

IDMS

connections

THE publication of the IDMS User Association

The CA-IDMS Database and Applications User Association

<http://www.iuassn.com>

March 2009, Number 70

INSIDE THIS ISSUE

Letter from Editor ...	1
Message from the International Chair ...	3
Benutzerkonferenz ...	3
CARE 2008 ...	4
CA World ...	4
Gebruikersdag/Jour Des Utilisateurs IDMS ...	5
Messages to Management ...	6
The COBOL XML Parser ...	8
CA IDMS CV Journal Sizing ...	9
We are I5 Going on I7 ...	12
Why is my Load Area so Full? ...	13
Preparing CA IDMS TCP/IP for Daylight Savings ...	14
Is it OK if I Sign on? ...	15
Securing CA IDMS TCP/IP Sockets ...	16

LETTER FROM THE EDITOR

This edition's "really big" news is the launch of "Mainframe 2.0" at CA-World, and of course the rampant success that CA-World 2008 was for all who attended - please see the report inside this edition.

The other news of significance to IUA members is that we have now changed our name from IUA to IUA. That is from the I nformation U ser A ssociation back to the original name of I DMS U ser A ssociation. This allows us to have IDMS play a more prominent role on the cover of Connections, and at our web site. Making the changes make take a while - but they will happen in the fullness of time, I'm sure!



This month we have a particularly large crop of useful information from CA. Originally I was going to hold an article, or two, back for next edition, but I couldn't decide which article we could ALL do without. This just means IUA and CA will have to work a bit harder to put September's issue together.

One of the articles, by Dave Pomeroy, is about the "Daylight Saving Time Issue". I expect most DBA's will want to spend some quality time reading this article - here is a relevant extract to pique your interest: In 2009, daylight saving time will begin on March 8 and will end on November 1. So as you are reading this, you should be taking the steps necessary to provide a forward transition for your CA IDMS systems to daylight saving time on March 8, 2009.

I don't know if you will have this article in time for March 8th, but you will certainly have it in time for November 1st - remembering that when our southern hemisphere users "spring ahead" is when the "northerners" do their "fall back".

(continued on page 3)

Backed out application changes?
Failed migrations?

Implement
**‘Zero
tolerance’**

with Change Control Manager (CCM)
from **zTools**.

CCM for **CA ADS** is a proven solution (developed within an experienced **ADS / COBOL** user for their own use) for effective and consistent change management. It enforces a degree of rigour, control and audit while improving developer productivity.

As a result it reduces error rates and the lost time and effort involved in failed migrations, configuration errors and backed out changes.

Do you have the tools to do your job as effectively as possible? How much time and effort does your organisation waste through failed migrations, configuration errors and backed out changes? What steps are you taking to ensure that your auditors are satisfied with the controls you have in place?

CCM for **CA ADS** provides the missing piece of the jigsaw for many **IDMS** users, without adding unnecessary process or functional complexity. What's more, it is easy to install and use, with low operating costs.

Because **CCM** for **CA ADS** is in production and delivering on its promise today, **zTools** are offering a unique fifty day money back guarantee on the license for all new customers

For further information on
CCM for **CA ADS** and **zTools**,

visit www.ztools.co.uk email info@ztools.co.uk

Have you developed your own solutions to help administer your **CA ADS / IDMS** or other **z/OS** environments? Would you like to know more about a cooperative business relationship with **zTools**?

For further information
contact businesspartner@ztools.co.uk



Proven z/OS solutions
sourced for you.



MESSAGE FROM INTERNATIONAL CHAIR

By Terry Schwartz

I do not know anyone who has not been affected in some way or another by the current global economic turmoil. Perhaps your retirement plan has taken a beating, your

company has reduced their benefits, or the rising cost of living is taking a larger chunk of your pay check. Our employers are feeling the effects of a weak economy and taking measures to ride out the storm. Even healthy companies are cutting back in an effort to anticipate the effects of the recession.

So how does all the economic uncertainty relate to you, the IUA, and CA IDMS?

I believe this is an excellent time to show management how to maximize their return on investment in CA IDMS. The time to get the most out of the tools you have at hand is when you have a “do more with less” business environment. Often management does not realize the power of the software already installed in their data center unless someone gives them the information.

I urge you to carry the message to your management and users that your existing CA IDMS software can take on new rolls in your business environment. Spread the word that CA IDMS is more than a database management system and green screen application platform. Let your companies know that CA IDMS can be a host to or can host web services, that CA IDMS data can be integrated into Java or .Net applications, that CA IDMS data can be accessed from applications as simple as Microsoft Excel, and that extensions from email to data encryption can be incorporated into your existing applications.

If you need help getting this message to your management, perhaps the IUA can help. The IDMS-L is a great place to network and find new ideas on how to leverage your existing CA IDMS software. The IUA web site (www.iuassn.org) has a large library of past presentations, many of which are on CA IDMS modernization techniques. In the near future we will be hosting our first virtual workshop where you can learn about the most recent developments that can help you do more with your CA IDMS software.

Terry Schwartz

CA IDMS PLC Chair

Letter from the Editor cont'd from page 1

I can't think of anything else to comment on for this issue - I must have done a complete brain dump in the last issue. But, I would be remiss in not thanking our contributors from CA and our regular and ad-hoc contributors from within the IDMS User Community without whom producing this publication would not be possible. From myself, and on behalf of all the readers who gain from your stories and your insights into a wide range of topics - THANK YOU!

That's all there is because there is no more – cheers - Gary

Gary Cherlet
Justice Technology Services
President Australian IDMS User Group (OZIUA)
IUA International Board Member responsible for Connections

The opinions expressed in this editorial are the personal opinions of the editor and they may not be shared by the IUA Board or its members, other contributors to Connections, by Justice Technology Services, or the Government of South Australia.

BENUTZERKONFERENZ

The IDMS user community of Germany is pleased to have their “Benutzerkonferenz” on Tuesday and Wednesday May 19th and 20th, 2009 at Mannheim (Germany).

- Some interesting IDMS session will be IDMS Status and Plans Steve Urquhart, CA, USA
- Mainframe 2.0 für IDMS Steve Urquhart CA, USA
- TCP/IP within CA IDMS Laura Rochon IUA
- EZ Source The Applications CMDB product for IDMS
Ayala Nagel EZ Legacy
- QA at CA for IDMS Releases und Service Packs Steve Urquhart CA, USA
- VegaSoft web services solution for IDMS Jaap Houtman
Formula OpenSoft B.V.
- Erste Erfahrungen mit Release 17.0 Schwerpunkt Installation Josef Lust T-Systems Enterprise Services GmbH
- Encrypting CA IDMS Data CA World session MI280SN Laura Rochon IUA
- IDMS r17 zIIP feature Manfred Höfer CA Deutschland

For further information and final agenda please contact peter.kotowski@generali.de or visit CA German website <http://www.ca.com/de/about/content.aspx?cid=193837>

CARE 2008

The main highlights of CARE included meeting other user Group presidents from around the world and sharing “best practices” for running User Groups. Another highlight was getting to meet the various VP’s from key product areas and getting to see “previews” of the main features of the announcements that were going to be made during CA-World in the following week.

I was asked to talk to the 120+ User Group presidents about “Best Practice for Publishing a User Group Newsletter”. Only 3 delegates were asked to speak at CARE so this was quite an honour. It got better when the IUA was presented with an award for “Worlds Best Practice for a User Group Newsletter” - which, as editor, I accepted on behalf of the IUA and all of our contributors.

A number of the attendees asked for a copy of my presentation and the “planning” documents we have written to help keep us “on track” when we publish each edition of “IUA Connections”.

CA-WORLD 2008

The main points being covered are:

- CA have a new focus on the Mainframe (CA Rock the Mainframe),
- New "Mainframe 2.0" to simplify implementation and configuration of CA mainframe products,
- zIIP engine exploitation,
- IDMS r17 features, and
- IUA meeting with John Swainson.

CA Rocks the Mainframe

CA have never focussed as much on the mainframe as at CA-World this year, at least not in the 8 years I have been attending CA-World(s). The Keynote speeches made note of: Green IT, Reduction of power usage, Smaller footprint, Centralised IT security, Recoverability, Supportability etc. This is not new stuff to us, but it was nice to hear recognition that the Mainframe provides all these solutions in one box.

It was full house for the Mainframe kick-off session. There was a definite buzz of anticipation. It was hosted by the Executive VP and General manager of the Mainframe Business Unit, Chris O'Malley. There was standing room only, and many more people outside couldn't get in.

Key messages:

Launch of Mainframe 2.0 (see below)

Mainframe Graduate education in Centres Of Gravity.

CA are partnering with colleges and universities to generate and produce graduates with mainframe skills. (IBM have a similar project). Hopefully, this will enhance and extend the life of mainframe technologies, including IDMS, far into the future.

Mainframe Value Program.

Headed by Mike Zinder, this program will assist businesses in understanding how to extract the most value from their mainframe investments.

In celebration of this new, enlightened, focus on the mainframe, all the attendees in the Mainframe focus area were invited to a “CA Rocks The Mainframe” Concert a few blocks down the road. There was a live Rock band playing rock hits from the 70s and 80s. It was a perfect event for the mainframe demographic.

Mainframe 2.0

There was an exciting announcement to launch a new facility called “Mainframe 2.0”. This is a facility which will simplify then installation, configuration and tuning of CA’s Mainframe software, including IDMS. The challenge being addressed is to increase the productivity of new, and inexperienced mainframe people. For example, it will install products without an in depth knowledge of SMP/E. It will give the installation the same feel and function as the familiar desktop software downloads.

