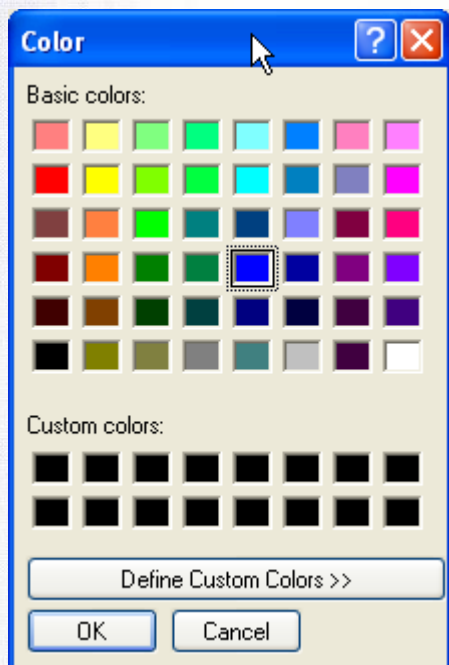
The various colours have various meanings and are user definable by accessing the Options Menu and choosing Legend



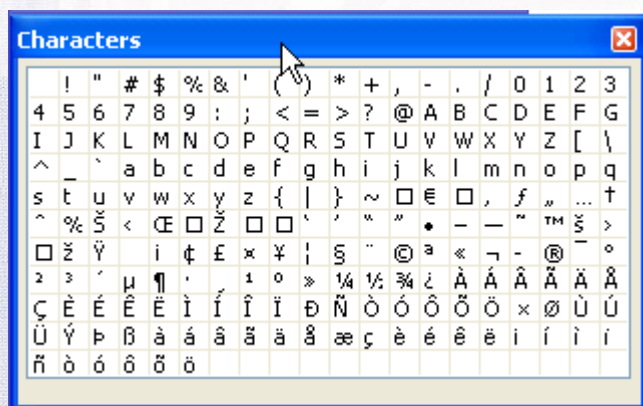
Option	Value
Character to use below space	*
Character to use above 246	#
Header colour	8421504
Row length colour	16711680
Id length colour	16776960
Id colour	8388736
Data colour	12632256
@RM colour	33023
@FM colour	7340031
@VM colour	12615935
Blank colour	16777215
Error colour	255

SPREZZATURA

The Legend dialog (in common with all other dialogs) remembers its location, so put it where it most suits you and that's where it'll be when you open it again. The options in Legend are quite self explanatory but double clicking a colour will launch a choose colour dialog



whilst double clicking a character will launch the choose character dialog.



Navigation

As you would expect navigation is possible using the keyboard or the mouse. Menu options are provided for keyboard users and buttons for mice users (although accelerator keys are provided for keyboard users on the buttons too). Groups may be navigated to directly as shown above or browsed using the appropriate choice. So from the first group of the SYSOBJ table we can navigate to the next group by pressing Alt-N. This group has overflow frames so we can navigate to the next frame. As we are on the first frame in the group we cannot navigate to the previous frame. Alternatively we could navigate to the previous or next

SPREZZATURA

group.

The screenshot shows the S/Dump application window titled "S/Dump - SYSOBJ (REV 30000.LK)". The interface includes a menu bar (File, Edit, Options, Navigate, Help) and a toolbar with "ASCII View" and "Hex View" options. The main area displays a hex dump of data, with columns numbered 1 to 32 and rows numbered 1 to 933. The data is presented in a grid format, with some characters highlighted in different colors (e.g., red, yellow, blue). On the right side, there is a "Table Property" table and a "Navigation" section.

Table Property	Value
Modulo	8286
Row Count	20,681
SizeLock	0
Frame Size	1024
Group Number	2
Frame Number	1
Overflow Number	0
Forward Frame	23077
Skip Frame	23077
Group Modulo	8194
Primary Pct (Alpha)	79.68
Threshold	80
Linear Hash Version	Standard Linear Hash

The "Navigation" section includes a "Go to group number:" field with a "Go" button, and four buttons: "Previous Frame", "Next Frame", "Previous Group", and "Next Group".

Moving to the next frame reveals a non zero overflow number and again the buttons and menu items change to remain logically consistent.

SPREZZATURA

The Edit Menu

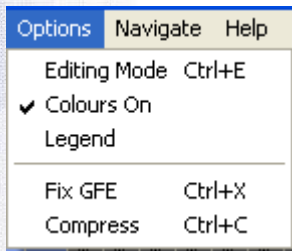
The edit menu contains three choices :-

The size lock keys are self explanatory but the Undo key is used when you have been editing a frame directly. It permits unlimited undoing of characters changed one character at a time.

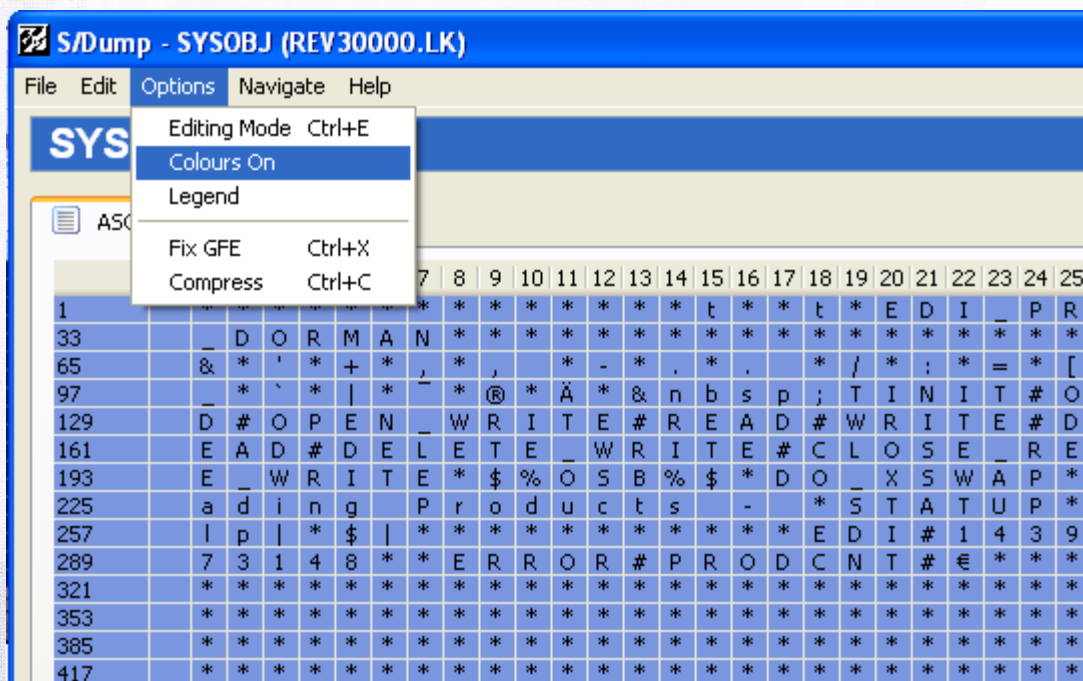
The Options Menu

The Options menu contains five choices. The last two choices (Fix GFE and Compress) simply pass control to the native OpenInsight routines to do the work. This means that while they are operating there is a dearth of feedback as to what is going on. Regretfully it made sense to reuse what already existed in what is, after all, a free tool.