There was a live demo! It showed how it is possible to examine an environment, see what is installed, suggest upgrades or new install, and by selecting the “download and Install” option, to schedule and automatic installation, through download, unpack, SMP/E Receive and Apply and, if required, the deployment and installation of the software. It looked impressive. (See session id MF102SN for further details).

I look forward to seeing more about mainframe 2.0 in future.

zIIP/zAAP exploitation

There were a few sessions describing the exploitation of IBM’s new engines. By shipping CPU cycles to the zIIP and zAAP engines, general CPU costs may be reduced. IDMS can exploit the zIIP engines to a significant degree from r17 onwards. Also some of the CA partners and other connectivity software will do the same.

IDMS r17 Features

There was a lot of good information in sessions surrounding the recent GA release of IDMS r17, including a general session on features and enhancements, a session on the zIIP exploitation and a session on the experience of a customer beta testing IDMS r17. Some features of IDMS r17:

- SQL enhancements
- Performance improvements (including zIIP)
- Non-Stop processing changes
- DBA and operations enhancements.

It was good to see how many DARS (now known as enhancement requests) are included in r17. It shows that enhancement requests can, and do, find their way into IDMS product eventually.

IUA meeting with John Swainson

Representatives of the IUA and IDMS PLC (and also with CADRE) were invited to a face-to-face meeting with CA’s CEO John Swainson. There was a lot of discussion about the influence Mr. Swainson could have on the opinion of our senior executives in our companies over the use of IDMS.

(continued on page 5)

His answer was very encouraging. Mr. Swainson said that he was in contact with many CIOs of many corporations and is currently active in promoting IDMS directly to CIOs of IDMS customers. We were asked if we could help identify the appropriate people within our organisations to influence decisions involving IDMS technology.

In summary

- The mainframe rocks!
- CA-World '08 was the best ever.
- IDMS is being promoted.
- IDMS is being enhanced.
- Facilities are coming to encourage "new blood"

The next CA-World (April 2010) will be in Los Angeles
- see you there?

WWW.IUASSN.ORG

YOUR PORTAL TO IUA SERVICES AND IDMS CONTACTS

GEBRUIKERSDAG/JOUR DES UTILISATEURS IDMS

The BeNeLux IDMS user group will have their IDMS Gebruikersdag/Jour des utilisateurs IDMS on Friday May 15th at Brussels CA Belgium office.

Preliminary agenda:

- IDMS SQL session given by Gary Cherlet via webcast with Australia
- IDMS encryption given by Laura Rochon
- IDMS status and plan given by Steve Urquhart
- QA at CA for IDMS releases and service pack also given by Steve Urquhart
- Vega soft web Services given by Formulaopensoft
- IDMS basics and introduction or Java application in IDMS given by Luc Hermans
- CA mainframe 2.0 given by Steve Urquhart
- EZ-source for ADS given by Simon Jennings

For further information and final agenda please contact Jan.Rabaut@sogeti.be or Luc.Denaeyer@colruyt.be.

The European IDMS user Association will have their Board Meeting with CA on Monday May 18th, 2009 at Utrecht (The Netherlands)

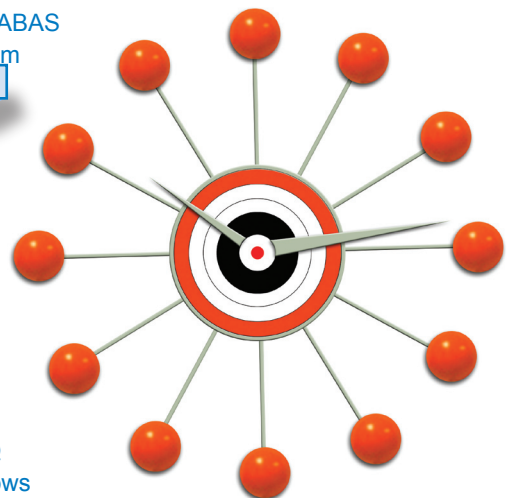
Regional user group presidents and chair persons are cordially invited to join the EIUA meeting. For further information please contact the EIUA board at Jan.Rabaut@sogeti.be

It's About Time and Money.

Best-of-Breed Tools For Mainframe Integration, In Real Time, To Any Target...

Treehouse Software enables your organization to become a more efficient enterprise—one that can easily adapt to today's fast-moving and uncertain conditions.

We offer the most complete and flexible portfolio of mature solutions available anywhere for integration and replication of data between mainframe sources and any target, application, or platform. Using ETL, CDC, and SQL/XML technologies, we connect your enterprise—from *anything* to *anything*.



Treehouse Software, Inc.
409 Broad Street, Suite 140 • Sewickley, PA, USA 15143
Phone: 412.741.1677 • Fax: 412.741.7245 • Web: www.treehouse.com

Since its establishment in 1982, Treehouse Software, Inc. (TSI) has garnered an excellent reputation as a leader in systems software development and marketing, with approximately 700 customers in the U.S. and dozens of other countries. In addition, TSI provides training and consulting services and maintains a full enterprise support infrastructure for customers worldwide.

MESSAGES TO MANAGEMENT:

By Don Casey

My intent is for this to be the first of a series of articles intended to take a management view of IDMS-related topics. I understand that most readers of this periodical are practitioners, not pointy-haired-bosses (I've been both): even so, by looking at things from a different



perspective one gains insights. My hope is you will find some of the viewpoints in these articles useful in either understanding senior IT management, or perhaps even in influencing management's behavior. You may not have any practical application for any of what follows in your day-to-day work, but someone in your shop might (please share!), in any event I hope you enjoy the articles.

Women and Children First!

The already leaking IT Financial ship has hit the iceberg of recession, and we're bow-down and sinking fast. Things were bad enough during the "good times", when companies made money and just wanted to squeeze IT a bit so they could make even more money... now almost everybody is having serious revenue shortfalls, and cutting expense becomes a matter of corporate survival, not mere greed.

We are past the point of arguing whether we should be using wooden lifeboats or inflatable rafts... you use whatever is handy that can get the job done before it's too late.

The point of this rant: if high on your survival agenda is saving mainframe costs, you need to take a hard look at what your projected NET savings are (savings – costs to implement), and how soon you expect to achieve those savings. Without debating the actual, realizable savings one might achieve in mass migrating existing applications from the mainframe to an alternate platform (the most likely topic for my *next* epistle), the point remains large migration efforts almost certainly will take more time than you have right now. You need savings this year and maybe next, not a 5-year ROI project. The solutions you should be looking at today need to have both a quick return and a manageable upfront investment.

You have available a few generic approaches to either reduce the size of your mainframe, or (more likely) stave off that next upgrade for as long as you can:

- Peak Management
- Demand Management
- Application Tuning
- Workload Relocation

I'll say a few words about each (for now), but as the final topic has interesting IDMS implications we'll cover it in more depth at the end.

Note: all of these approaches assume someone or some group in your organization has the information and tools necessary to understand what is being done, when, why, and by whom in your mainframe workload. This information is critical in order to intelligently drive a CPU reduction project.

Peak Management: you need to size your mainframe to handle the peak load. Analogy: how big a freeway do you need if everybody in the city drives to work at exactly 8am every day? If you are using 1,000 MIPs from 8am to 5pm, but the machine sits idle for the remainder of the day, you have a huge opportunity. What can you do to adjust the workload to level out the peaks (to have people stagger their commute)? User-driven workload (online transactions, etc.) is difficult to control, but scheduled (batch) workload provides opportunities. Have you analyzed what work is being done during your peak processing periods, and can you move some of it to your slack periods? Bulldoze the workload peaks into the valleys.

Demand Management: are your users doing things they really don't need to do? Analogy: are teenagers cruising Main Street repetitively every Friday night? Can you identify (by application and requestor) significant workloads that appear to be subject to negotiation or control? Is that daily report really needed daily? Could it be made into a weekly? Focus on those things running during the peak periods.

Application Tuning: are (parts of) some applications just pigs? Can you identify what transactions and batch jobs are major CPU consumers? Are any of them worth a performance code review? Do any of them appear candidates for Demand Management? Again, focus on those things running during the peak periods.

All the above efforts should be well within the capabilities of a seasoned IT department, given some priority-setting and appropriate project leadership. All of them most shops do from time to time, but not always as a focused effort. This may be the time for a focused effort. Pull a team together and task them with holding back the tide. In medium to large environments every month you can defer an upgrade can save hundreds of thousands of dollars in CPU-related charges. The payback can be both large and quickly achieved, especially if aggressive workload management has not been the norm in your shop.

As many of the decisions to be made relate to application and usage-specific questions, engagement of in-house staff is almost always required, but project leadership and direction could come from an external consultant, if no in-house resource is available.

What follows is a more of a medium-term option, but one with both the potential for significant CPU savings. It also has some side benefits as well.

Workload Relocation: are there some workloads, or portions thereof, that can be moved to a less expensive platform? The "mass migration" scenario I discarded

(continued on page 7)

as a short-term option at the beginning of this article is the extreme example; there are other, faster ways to move some workload off the mainframe. What if instead of moving entire applications, we move only that functionality which can be moved easily and with minimal disruption. The mainframe (and IDMS) architecture stands second to none in being able to deliver stable and well performing transaction-based applications.

However; the mainframe has no particular advantage in a reporting or data warehousing application. The proliferation of reporting tools and relational databases of all shape and hue on distributed platforms provides reporting capabilities that are equal to or superior to what the mainframe can provide. What is hard is getting them to work cheaply against mainframe data. As long as the data remains on the mainframe, it takes mainframe cycles to support reporting requests from other platforms.

This has led to a well-understood and widely embraced approach of creating a clone of selected mainframe data on a distributed platform, to support reporting/retrieval processes. Creating these second copies can be done by using a commercial product, or through in-house developed methods (sometimes requiring application modifications).