SPREZZATURA



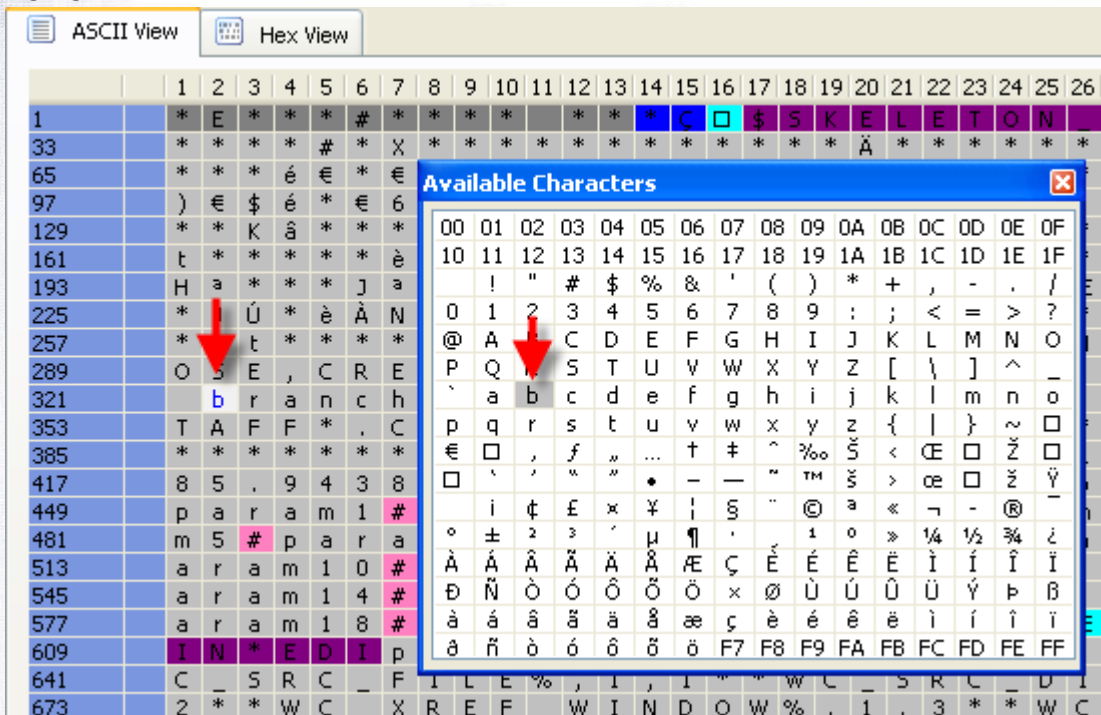
The other options are in descending order of complexity! Legend we have seen in action already. “Colours on” is a check menu item that does exactly what it says on the tin – turns colours on or off. So unsetting this check makes display much quicker but less interesting.



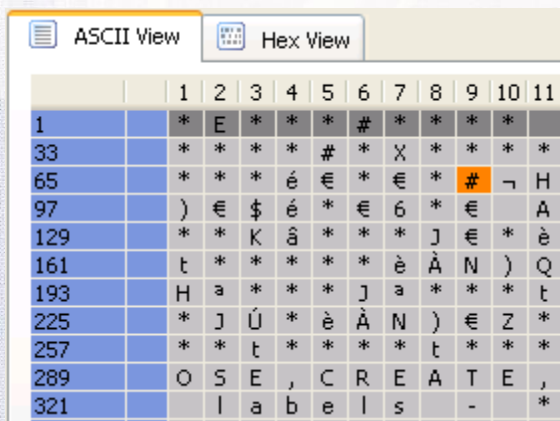
The final option “Editing Mode” is possibly the most powerful feature of S/Dump as it permits the editing in place of the actual frame. This can be very useful if a GFE is detected and you want to fix it by hand.

SPREZZATURA

Now double clicking on a character invokes the character dialog with the character clicked on pre-highlighted.

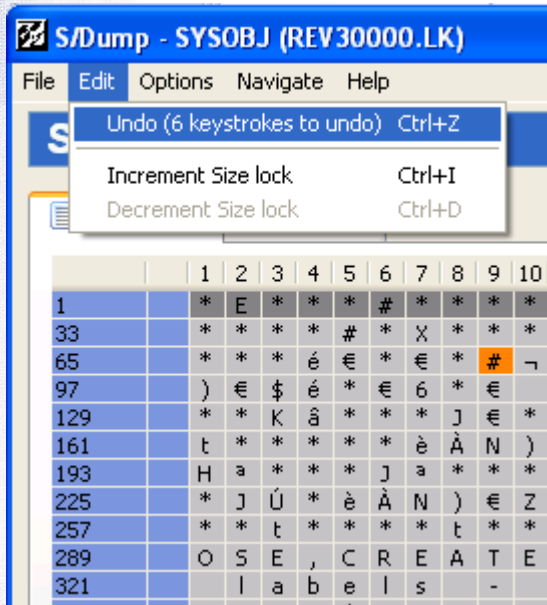


So let's change branch to labels

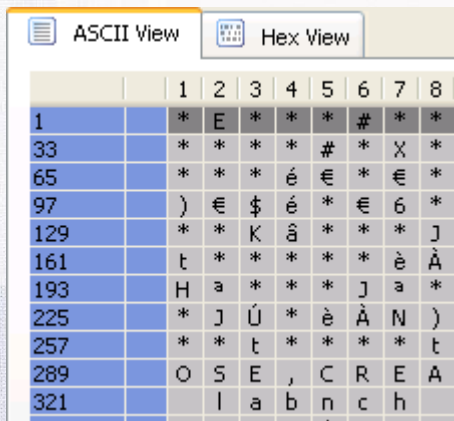


And look at the Edit Menu

SPREZZATURA



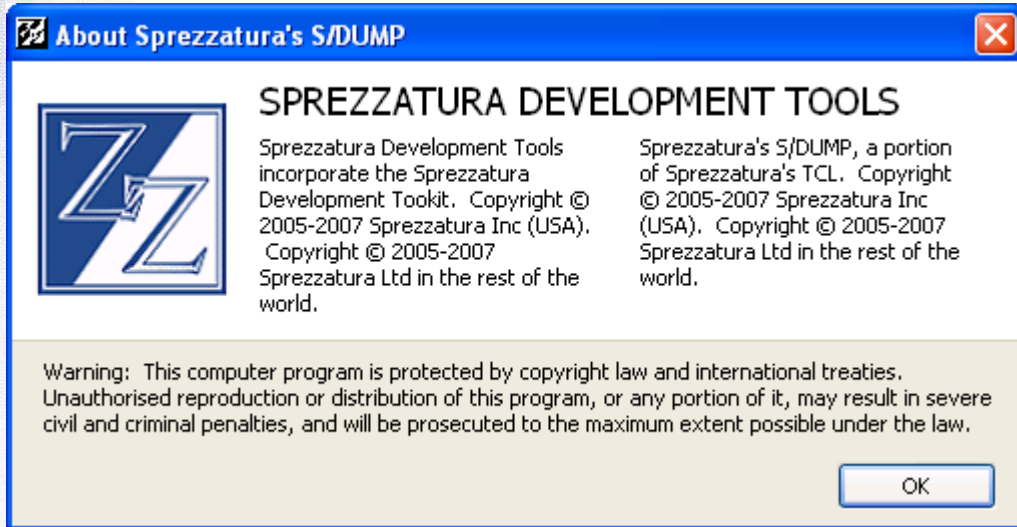
It has noticed 6 characters changed so let's press Ctrl-Z three times



And undo our changes for the last three characters.

The other menu options are self explanatory. As mentioned previously the options which compress of fix GFEs use underlying RevSoft code. The only other option is the Help/About option which simply displays this screen

SPREZZATURA



Here at Sprezz Towers we believe this utility to be 99% complete and after rigorous internal testing we're now happy to release to the people who originally volunteered to beta and to any other interested parties. If you'd like to join the beta then please mail info@sprezzatura.com with subject "S/Dump Beta".

Enjoy!

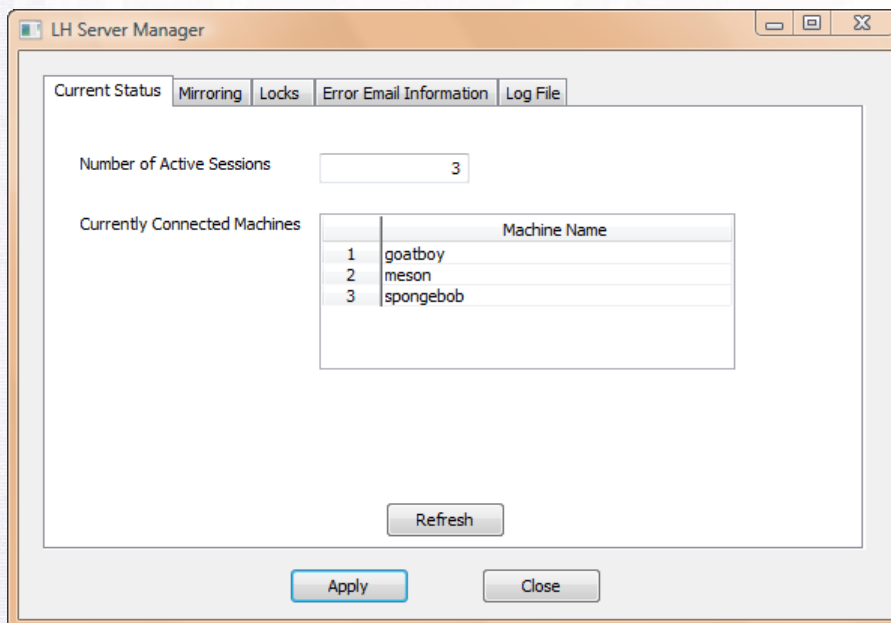
SPREZZATURA

Advanced Universal Driver Heavy Configuration

With the release of the Universal Driver Heavy 3.1 (UDH), there are a host of new configuration settings available to help fine tune the service to better suit the needs of your installation. When you install the UDH, an "LH Manager" directory is created underneath the installation directory. This directory contains a stripped down version of OpenInsight which contains the "LH Manager" application. To run, just execute OINSIGHT.EXE and login to SYSPROG. You will be displayed with the LH Manager application window.