Back to our dilemma: you need a quick hit. Off-the-shelf can be implemented much much quicker than trying to create a unique in-house solution. Several robust commercial products exist which allow mainframe databases, such as IDMS, to be replicated to these other, cheaper platforms. These products can be implemented generally without any changes to existing mainframe application code. Since many shops already have end-user reporting tools that are used against relational data, providing access to replicated mainframe data may actually be viewed as an enhancement by the end-user community.

The catch is; you need to pick a product and spend some money. So if you choose to look into this option, and as you look at products, consider these questions:

- What workload will you be able to offload? Will it be enough to make this worthwhile?
- How soon can you implement and offload work? How long before I recoup my investment?
- What overhead does a particular replication product add to the mainframe? This eats into your offload savings.
- Any product limitations? What can't it do that I need it to do? What changes do I have to make to existing mainframe applications or environments?
- Any additional functionality in the product you may need for future projects? If I buy it for simple replication now, what future needs might I have that I need to think about as I commit to a given product?
- Any additional benefits to the company from creating a replicated data store? What benefits can I show the business community from having a local relational copy of mainframe data? Can I allow local applications to access this data?

Bottom line: shops looking to quickly reduce CPU usage, or even just slow the upgrade cycle, should place immediate focus on managing workload and user demand, and tuning those application components which are significant contributors to peak usage. Beyond that one should take a serious look at what specific workloads might be quickly and easily offloaded; either through data replication or other means.

Don Casey

Principal Consultant

Run Right, LLC

Don first learned about IDMS as a systems programmer at Contra Costa County, where he installed release 3.x over the July 4th weekend in 1975.

After being lured to Cullinet in 1979, Don held positions in field and phone support, both as technician and management. He became one of a select group of Senior Consultants known for their expertise in handling difficult problems.

In 1989 Don joined APL Limited, rising to the rank of Vice President of Technical Services.

In 2007 Don left APL to join Linda at Run Right, LLC.

WWW.IUASSN.ORG

**YOUR PORTAL TO IUA
SERVICES AND IDMS
CONTACTS**

**CONTRIBUTED
SOFTWARE
LIBRARY**

Save time
and use the
experience of
others to resolve
problems.

THE COBOL XML PARSER

By Kay Rozeboom

In the previous issue of “*IUA Connections*”, I explained how to extract a SOAP message from an HTTP message. In this issue, I will demonstrate how to extract information from a SOAP message.

Figure 1 shows a SOAP message returned by a stock price web service. (This is the same web service used for the examples in my previous article.) The part of the message that we are actually interested in, the stock price, is highlighted in green. You cannot treat the SOAP message as a fixed record layout because no part of the message is guaranteed to begin or end in any particular column. The way to locate the stock price is to find its start tag “<GetQuickQuoteResult>” and its end tag “</GetQuickQuoteResult>”. These are highlighted in peach. The stock price is located between the start and end tags. These three fields (the start tag, the stock price itself, and the end tag) together constitute an XML element.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetQuickQuoteResponse xmlns="http://ws.cdyne.com/">
      <GetQuickQuoteResult>
        112.55
      </GetQuickQuoteResult>
    </GetQuickQuoteResponse>
  </soap:Body>
</soap:Envelope>
```

Figure 1

A SOAP message is a type of XML document. It is possible to locate information within an XML document by using the COBOL “INSPECT” command, or by defining the XML document as a table and looping through it with a subscript. But these methods are clumsy and time-consuming to code. Thankfully, IBM has provided a better solution: the “XML PARSE” command.

XML-EVENT	XML-TEXT
-----	-----
START-OF-DOCUMENT	
VERSION-INFORMATION	1.0
ENCODING-DECLARATION	utf-8
START-OF-ELEMENT	soap:Envelope
ATTRIBUTE-NAME	xmlns:soap
ATTRIBUTE-CHARACTERS	http://schemas.xmlsoap.org/soap/envelope/
ATTRIBUTE-NAME	xmlns:xsi
ATTRIBUTE-CHARACTERS	http://www.w3.org/2001/XMLSchema-instance
ATTRIBUTE-NAME	xmlns:xsd
ATTRIBUTE-CHARACTERS	http://www.w3.org/2001/XMLSchema
START-OF-ELEMENT	soap:Body
START-OF-ELEMENT	GetQuickQuoteResponse
ATTRIBUTE-NAME	xmlns
ATTRIBUTE-CHARACTERS	http://ws.cdyne.com/
START-OF-ELEMENT	GetQuickQuoteResult
CONTENT-CHARACTERS	112.55
END-OF-ELEMENT	GetQuickQuoteResult
END-OF-ELEMENT	GetQuickQuoteResponse
END-OF-ELEMENT	soap:Body
END-OF-ELEMENT	soap:Envelope
END-OF-DOCUMENT	

Figure 2

The “XML PARSE” command reads through the XML document sequentially. Every time it encounters what it considers an “XML event”, it returns control to your program, along with the following special registers:

- XML-EVENT: the name of the event
- XML-TEXT: the document text associated with the event
- XML-CODE: the return code from the parser

Figure 2 lists all of the XML events in our SOAP message. The parser will return control to your program for each one of them. Your program must determine which events are of interest, and which can be ignored. This is done in the “parsing paragraph” shown in figure 3. Note that the color coding in figure 3 corresponds to figures 1 and 2.

```
2000-PARSE-XML.

EVALUATE XML-EVENT

  WHEN 'START-OF-ELEMENT'
    MOVE XML-TEXT TO SAVE-CURRENT-ELEMENT

  WHEN 'CONTENT-CHARACTERS'
    EVALUATE (SAVE-CURRENT-ELEMENT)
    WHEN 'GetQuickQuoteResult'
      MOVE 'YES' TO WAS-STOCK-PRICE-FOUND
      MOVE XML-TEXT TO DISPLAY-STOCK-PRICE
    END-EVALUATE

  WHEN 'END-OF-ELEMENT'
    MOVE SPACES TO SAVE-CURRENT-ELEMENT

  WHEN 'EXCEPTION'
    PERFORM 2100-HANDLE-XML-ERRORS

END-EVALUATE.
```

Figure 3

The important thing to remember is that the parsing paragraph will be executed many times - once for every XML event. The pieces of code that you test for will be executed every time a particular event occurs. For our SOAP message, the ‘START-OF-ELEMENT’ code is executed four times, the ‘CONTENT-CHARACTERS’ code once, and the ‘END-OF-ELEMENT’ code four times. Although there is only one instance of ‘CONTENT-CHARACTERS’ in this example, there will usually be more. That is why we save the start tag name when ‘START-OF-ELEMENT’ occurs. Then we can check it when ‘CONTENT-CHARACTERS’ occurs, to determine which element this is.

```
XML PARSE WS-INPUT-XML.
PROCESSING PROCEDURE 2000-PARSE-XML

ON EXCEPTION
  MOVE 'NO' TO IS-IT-OK-TO-CONTINUE
  MOVE XML-CODE TO DISPLAY-XML-CODE
  DISPLAY '--- XML DOCUMENT ERROR: ',
    DISPLAY-XML-CODE
NOT ON EXCEPTION
  DISPLAY 'XML DOCUMENT SUCCESSFULLY PARSED'