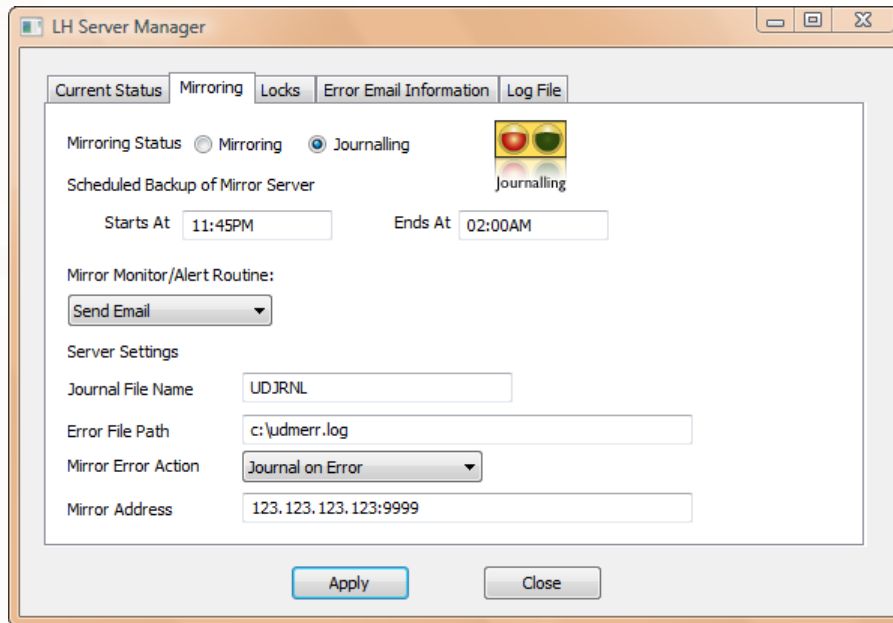
LH Server Manager Window

This window allows you to configure most of the settings you will need to have your UD Heavy server up and running. From this window, you can see the current sessions that are logged in:

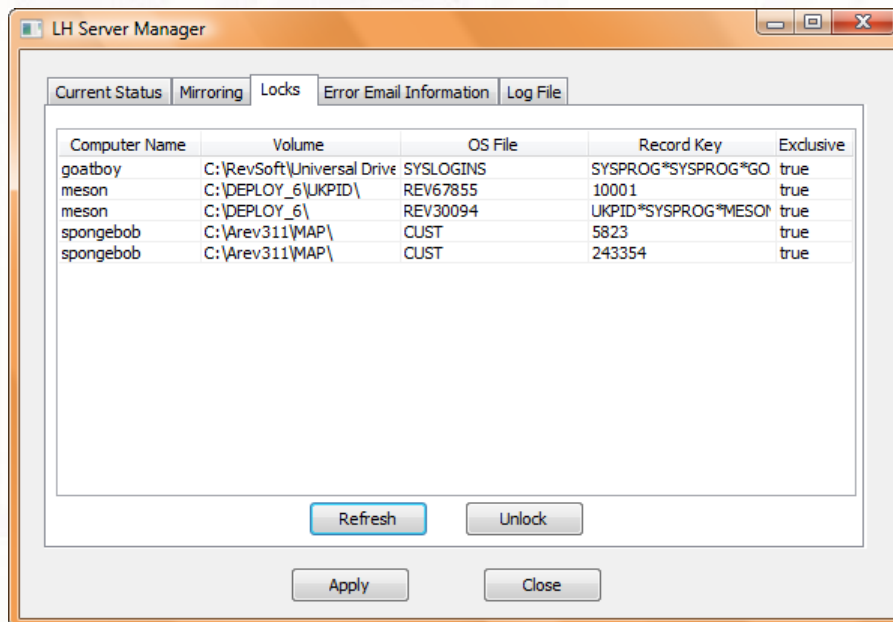


SPREZZATURA

The current status of mirroring and journaling, along with what to do in case of errors:

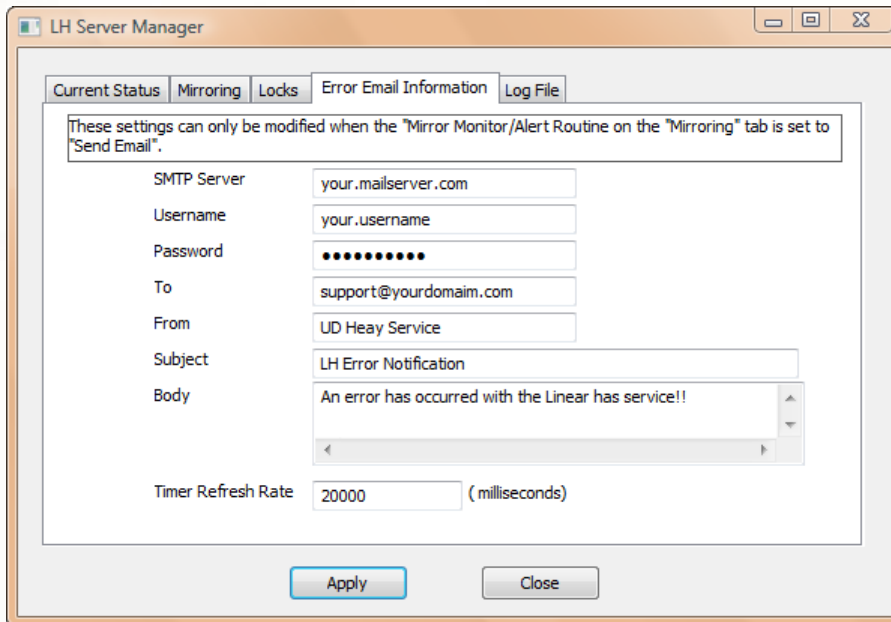


What records are currently locked on the server:

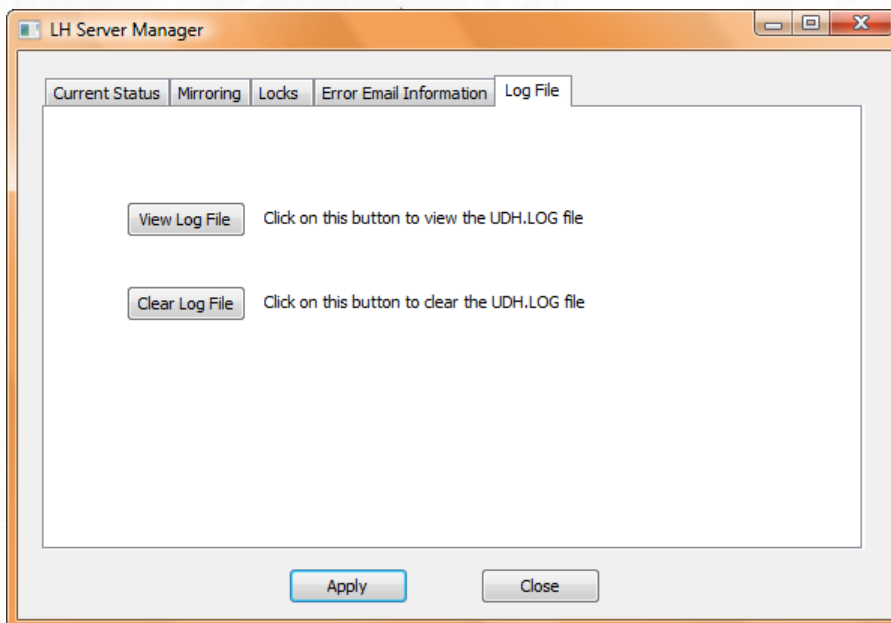


SPREZZATURA

The layout of the message that can be sent by email on an error:



And the ability to view or clear the log file:



Most of these configuration settings get stored in the registry under the “My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\RevSoft\Revelation Universal Driver Heavy\3.1” registry key. There are a few additional settings that must be modified through the registry.

SPREZZATURA

Registry Entries Used By UDH

The table below shows all the registry values created by the UDH installation, along with its description, possible values, and text label in the LH Manager application. A Server Manager label of "N/A" means that the registry setting cannot be modified by the LH Manager.