END-XML.
```

```
05 DISPLAY-XML-CODE PIC ++++++9.
```

Figure 4

(continued on page 9)

Figure 4 shows how to code the “XML PARSE” command itself. The first variable (WS-INPUT-XML) specifies the location of the XML document to be parsed. This will usually be a field in working storage or in the linkage section. The second variable (2000-PARSE-XML) is the name of the parsing paragraph used to parse this document. The “ON EXCEPTION” code is executed if an error occurs while parsing the document. The “NOT ON EXCEPTION” code is executed when the parse completes successfully.

Figure 5 is a generic XML error-handling routine. This routine is called when an “EXCEPTION” event is encountered by the parsing paragraph in figure 3. The “IF” part of the code demonstrates how to bypass selected errors. The “ELSE” part shows how to display the location of the error in the XML document.

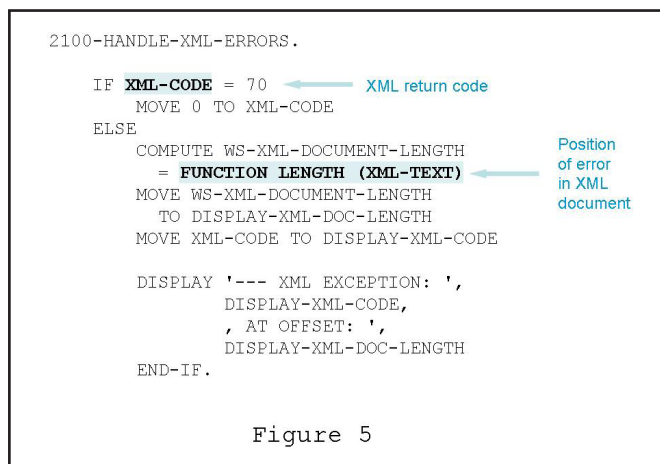


Figure 5

Additional resources:

- Both the “COBOL Reference” and the “COBOL Programming Guide” contain detailed information about COBOL XML processing.
- XML error codes are listed, with explanations, in Appendix D of the COBOL Programming Guide.
- The report in figure 2 was generated by a batch COBOL program that takes an XML document as input. You can download the source code and JCL for this program from the IUA Contributed Software Library at “<http://www.iuassn.org/>”. Look under the heading “Batch Utilities” for “Generic XML Parser”.

[*Editor’s note:* you must have “Joined” into this web site in order to enter and download the UCL portion. Having signed up to the CA User Group web site is not adequate.]

Biographical note:

Kay Rozeboom is a DBA/Systems Programmer with the State of Iowa. She has 20 years of IDMS experience. Her special interest is in integrating mainframe data and applications with other platforms.

CA-IDMS CV JOURNAL SIZING

By Gilbert Jonswold

Within a Central Version (CV) journal activity can be a hidden yet significant processing bottleneck for update transactions. When CA IDMS updates a page all journal blocks containing images associated with that page must be successfully written before an attempt is made to write the database page. Therefore the number of journal blocks needed to contain a page’s updates can significantly affect the amount of time a transaction must wait before a database page is written and the transaction can continue processing. **This article will attempt to help you minimize the journal bottleneck by addressing the 3 major journal sizing considerations for a CV:**

- Journal block size
- Journal transaction level
- Journal Buffer size

Journal Block Size

A goal of the DBA is to minimize the number of journal blocks needed to contain a page’s updates thus reducing the amount of time a transaction needs to wait for journal I/O. Unfortunately there is no straight-forward formula that can be used to estimate the ideal journal block size for a CV. There are numerous types of records written to the journals based on the types of processes that occur within a CV and each type may have a unique length. In addition a single journal block has the potential to contain records generated from multiple transactions making it very difficult to predict the contents of a typical block.

The most numerous types of records found within a journal are BFOR and AFTR records. These records reflect the contents of a database record before an update operation (BFOR) and the contents following the update (AFTR). The lengths of these records consist of 64 bytes of overhead and the length of the data record. The length of the data record should be calculated as the length of the record’s prefix and the data length.

The **most significant** recommendation regarding journal block size is to make it large enough to hold a BFOR and an AFTR image of your largest data record so that a single update might be contained on a single block although this can never be guaranteed due to the possible presence of other record types already existing on the block. However having a journal block smaller than this value will guarantee at least two journal blocks must be used to record the update thereby necessitating two writes to the journal file.

Once a journal block size is selected the output from a number of ARCHIVE JOURNAL jobs should be reviewed to determine the general percentage of space used on a journal file’s blocks. Figure 1 shows a sample space utilization histogram produced by the ARCHIVE JOURNAL utility.

(continued on page 10)

PERCENTAGE DISTRIBUTION PER PAGE

PERCENT NO OF PAGES

0-10	6
11-20	7
21-30	33
31-40	112
41-50	46
51-60	94
61-70	11
71-80	556
81-90	928
91-100	595

DISK BLOCKS OFFLOADED	2,388
TAPE BLOCKS WRITTEN THIS SEGMENT	262
TOTAL TAPE BLOCKS WRITTEN	262

Figure 1

If the majority of pages fall into the higher utilization range of the histogram it is probably an indication that the selected block size for the journal file is too small and should be adjusted so that there is mix of page space usages with the average page usage in the 70 to 80% range.

Journal Transaction Level

The downside of selecting a larger block size for the journal files is that partially filled blocks waste disk space and increase the number of times the CV journals fill in the course of a day resulting in more frequent journal swaps. An increased number of swaps will also mean that more journal archive files are created which will result in more files having to be maintained and processed during manual recovery operations.

To maximize the space in a journal block CA IDMS provides a means to defer writing journal blocks called JOURNAL TRANSACTION LEVEL. The purpose of deferring the journal write is to enable additional images for other update transactions to be placed in the journal block before it is written, hopefully resulting in full or nearly full journal blocks. When this is achieved fewer journal writes are issued, disk space is better utilized, and fewer journal swaps occur.

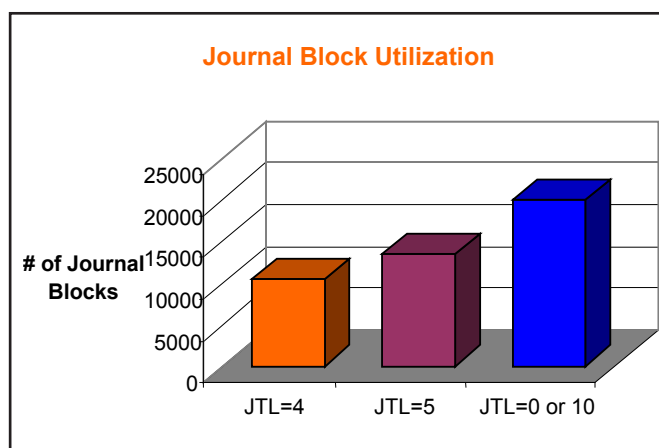
Journal transaction level can be set by specifying the parameter on the sysgen SYSTEM statement or by issuing the DCMT VARY JOURNAL TRANSACTION LEVEL command. The value set should be either a 0 which disables the functionality or a value of 3 or larger. Setting a value that is too small or too large may prevent a CV from getting the full benefit of the function or may completely eliminate any potential benefit. As long as the number of active update transactions in the CV exceeds the specified value, the write of a journal block is deferred until the block is full.

After establishing or changing a journal transaction level the space utilization histogram from several ARCHIVE JOURNAL runs should be reviewed to see if the change has had an impact and whether that impact is positive or negative. The average response time of update transactions should also be monitored. **When a transaction's journal write is deferred that transaction**

is placed in a wait state until the journal block is filled and the I/O completed.

If there are sufficient update transactions active but the amount of updating is low it is possible that the use of a journal transaction level may increase the amount of wait time for a transaction to levels greater than those experienced without the feature. In those cases it may be more desirable to accept a less efficient usage of disk space for faster throughput.

The graph in Figure 2 shows the number of journal blocks that were written to perform 20000 debit/credit transactions at varying journal transaction levels. In this example the most benefit was derived by setting the journal transaction level to 4. More than 50% fewer journal blocks were written to perform the same volume of work compared to what was required when the journal transaction level was set to 0 or 10. A value of 0 disabled the journal transaction level while a value of 10 resulted in an environment where there were rarely enough active update transactions to cause the writing of journal blocks to be deferred.

*Figure 2*

The chart in Figure 3 illustrates the effect that Journal Transaction Level has on how full the journal blocks were when written. With a journal transaction level of 4, about 50% of the journal blocks were almost or completely full versus 0% when the journal transaction level was set to a

(continued on page 11)

WWW.IUASSN.ORG
YOUR PORTAL TO IUA
SERVICES AND IDMS
CONTACTS

value of 0 or 10. Conversely, with a journal transaction level of 4, less than 20% of the blocks had a utilization of less than 10% versus about 50% of the blocks with a journal transaction level of 0 or 10.

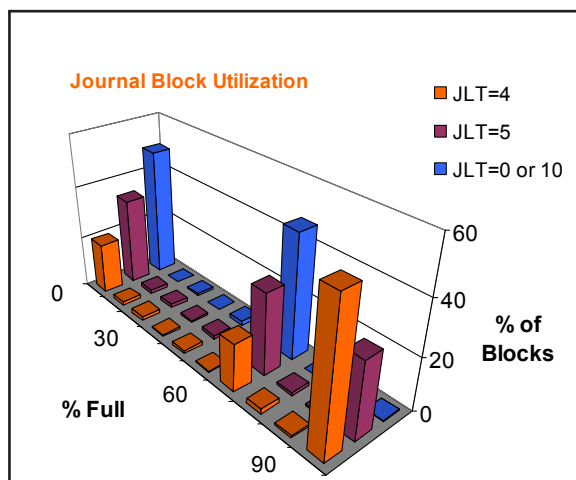


Figure 3

Journal Buffer Size

To allow for journal images to be created while earlier blocks are being written a pool of buffer pages is maintained by CA IDMS for the journal environment. Unlike a database buffer pool which is accessed randomly, the journal buffer pool is accessed circularly. When a journal buffer needs to be written its I/O is started and the next journal buffer becomes active. This reduces or eliminates waits for a journal buffer when a write to a journal file is required.

When the I/O is completed the buffer remains as is until it becomes active again and the previous journal images are overlaid. Multiple journal buffer pages may also reduce journal I/O during recovery by allowing a BFOR image to reside in memory longer. If a recovery operation must read in a journal page in order to access a BFOR image, it will look in the non-active buffers for the BFOR image before accessing the journal.

A journal buffer should minimally contain 5 pages and may benefit from many more pages in active systems that incur many abends that lead to database recovery. The adequacy of the journal buffer pool can be monitored by issuing the DCMT DISPLAY BUFFER command.

- Journal Buffer -	Size	# In-Use	Waits	DB	Ckpt
JNL_BUFFER	2004	5	0	0	23

# of Recoveries	I/O's	in Buffer
0	0	0

Figure 4

In almost every CV the most important statistic in Figure 4 is the number of waits. The number of waits should always be zero. When an I/O is started on a journal buffer the buffer is flagged as unavailable and the next buffer becomes the active journal buffer. Upon completion of the I/O the buffer being written again becomes available. Many journal buffers may be unavailable at any one time because they are all concurrently in an I/O state.

If all of the buffers in the journal buffer pool are unavailable the next request for a journal buffer will add to the wait count. Therefore if the wait count is non-zero it means that all of the buffers in the pool were in a wait state at the same time and all transactions had to wait for a journal buffer to perform an update operation.

When the wait count is greater than zero the number of pages within the journal buffer should be increased until this number is once again at zero. Increases in the pool size should be made in small increments as a small change can have a visible change in the number of waits encountered.

In a few CVs that experience high numbers of recovery operations there may be some value in defining a large journal buffer pool. The '# of Recoveries' in Figure 4 represents the number of aborted transactions that had to roll out updates and 'I/Os' are the number of journal blocks that were read to perform the recoveries. The 'in Buffer' value is the number of journal blocks that were found within the journal buffers and therefore did not require an I/O operation to be performed.

Recovery will not use the CV's journal buffer pool to read the journal but maintains its own buffer consisting of a single page. Recovery will search the CV's journal buffer pool to locate a required BFOR image and if found copies that buffer's contents into its own buffer causing the 'in Buffer' value to be incremented. If a CV is very active and is writing journal images at a high rate the BFOR images for an aborted transaction may not be in the journal buffers by the time the recovery starts. This is especially true for abends resulting from deadlocks and timeouts. If the CV does a large number of recovery operations increasing the journal buffer pool to a large number such as 300 or 400 may result in an increase in the value of the 'in Buffer' statistic indicating a more efficient recovery environment. However if the 'in Buffer' value remains small after making the increase the enlarged buffer pool is probably just wasting storage space and should be reduced to the minimum level where journal waits will be maintained as zero.

Summary

The journal sizing considerations presented should be reviewed whenever the processing characteristics of a CA IDMS CV change. The change may be due to the addition of more users to the environment or the addition/modification/deletion of applications controlled by the CV. Pro-active monitoring will help to insure that a CV's journal environment maximizes the available resources and minimizes journal I/O and the potential associated processing bottlenecks.

Gilbert Jonswold: *I started with Cullinet in 1979, working in support on IDMS. In January of 1981, I became manager of IDMS/DC support. Somewhere around 1983, I joined development, and re-designed and developed the journal system and the buffer management system. After CA bought Cullinet, I moved back into support, where I've been part of the IDMS/DB support team.*

WE ARE 15 GOING ON 17 ...

by Chris Hoelscher

Sorry – not a revival of *The Sound of Music*. Rather, how I have been keeping busy at work. After much reading, experimenting, and opening issues with CA, I am ready to go live with Release 17. Following are a list of “challenges” I experienced, reading them here may hopefully allow you to avoid them or deal with them more gracefully than did I. CAVEAT – we do not license the SQL option, so we may have avoided much of the upgrade efforts that other sites will experience.

New Delivery System – Find the Document *Electronic Software Delivery* from CA – it will guide you through the new process:

- I was not authorized to create a directory to hold the pax.Z file – needed help from our system software group
- If you can download the pax.Z file directly to your mainframe, doing so will save a step (and some time)
- Again, I required assistance from our system software group in defining directories and permissions to unpack the pax.Z files

Read the Manuals

- The *R16 Release Summary*, *R17 Release Summary*, *Installation and Maintenance Guide*, and *Systems Operations* manuals provide invaluable information regarding how to install system software and site customizations; however they not always agree with each other, nor at times do they agree with themselves (SMP/E vs non-SMP/E install). If you believe you see a discrepancy, do not hesitate to contact CA (check the PEAs/PIBs or open an issue).

Surprise!

- Release 17 requires that each environment run with an LE runtime loadlib (as opposed to a COBOL II or previous runtime loadlib). Maybe we were the only shop left in IDMS land that was running with COBOL II runtime loadlibs, but we were. We promoted our COBOL runtime to LE before the end of 2008
- No more startup module! Once I created a USERMOD for WTOEXIT, I immediately jumped on board with this feature to make our systems more easily modifiable.
- SVC internal format changed – we have a user program that walks the SVC module in memory looking for which CVs are UP (please do not ask why). The offset for these “up” slots changes, so I had to modify the user program when the new SVC went in, NOT when the new release of IDMS went in. (to be fair, this offset has changed for previous SVC versions; this is my first install for this division of our company in which the offset DID change)
- RHDCSSFM – since there is a new version of RHDCSSFM for release 17, but it is names the same as the release 15 version, we were in quandary if reloading SVCs while IDMS CVs were up would also apply to RHDCSSFM (CA provides for differently named (actually numbered) SVCs to allow

staging for different versions of IDMS SVCs within the same LPAR, but not so with RHDCSSFM. Since we had no time when all IDMS CVs were done on an LPAR at the same time, we were forced to load RHDCSSFM with the TEST SVC and the prod CVs up. This did not cause any problems, but I am not sure whether this was due to luck or design

- The loadlib from which certain system software modules are loaded MUST be defined as APF-authorized in order to exploit zIIP. This does not mean, however, that the entire STEPLIB or CDMSLIB concatenation need be authorized. Additionally, it was at one time required (and documented) that the load library from which RHDCUXIT is loaded also be APF-authorized. However, I realized that this library was not APF-authorized. And yet my sandbox Release 17 CV came up zIIP eligible. CA then confirmed that the requirement has been relaxed for RHDCUXIT.
- I attempted to load/execute my IDMS system software from a PDS/E. This worked as designed everywhere ... except in my TSO interface CLIST I was getting 3907 error code <?> messages. CA offered what were most likely workable solutions, but I decided to revert to PDS loadlibs and fight this battle another day

Other thoughts

- HPSPO – I am adding /modifying storage pools (both XA and non-XA) to isolate User, User-Kept, Shared, and Shared-Kept from all other storage types, and am altering my Storage key to 9 to support the High-Performance Storage Protection Option. The **Release Summary** suggests this might be done in production environments only, but I would not want to set this up in production without previewing in a non-production environment
- CONVERT CATALOG – although the manual does not explicitly state this, the CONVERT CATALOG process will fail for a non-sql shop (I had thought that due to internal use of the SQL engine and other features (Visual DBA?) the relational view of the CATALOG might be required, and therefore need upgrading. Not only is this not necessary, it will fail.
- Apply maintenance frequently – while in pre-production, I have been checking weekly for release 17 maintenance. In the four months of general availability, I have seen (what I perceive to be) relatively few fixes issued. While this might be due to a limited user base testing the software, I would rather attribute this to a solid IDMS release.

What's next?

- I will be upgrading my non-production environments to release 17 on March 1st, and my production environments on March 29th. Stay tuned

“WHY IS MY LOAD AREA SO FULL?”

By Gary Cherlet

For years our site kept all of the Dictionary (IDD) Source and Load areas (DDLDM and DDLDCLOD respectively) the same size in all environments. This was particularly useful if we ever wanted to “synchronise” two environments without having to first identify all the differences and then perform a massive migration. All we had to do was physically copy one IDD from one environment over the IDD in another environment.

Then one day one of our ever vigilant DBA’s noticed that while the Production IDD Load Area was only 53% full, the Development IDD was over 80% full. So the DBA’s asked, “Why is this so?”

I thought there must be a simple explanation, so I thought that I would determine the distribution of Record types in the Load Area by running the command:

```
PRINT SPACE FOR AREA JISLOD.DDLDCLOD FULL;
```

through Command Facility, for our Production IDD. The following screen shot shows you the results of this exercise. Notice Space Available of 47.25% near the top of the report, and that SR157’s take up 94.34% of the space used. While “Space Available” may mean something to most of us, the predominance of SR157’s may not.

Total Space Allocated	408,482,480	
Total Space Available (Percent)	193,010,968	(47.25%)
Total Space Used	215,471,437	
Logically Full Pages	0	
Total Space Unusable (Percent)	0	(0.00%)
AREA JISLOD.DDLDCLOD	Distribution of USED Space Report	
Record Type	Maximum	Percent of
Null Line	8	0.56
SR4	524	0.07
SR155	28	0.00
SR156	104	0.00
SR157	524	94.34
SR158	532	1.09
SR171	84	0.00
SR172	96	0.00
SR174	80	0.00
SR175	524	3.01
SR176	104	0.00
Space Inv.	23,444	0.02

Figure 1

For purposes of this exercise it is necessary to go to your “Advantage CA-IDMS Dictionary Diagram”. Don’t be intimidated by it, we are just going to focus on the cluster of records in the lower left hand part of the diagram. Notice that taken together they represent what we have come to know and love as “IDD Dictionary Load Modules”. They usually occur in the DDLDCLOD for ADS and IDMS-DC components: dialogs, maps, subschemas, and IDD Tables. There is an analogous structure that can also contain SQL components such as Access Modules, which are stored in the DDLDCATLOD area.

In the table below I have provided a list of the most common System records (SR’s) in the DDLDCLOD / DDLDCATLOD area(s):

SR	Record Name	Description
SR155	LOOAK-155	Load Area “One of a Kind”
SR156	LOADHDR-156	Load Module “header” - owner of “module text” and “symbol text” (when Symbol Tables are on In ADSC)
SR157	LOADTEXT-157	What gets loaded and executed at run time - this is what you get when you “PUNCH LOAD MODULE WITH SYNTAX.” in IDD - you can LINK this into a Load Library
SR174	SYMHDR-174	The owner of “symbol text” for ADSC Symbol Tables
SR175	SYMTEXT-175	The actual “Symbol Text” for Symbol Tables

Table 1

Total Space Allocated	798,184,080	
Total Space Available (Percent)	758,155,936	(31.34%)
Total Space Used	548,028,064	
Logically Full Pages	0	
Total Space Unusable (Percent)	0	(0.00%)
AREA JISLOD.DDLDCLOD	Distribution of USED Space Report	
Record Type	Maximum	Percent of
Null Line	8	0.25
SR4	524	0.06
SR155	28	0.00
SR156	104	0.28
SR157	524	52.14
SR158	532	0.44
SR171	84	0.09
SR172	96	0.15
SR174	80	0.07
SR175	524	46.20
SR176	280	0.07
Space Inv.	23,444	0.01

Figure 2

Now with this knowledge let’s examine the preceding screen shots. We see that 94.34% of the Space Used in Production are SR175’s, that is Load Module Text - but in Development this is only 52.14% of the Space Used, the other 48% being made up primarily of “Symbol Table Text”, the SR175’s. Let’s “drill down” to make sure we fully understand what is happening and how these reports help to explain it away.

SR	Description	PRO Occ	PRO Occ	PRO %	DEV %
SR156	Load Header	12,502	12,502	0.60	0.28
SR157	Load Text	387,959	387,959	94.34	52.14
SR174	Sym Header	108	108	0.00	0.07
SR175	Sym Text	12,395	12,395	3.01	46.20

Table 2

The difference of 3,000 Load Modules between PRO and DEV reflects a lot of “experimental” and “testing” type of dialogs and maps, as well as (in our case) some significant developments that have yet to be either migrated or cleaned up.

The difference of 5,000 Symbol Text Headers reflects the fact that we, generally, turn Symbol Tables “OFF” as part of our migration process - but that at any given time we might have Symbol Tables “ON” to help in debugging

(continued on page 14)

Production problems.

Notice the “Total Space Allocated in PRO is 408Mb, compared to 798Mb in DEV (which is why we are now 64% full instead of >80%). This is because realising the situation, and that it is a fact of life that developers will use Symbol Tables in Development, we expanded the area before we filled it up. A case of “forewarned is forearmed”. I hope that this article will provide you with some useful information the next time that you run a Print Space against your Dictionary Load Area.

In a future article, if there is any interest, we might apply this technique to the DDLML area, the IDD Source area to see what interesting things we might be able to learn from a simple little Print Space report. If you're interested please let us know. If you found this article helpful - we would like to know that as well.

Gary Cherlet
Mainframe Technical Support Analyst
Justice Technology Services
Attorney General's Department, South Australia

PREPARING CA IDMS CV FOR EXITING AND ENTERING DAYLIGHT SAVING TIME

By Dave Pomeroy

With the need for uninterrupted processing for CA IDMS systems, the time change requirements resulting from daylight saving time can pose a legitimate challenge for even the most experienced workload scheduler. Over the years and various releases of CA IDMS, there has been some confusion on the impact of daylight saving time on the CA IDMS logs and journals and database integrity.

However, with proper planning, the information from CA IDMS Product Alerts and the installation of CA IDMS program and task IDMSDST (described below), the impact should be minimal. Note: Installation of IDMSDST applies to z/OS systems only.

In 2009, daylight saving time will begin on March 8 and will end on November 1. So as you are reading this, you should be taking the steps necessary to provide a forward transition for your CA IDMS systems to daylight saving time on March 8, 2009.

According to CA IDMS PEA **QI52458**, the following steps are necessary for CA IDMS systems on all operating systems when exiting daylight saving time:

Use the following steps to prevent journal corruption and date/time stamp errors with maps.

1. Shutdown the CA IDMS CV.
2. Archive any full and the active journal files.
3. Format the journal files.
4. Offload the CA IDMS log file.
5. Format the CA IDMS log file.
6. Perform the machine's date/time change operation.
7. Start up the CA IDMS CV (the TOD clock will now

be in synch with the regressed time from the system clock).

8. Disable the MAPC task and RHDCMPUT program for one hour to avoid date/time stamp errors with maps

However, if the CV was not shutdown during the change to standard time, the CA IDMS time management routines will recognize the backward change in the system clock and will not update the local time of day clock. This will prevent damage to the journals and maps relying on a date/time stamp. Also, if the task/program IDMSDST is installed and enabled, then the following warning message is issued to the CA IDMS log: DC998006 current time julian date TOD clock will not be updated CV shutdown is recommended!

Be aware, however, as indicated in QI52458, if CA IDMS is active during a backward time change, some tasks may stall unexpectedly and timer initiated tasks may have unanticipated start times.

When we enter daylight saving time again on March 8, 2009, however, there is no need to shutdown the CA IDMS CV on z/OS or BS2000 operating systems during the one hour advance of the system clock from standard time to daylight saving time.

On z/OS systems, the system clock is normally updated by the MVS SET CLOCK command or sysplex timer. On BS2000 systems, the system clock update is handled in the GTIME section of the system startup parameter file. If the IDMSDST program/task has been installed on z/OS systems and is enabled, then the following notification message is issued to the CA IDMS log:

DC998005 current time julian date TOD clock has been updated to match the advance in system clock.

As PEA QI52458 indicates during CA IDMS CV startup, the difference between the local time and GMT time (in binary units of 1.048576 seconds) is saved by the CA IDMS time management routines. For each time request, the current difference between the local time and GMT time is compared with the “saved” difference between these 2 values. If there is a positive difference (indicating that the system clock was advanced), the CA IDMS time management routines will update the local TOD clock with the new clock time and save the new difference between the local time and GMT time. This avoids the necessity of cycling the CV.

Care should be taken when changing the system clock while CV is up (z/OS), if the time is advanced too far and then corrected to adjust the system clock back to the correct time, then the TOD clock will be adjusted with the first time change but not the second (the difference between the first time change and the second time change will be negative).

On z/VSE or z/VM operating systems which require an IPL to update the system clock, the CA IDMS CV should be shutdown before the system IPL. Once the system clock has been advanced the system can be IPLed and the CA IDMS CV's local TOD will reflect the time change.

(continued on page 15)

Function and installation of CA IDMS program/task IDMSDST (z/OS systems only)

CA program and task IDMSDST is a message generation program which will issue a #WTL when there is an update of the system clock as determined by the time management routines in the CA IDMS module RHDCOESA executing on a z/OS operating system.

For Release 17.0 SP0 or higher, APAR RI05410 includes the user mode program IDMSDST.

For Release 17.0 there are no required zaps to implement IDMSDST.

For Release 16.0 SP2 or higher, APAR QO70340 includes the user mode program IDMSDST and the zaps required to implement IDMSDST.

For Release 16.0 SP1, APAR QO70338 includes the user mode program IDMSDST and the zaps required to implement IDMSDST.

For Release 15.0, APAR QO71179 includes the user mode program IDMSDST and the zaps required to implement IDMSDST.

IDMSDST performs the #WTL of either of two messages

When entering DST: **DC998005** current time julian date TOD clock has been updated to match advance in system clock.

When exiting DST: **DC998006** current time julian date TOD clock will not be updated. CV shutdown is recommended!

Program IDMSDST and task IDMSDST need to be defined in the system dictionary. Sample definitions are also included in the apar. Program IDMSDST can be modified as needed.

Each version of the apar requires the pre-compile, assemble and link of IDMSDST.

Dave Pomeroy is a Senior Sustaining Engineer for CA IDMS Level II Support located in Framingham, MA. Dave joined Cullinet Corp. in March 1985. Since then, he has provided support for CA IDMS/DC on z/OS and z/VM operating systems.

IDMS-L WHERE IDMS TECHIES MEET

IS IT OK IF I SIGNON? IF I DO, “WHAT AM I AGREEING TO?” OR - YOU MAY NEVER WANT TO LOGON AGAIN !

Gary Cherlet

Figure 1 below shows the content of our “Netmaster Solicitor Screen”, known by most people as the “signon” or “Login” screen. I know that ours will differ from many, but in these days of increased security and “conformance requirements” I would expect that most such screens around the world have a similar “look and feel” and make similar declarations on the part of the employer and the employee - so here is an sample “Signon screen” which I will break down into it’s key elements for purposes of discussion