<i>Registry Key</i>	<i>LH Manager Label</i>	<i>Description</i>	<i>Values</i>
AlertData	N/A	This field does not appear to be used	
AlertType	Mirror Monitor/Alert Routine	Determines what the Server Manager application does on error.	No Action Send Email
AnonymousUserName	N/A	This is the anonymous user name for named pipes access.	
ExcludedPaths	N/A	A listing of directory paths that not tracked by the UD Heavy. Multiple directories are separated by semi-colons (;). This defaults to the UD Heavy Service manager location.	
MirrorBackupEnd	Ends at	Defines the time that automatic mirroring service stops journaling, and begins to mirror transactions again. The service also starts posting the journal entries at this time. You would set this to a time just after the backup of the mirror server stops.	Time is OCONV MTH Format ex. 2:00AM
MirrorBackupStart	Starts at	Defines the time that automatic mirroring ends and journalling begins. You would set this to a time just before the backup of the mirror server starts.	Time is OCONV MTH Format ex. 11:45PM
MirrorErrorAction	Mirror Error Action	What should the system do on an error. Be aware that setting this value to 0 (Send error and halt) means that the LH service will terminate.	0: Send error and halt 1: Journal on error
MirrorErrorFile	Error File Path	The location and name of the error file. This defaults to "c:\udmerr.log"	
MirrorInBackup	Mirroring Status	Indicates whether the service is in mirror or journal mode.	1: Journaling 0: Mirroring

SPREZZATURA

MirrorJournalDir	N/A	This is the location the UD Heavy will store the journal files. This defaults to the UDH installation directory.	
MirrorJournalName	Journal File Name	The initial prefix name of the journal files. This defaults to UDJRNL. The initial file starts UDJRNLO. Each subsequent journal file is incremented by 1. New journal files are created at about 1,000,000 characters.	
MirrorLogFile	N/A	The name of the mirror log file. This defaults to "udh.log" in the installation directory.	Defaults to c:\revsoft\ntservice\UDHeavy\udh.log
MirrorServerName	Mirror Address	This is the IP address and port address of the mirror server. This defaults to the IP and port numbers entered during installation.	IP and port in the format 123.123.123.123:12345
NamedPipeName	N/A	Named pipe name used if named pipe connections are used.	
TcpIpPort	N/A	TCP/IP port for access to this server. This defaults to the IP address entered during installation.	

You might have noticed that there were no registry settings for the email on error data. The email configuration information is stored in the CFG_UDHMGR_EMAIL record in the SYSENV table.

Email on Error Information

Field	Description
1	SMTP server name
2	SMTP user name
3	SMTP password (in clear text)
4	Recipient address
5	Subject Line
6	Message Body
7	Timer Refresh Rate
8	Sender Address

SPREZZATURA

Unlocking Other Station's Locks

One of the flaws of the UDH's monitoring and maintenance is that in order for journalling to kick in or to clear out stray locks, you must be logged onto a desktop on the server. You don't necessarily have to physically be at the server, you could use screen scraper (pcAnywhere or VNC) or connect in through a terminal server or other remote control software. However, some organisations might have security policies that prevent this, while others might not want their server logged in to an active desktop.

It is possible to change these settings remotely using direct RTP57A calls. Please note that Sprezzatura does not endorse using these function calls, and that they should be considered highly undocumented and subject to change by Revelation at any time. As always with undocumented features, there might be ramifications to using these functions, as outlined with the unlock call. You should only use these if you cannot gain access to the main server.

You can change the journalling and mirroring status by calling RTP57A such that

```
rtp57a( 'CO', '', 0, '', journalMirrorState, errorCodes, status )
```

where journalMirrorState is a valid value for the MirrorInBackup registry value.

To unlock a record, you would call RTP57A such that

```
rtp57a( 'CO' recordKey, fileHandle, osFileName, funcValue, errorCodes, status )
```