```
13.59 03/02/2009      JUSTICE INFORMATION SYSTEM      TTCPJ088
                     HOST LOGON BANNER For PRODUCTION SYSTEM      MAINPANL
z/OS  01.09.00 -----
                        ** CONDITIONS OF USE **
The information from this system is CONFIDENTIAL and must not be disclosed to
unauthorised persons under any circumstances, nor are you authorised to access
such information for personal and/or unauthorised training purposes.
Unauthorised access to or use of this system and of the information contained
within may result in criminal charges and/or disciplinary action in accordance
with relevant employment legislation.
Users are reminded that they are responsible for all system access by their
userid. To protect your userid from unauthorised use you are required to logoff
at the completion of tasks requiring online access.
Users must logon using their personally issued userid and a confidential
password. The use of another person's userid is strictly prohibited.
All use of the system is logged and subject to audit.

By entering my  USERID   :
                and PASSWORD :      I acknowledge the conditions of use.
                SYSTEM NAME :      (Optional)
```

Figure 1

I don’t know how often people actually look at the “Conditions of Use” in the contents of the “login” screen as they sign on to the mainframe or the LAN - so I thought I’d take the opportunity to point out a couple of the more interesting bits to consider the next time you do sign on.

Let’s start off with the part of the signon screen that we all use the most:

```
By entering my  USERID : _____
                and PASSWORD : _____
                I acknowledge the conditions of use.
```

You’ll notice that I have reformatted the text - when looked at in this way it appears to be a “statutory declaration”, that is a legally binding acknowledgement on your part - and indeed, each time you sign on you are in fact supplying a “digital signature” of sorts to state that you accept the “conditions of use”.

(continued on page 16)

I think that is pretty clear - are you concerned yet about what you have agreed to? No - well maybe you should be. Let's look at some of the other content of the agreement you have just made when you signed on to the system. I particularly like this part:

The information from this system is
CONFIDENTIAL and must not be disclosed to
unauthorised persons
under any circumstances...

How can you tell if a person is 'authorised' or not?", I hear you ask. Well it's actually quite easy - if a person is authorised they will have the ability to signon to the system and access the information for themselves. Stated quite simply then, if you use your access to the system to acquire information for the benefit of a person who is not authorised to access it themselves - then you have not only just broken the terms of the agreement, you have in fact committed a criminal act and that, as the banner goes on to mention:

.... may result in criminal charges and/or
disciplinary action in accordance with
relevant employment legislation.

Many people are aware of the "enquiry logging" capability that is built into JIS applications that reveal sensitive through to highly classified information. The fact that "use of the system is logged and subject to audit" is clearly stated in the line immediately above your "signature" (your User Id and Password). A similar logging capability for file level accesses is active in both the mainframe and LAN environments.

Remember that I am writing from an Australian perspective, where we don't have the same level of legislative requirements for audit/logs as has become required through SOX legislation in the USA. So I fully expect that similar statements appear on most user's screens in the US, as well as other parts of the world.

All use of the system is logged and
subject to audit.

That's fair enough - but what about data that does not have this application built in logging and audit trail? What about accessing the same data through system utilities or some special privileges that might appear to bypass the logging?

You should be aware that in the case of (JIS) data contained in IDMS or Oracle databases, it is possible for us to turn on logging for sensitive or critical data inside the database itself - without this being done explicitly by the application programmer. Similarly, file level accesses performed by users with special privileges can also be logged.

So - if you don't want to run the risk of having your "digital fingerprints" on some "digital data" - then you should not use either applications, system tools or special privileges to even "look at" data on another person's behalf.

Forget about "buyer beware" - when using the JIS systems and JTS LAN you should be thinking in terms of "viewer beware"! We are all well aware of times when we may have tried to do something for somebody in an effort to "be helpful" - but the next time you are trying to help somebody out by accessing some information for them - stop and think about whether what you are about to do might be a criminal offence!

SECURING CA IDMS TCP/IP SOCKETS USING AT-TLS ON Z/OS

By Dave Pomeroy

In release 16.0, CA IDMS began exploiting TCP/IP as an industry standard communications protocol. As with most communications protocols, securing the data exchange between client and listener sockets (i.e. endpoints) is a priority issue.

On the z/OS mainframe, prior to z/OS 1.7, socket security for TCP/IP applications is provided by TLS (Transport Layer Security) and/or SSL (Secured Socket Layer). However, using TLS and SSL usually requires significant application coding and complex client and listener negotiations.

With the introduction of the AT-TLS (Application Transparent Transport Layer) feature in z/OS, CA IDMS inbound and outbound TCP/IP sockets can be secured using AT-TLS without requiring program changes for basic CA IDMS TCP/IP applications. AT-TLS invokes Secure Sockets Layer (SSL) and/or Transport Layer Security (TLS) to provide policy defined security.

AT-TLS is policy driven. The policy establishes rules for the inbound and outbound TCP/IP traffic. An AT-TLS policy contains three areas of rules:

- 1) Traffic Descriptors which describe the properties of the traffic (i.e. ports, protocols, direction, etc).
- 2) Security Levels which define security ciphers used for protection
- 3) Requirement Maps which map the Traffic Descriptors to the security levels (i.e. Ciphers). A cipher is an algorithm used to encrypt and decrypt data. TLS and SSL provide cipher suites which provide different levels of security.

Once the digital certificates are defined and the AT-TLS policy is configured, then the CA IDMS client can attempt a

connection to the CA IDMS listener. If accepted, the CA IDMS listener responds with unencrypted data and TCP/IP queues the CA IDMS listener response data. AT-TLS then causes a SSL handshake to occur using the CA IDMS listener's identity. If the handshake proceeds normally (i.e. digital certificates verify), then SSL encrypts the CA IDMS listener's data and sends it to the CA IDMS client socket. The CA IDMS client will then send encrypted data which causes AT-TLS to invoke SSL again to decrypt the encrypted CA IDMS client data and the CA IDMS listener receives the decrypted CA IDMS client data. This is accomplished without any application program coding changes in CA IDMS.

The time spent configuring AT-TLS to secure CA IDMS sockets, is not too cumbersome and well worth the effort. However, before attempting an AT-TLS configuration of CA IDMS TCP/IP sockets, it is highly recommended reading the IBM documentation: z/OS Communication Server

IP Configuration Guide Application Transparent Transport Layer Security (AT-TLS).

With the information provided in the IBM AT-TLS documentation, you should be able to configure a secure client and listener socket within a CA IDMS TCP/IP environment.

The following items can be used as a reference to secure a CA IDMS client and listener socket using AT-TLS:

Item 1: Generating digital certificates and keyrings for the secure handshake

As indicated by the IBM documentation, security provided by AT-TLS is done at the TCP/IP stack (i.e. transport) level. However, during certificate authentication, the application must present a valid digital certificate (in a keyring) for the secure handshake. To begin the AT-TLS setup, client and listener keyrings (collection of digital certificates) need to be created in a security database (e.g. CA Top Secret). The listener keyring contains a listener certificate and any certificate used to sign it. The client keyring contains the local Certificate Authority (CA) certificate used to sign the listener certificate.

There are two methods available to create the digital certificates for the client and listener keyrings:

Method 1: Using the gskkyman utility - menu driven z/OS shell based program

The gskkyman utility is documented in z/OS System SSL Programming. It allows for the creation of certificates in PKCS#12 format which can be exported and imported to a CA-Top Secret keyring or JAVA keystore in X.509 format.

To execute gskkyman successfully you need superuser authority within OMVS. Before starting gskkyman set the current directory to a directory which will contain the created digital certificates.

See "z/OS System SSL Programming" for the directory location of gskkyman and message catalogs. Normally, the following path can be used to start gskkyman in OMVS:

/usr/lpp/gskssl/bin/gskkyman

Example: gskkyman utility to open existing certificate database IDMSKEYS which was created using gskkyman (see Create new database Option 1 for details on creating a certificate database).

Note: current directory was set to u/certs/idms/attls

(continued on page 18)

**CHECK OUT THE IUA
ARCHIVE LIBRARY OF
IDMS PRESENTATIONS**

WWW.IUASSN.ORG
**YOUR PORTAL TO IUA
SERVICES AND IDMS
CONTACTS**

Database Menu

- 1 - Create new database
- 2 - Open database
- 3 - Change database password
- 4 - Change database record length
- 5 - Delete database
- 6 - Create key parameter file

0 - Exit program

Enter option number

==> 2

Enter key database name (press ENTER to return to menu): IDMSKEYS

Enter database password (press ENTER to return to menu): IDMSKEYS

Key Management Menu

Database: /u/certs/idms/attls/IDMSKEYS

- 1 - Manage keys and certificates
- 2 - Manage certificates
- 3 - Manage certificate requests
- 4 - Create new certificate request
- 5 - Receive requested certificate or a renewal certificate
- 6 - Create a self-signed certificate
- 7 - Import a certificate
- 8 - Import a certificate and a private key
- 9 - Show the default key
- 10 - Store database password
- 11 - Show database record length

0 - Exit program

Enter option number (press ENTER to return to previous menu):

==>

You can then enter options to manage keys, certificates, etc.

Method 2: Using CA-Top Secret commands executed via batch jobs to create digital certificates and keyrings.

```
/******
/* Create local certificate authority certificate          ***
/* Owner will be CERTAUTH userid                        ***
/* Cannot run this job under MASTER id                  ***
/* You need Top Secret MISC4 authority to run this job   ***
/******
//S1      EXEC   PGM=IKJEFT01
//SYSUADS  DD DSN=SYS1.UADS,DISP=SHR
//SYSLBC   DD DSN=SYS1.BROADCAST,DISP=SHR
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN  DD *
TSS GENCERT(CERTAUTH) -
DIGICERT(IDMSAUTH) SUBJECTN('CN="IDMSAUTH" -
T="title" -
OU="CA-IDMS" -
O="TCP/IP Support" -
L="Yourtown" -
ST="IA" -
C="US"') LABELCERT(IDMSAUTH) USAGE(CERTSIGN)
/*
```

(continued on page 19)

Securing CA IDMS TCP/IP sockets.. cont'd from page 17

```
//*****
//* Create listener certificate ***
//* Owner is IDMSUSER ***
//* You need Top Secret MISC4 authority to run this job ***
//*****
//S1      EXEC   PGM=IKJEFT01
//SYSUADS DD DSN=SYS1.UADS,DISP=SHR
//SYSLBBC DD DSN=SYS1.BROADCAST,DISP=SHR
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
TSS GENCERT(IDMSUSER) -
DIGICERT(IDMSLISN) SUBJECTN('CN="IDMSLISN" -
T="IDMS listener Cert" -
OU="CA-IDMS" -
O="TCP/IP Support" -
L="Yourtown" -
ST="IA" -
C="US") LABELCERT(IDMSLISN) SIGNWITH(CERTAUTH,IDMSAUTH)
/*

//*****
//* Create client certificate ***
//* Owner is IDMSUSER ***
//* You need Top Secret MISC4 authority to run this job ***
//*****
//S1      EXEC   PGM=IKJEFT01
//SYSUADS DD DSN=SYS1.UADS,DISP=SHR
//SYSLBBC DD DSN=SYS1.BROADCAST,DISP=SHR
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
TSS GENCERT(IDMSUSER) -
DIGICERT(IDMSCLT) SUBJECTN('CN="IDMSCLT" -
T="IDMS Client Cert" -
OU="CA-IDMS" -
O="TCP/IP Support" -
L="Yourtown" -
ST="IA" -
C="US") LABELCERT(IDMSCLT) SIGNWITH(CERTAUTH,IDMSAUTH)
/*

//*****
//* To create client and listener keyrings and ***
//* add the certificates to the keyrings. ***
//* You need Top Secret MISC4 authority to run this job ***
//*****
//S1      EXEC   PGM=IKJEFT01
//SYSUADS DD DSN=SYS1.UADS,DISP=SHR
//SYSLBBC DD DSN=SYS1.BROADCAST,DISP=SHR
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
TSS ADD(IDMSUSER) KEYRING(IDMSATLS)
TSS ADD(IDMSUSER) KEYRING(IDMSATLS) RINGDATA(CERTAUTH,IDMSAUTH) -
trust
TSS ADD(IDMSUSER) KEYRING(IDMSATLS) RINGDATA(IDMSUSER,IDMSLISN) -
default trust
TSS ADD(IDMSUSER) KEYRING(IDMSATLS) RINGDATA(IDMSUSER,IDMSCLT) -
trust
TSS ADD(IDMSUSER) KEYRING(IDMSCLKR)
TSS ADD(IDMSUSER) KEYRING(IDMSCLKR) RINGDATA(CERTAUTH,IDMSAUTH) -
default trust
TSS LIST(IDMSUSER) KEYRING(ALL)
/*