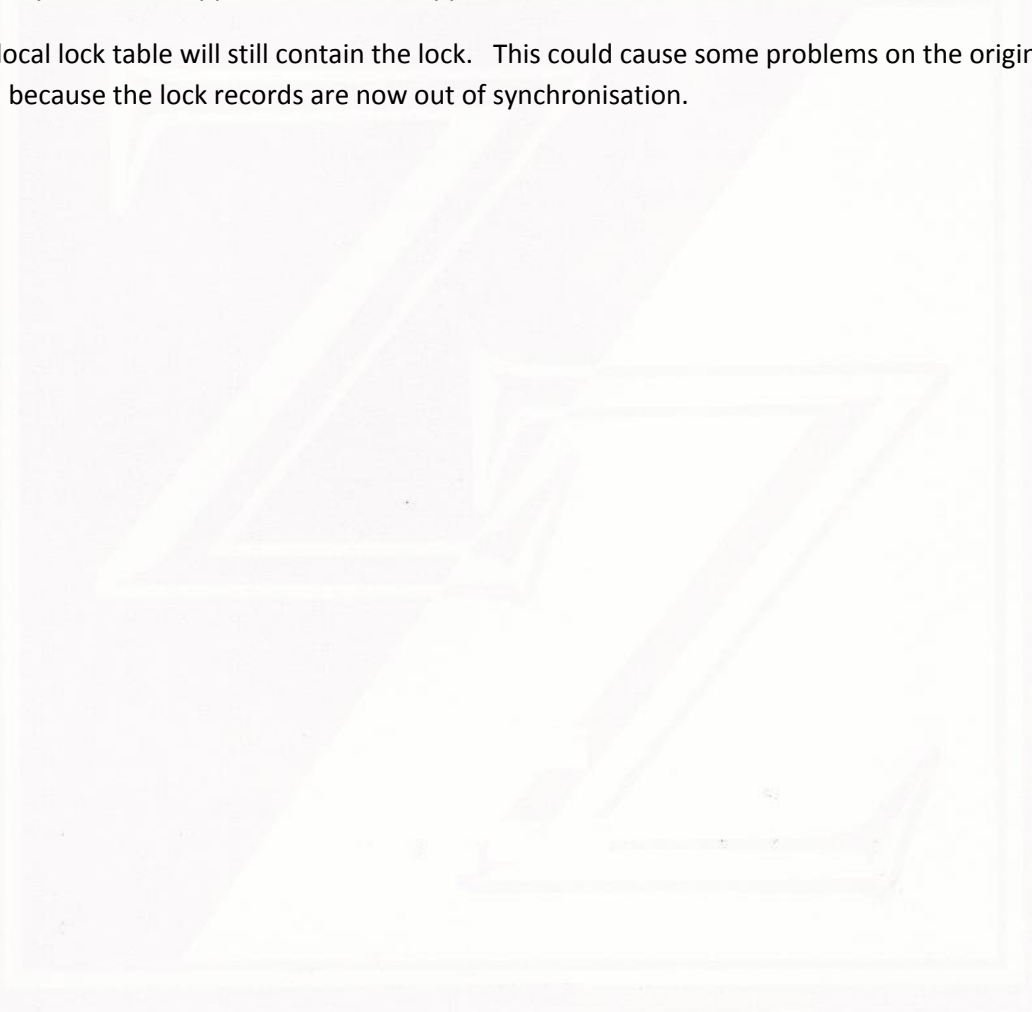
<i>Variable</i>	<i>Description</i>
recordKey	The key you wish to unlock
fileHandle	The second @VM delimited value of the handle returned by getRawHandle_sub fileHandle = getRawHandle_sub(tableName) fileHandle = fileHandle< 1, 2 >
osFileName	This is the OS file name as the file is known on the server without the .LK or .OV extention.
funcValue	This appears to be the number 2
errorCodes	Returns back the contents of @FILE.ERROR.
status	BFS status value. 1 for success, 0 for failure

SPREZZATURA

There are some things you need to keep in mind if you plan on implementing this functionality in your applications. First is that the workstation must be running the UD 3.1 Heavy driver. This function does not work on previous versions of the UD driver, including UD 3.0.0.3.

Secondly, the `osFileName` variable must be the name as the server knows this. For example, suppose you have a server directory 'C:\UserApps\SalesFloor\DBapps\OINSIGHT', which is where OpenInsight resides. You have a share on the SalesFloor directory as 'SalesFloor'. The workstations on the sales floor map this share so that drive P is mapped to '\\myserver\salesfloor'. Even though the workstation would see a file as 'P:\DBapps\OINSIGHT\SalesVol', you must pass in `osFileName` with a directory of 'C:\UserApps\SalesFloor\DBapps\OINSIGHT\SalesVol'.

Finally, the local lock table will still contain the lock. This could cause some problems on the original workstation because the lock records are now out of synchronisation.



SPREZZATURA

Peripheral Trivia

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

TV:	Dr Who - Blink!
Book:	Secrets – Sissela Bok
CD:	The return of the magnificent – DJ Jazzy Jeff
WEB:	http://www.miguelmesas.com/fanfilms/VerSLAINE.htm

[\[top\]](#)

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 - ©2007 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

SPREZZATURA

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 4 Issue 5, 30th July 2007.

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.