//*****
```

(continued on page 20)

Securing CA IDMS TCP/IP sockets.. cont'd from page 18

```
//To allow the ACID associated with the keyrings access to IBM ***
//facilities required to use AT-TLS. ***
//*****
//S1      EXEC   PGM=IKJEFT01
//SYSUADS DD DSN=SYS1.UADS,DISP=SHR
//SYSLBCL DD DSN=SYS1.BROADCAST,DISP=SHR
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD *
TSS PERMIT(IDMSUSER) IBMFAC(IRR.DIGT)
TSS PERMIT(IDMSUSER) IBMFAC(BPX.listener) ACCESS(ALL)
TSS PERMIT(IDMSUSER) IBMFAC(BPX.FILEATTR.PROGCTL) ACCESS(READ)
/*
```

Item 2: Creating the AT-TLS policy for CA IDMS and executing the Policy Agent (PAGENT)

Once the digital certificates for the client and listener have been defined then AT-TLS policies can be defined and loaded into the stack by the Policy Agent. PAGENT executes as a started task and is located in SYS1.PROCLIB(PAGENT). PAGENT may start automatically (after the TCPIP started task begins) if it is also defined in the list of autolog listeners in the TCP/IP profile dataset.

The PAGENT started task can also be started manually by issuing S PAGENT from the z/OS console.

The Policy Agent configuration file can be created using the z/OS Network Security Configuration Assistant for Windows which can be downloaded from IBM's support website. It is a very useful GUI interface. The configuration file can be FTPed to a HFS file (for example):

```
/u/certs/idms/attls/ATTLSSTCP.policy
```

and then it can be edited manually, if needed.

Using the z/OS Network Security Configuration Assistant for Windows is optional but highly recommended for its ease of use. The policy configuration file can also be created manually in a z/OS dataset or HFS file.

To disable or stop the PAGENT started task, enter STOP PAGENT from the z/OS console.

To update the Policy Agent with a modified policy, enter F PAGENT, REFRESH from the z/OS console.

Item 3: Enabling AT-TLS in the TCP/IP stack and starting AT-TLS

To enable AT-TLS include a TCPCONFIG TTLS statement in the TCPIP profile dataset. (Not all statements are shown). Sample statements:

```
TCPIPTLS.PROFILE.TCPIP dataset
...
DATASETPREFIX TCPIPTLS
...
; AUTOLOG the following listeners.
AUTOLOG 5
PAGENT JOBNAME PAGENT ; Policy agent AT-TLS
...
TCPCONFIG TTLS; enable AT-TLS
....
RESTRICTLOWPORTS; use ephemeral ports for AT-TLS testing
```

Sample statements:

```
TCPIPTLS.TCPIP.DATA dataset
...
; TCPIPJOBNAME specifies the name of the started procedure that was
; used to start the TCPIP address space. TCPIP is the default.
;
ATTLSSTCP TCPIPJOBNAME TCPIPTLS
DATASETPREFIX TCPIPTLS
```

To manually start the AT-TLS enabled TCP/IP stack in this example enter: S TCPIPTLS from the z/OS console.

Step 4: Stopping the Policy Agent and TCPIPTLS

(continued on page 21)

Stopping the Policy Agent: STOP PAGENT

```
10:44:06.92 STC00452 00000090 EZZ8445I PAGENT SHUTDOWN IN PROGRESS
10:44:06.99 STC00451 00000090 EZZ4249I TCPIPTLS INSTALLED TTLS POLICY
HAS NO RULES
10:44:07.02 STC00452 00000090 EZZ8773I PAGENT POLICIES PURGED
FOR TCPIP : TTLS
```

Stopping the TCP/IP AT-TLS enabled started task: P TCPIPTLS

Step 5: Sample AT-TLS policy configuration file for stack ATTLSTCP
(/u/certs/idms/attls/ATTLSTCP.policy)

```
##
## AT-TLS Policy Agent Configuration file for:
## Image: IDMS
## Stack: ATTLSTCP
##
## Created by the z/OS Network Security Configuration Assistant
## Date Created =
##
## Copyright = None
##
TTLSRule          ASMCLI~1
{
  LocalAddrSetRef   addr2
  RemoteAddrRef     addr1
  LocalPortRangeRef portR1
  RemotePortRangeRef portR2
  USERID           IDMSUSER
  Direction         Outbound
  Priority           253
  TTLSGroupActionRef enableGrpAct~IDMS
  TTLSEnvironmentActionRef eAct3~ASMCLI
}
TTLSRule          ASMLIS~2
{
  LocalAddrRef      addr1
  RemoteAddrSetRef  addr2
  LocalPortRangeRef portR2
  RemotePortRangeRef portR1
  Userid            IDMSUSER
  Direction         Inbound
  Priority           252
  TTLSGroupActionRef enableGrpAct~IDMS
  TTLSEnvironmentActionRef eAct2~ASMLIS
}
TTLSGroupAction    enableGrpAct~IDMS
{
  TTLSEnabled       On
  TTLSGroupAdvancedParmsRef gAdv~IDMS
}
TTLSGroupAdvancedParms gAdv~IDMS
{
  Envfile           DD:STDENV
}
TTLSEnvironmentAction eAct1~ASMCLI
{
  HandshakeRole      Client
  EnvironmentUserInstance 3
  TTLSKeyringParmsRef keyR2
  TTLSCipherParmsRef  cipher1~AT-TLS__Silver
```

(continued on page 22)

Securing CA IDMS TCP/IP sockets.. cont'd from page 20

```
TTLSEnvironmentAdvancedParmsRef eAdv1~ASMCLI
  CtraceClearText      On
  Trace                47
}
TTLSEnvironmentAction      eAct2~ASMLIS
{
  HandshakeRole          listener
  EnvironmentUserInstance 3
  TTLSKeyringParmsRef     keyR1
  TTLS cipherParmsRef     cipher1~AT-TLS__Silver
  TTLSEnvironmentAdvancedParmsRef eAdv2~ASMLIS
  CtraceClearText      On
  Trace                63
}
TTLSEnvironmentAdvancedParms eAdv1~ASMCLI
{
  HandshakeTimeout      60
}
TTLSEnvironmentAdvancedParms eAdv2~ASMLIS
{
  HandshakeTimeout      60
}
TTLSKeyringParms          keyR1
{
  Keyring                IDMSATLS
}
TTLSKeyringParms          keyR2
{
  Keyring                IDMSCLKR
}
TTLS cipherParms          cipher1~AT-TLS__Silver
{
  V3CipherSuites         TLS_RSA_WITH_DES_CBC_SHA
  V3CipherSuites         TLS_RSA_WITH_3DES_EDE_CBC_SHA
  V3CipherSuites         TLS_RSA_WITH_AES_128_CBC_SHA
}
TTLS cipherParms          cipher2~AT-TLS__Gold
{
  V3CipherSuites         TLS_RSA_WITH_3DES_EDE_CBC_SHA
  V3CipherSuites         TLS_RSA_WITH_AES_128_CBC_SHA
}
}
IpAddr                    addr1
{
  Addr                    911.555.000.12
}
IpAddrSet                  addr2
{
  Prefix                  0.0.0.0/0
}
PortRange                  portR1
{
  Port                    1024-65535
}
PortRange                  portR2
{
  Port                    40246
}
```

Dave Pomeroy is a Senior Sustaining Engineer for CA IDMS Level II Support located in Framingham, MA. Dave joined Cullinet Corp. in March 1985. Since then, he has provided support for CA IDMS/DC on z/OS and z/VM operating systems.



International Chair
 Terry Schwartz
 Company: Perot Systems
 Address: PO Box 269005
 Phone: 972 577-3722
 Email: terry.Schwartz@ps.net

Board Member
 Craig McGregor
 Axiom
craig.mcgregor@axiom.com

Board Member
 Diane Montstream
 Allen Systems Group
diane.montstream@asg.com



Secretary/Treasurer
 Email Coordinator
 Bob Wiklund
 Company: Tiburon Technologies
 Address: 17101 W. Gable End Lane,
 Surprise, AZ 85387
 Phone: 623 594-6022
 Email: bob.wiklund@tiburontech.com

Board Member
 Jan Rabaut
jan.rabaut@dexia.be

Editor
 Gary Cherlet
 Justice Technology Services
 South Australian Department of Justice
cherlet.gary@saugov.sa.gov.au



International Vice Chair
 Contributed Software Librarian
 Laura Rochon
 Company: Ajilon Professional Services
 Address: 22 Jolliet, St-Bruno,
 Quebec J3V 4Z1 Canada
 Phone: 514-943-8290
 Fax: 450 441-6880
 Email: l.rochon@videotron.ca

Desktop Publishing
 Rebecca Shaw 404 377-6982
shawrh@bellsouth.net

IUA Connections is a quarterly publication of the CA-IDMS Database and Applications User Association (IUA). It is designed to promote its members' objectives. *IUA Connections* is not responsible for the opinions expressed by its writers and editors.



European IUA Representative
 Steve Rundle
 Company: British Telecom BT Group plc.
 Address: PP2B33 Angel Centre,
 403 St. John Street, London
 EC1V 4PL UK
 Phone: +44 (0)20 7777 6920
 Fax: +44 (0)20 7777 6921
 Email: steve.rundle@bt.com

Information User Association
 401 N. Michigan Ave.
 Chicago, IL 60611-4267
 Phone: 312/321-6827
 Fax: 312/245-1081

Internet: iua@iuassn.org
 Web: <http://iuassn.org>



Board Member
 Linda J. Casey, PMP, CSM
 Company: Managing Member
 Run Right, LLC
 Email: lindajcasey@comcast.net