

# JOURNAL



**DB2 AND CICS ARE MOVING ON:**

## **Avoiding Potholes on the Yellow Brick Road to an LE Migration**

**BY M. CARL GEHR JR. & NATE MURPHY**



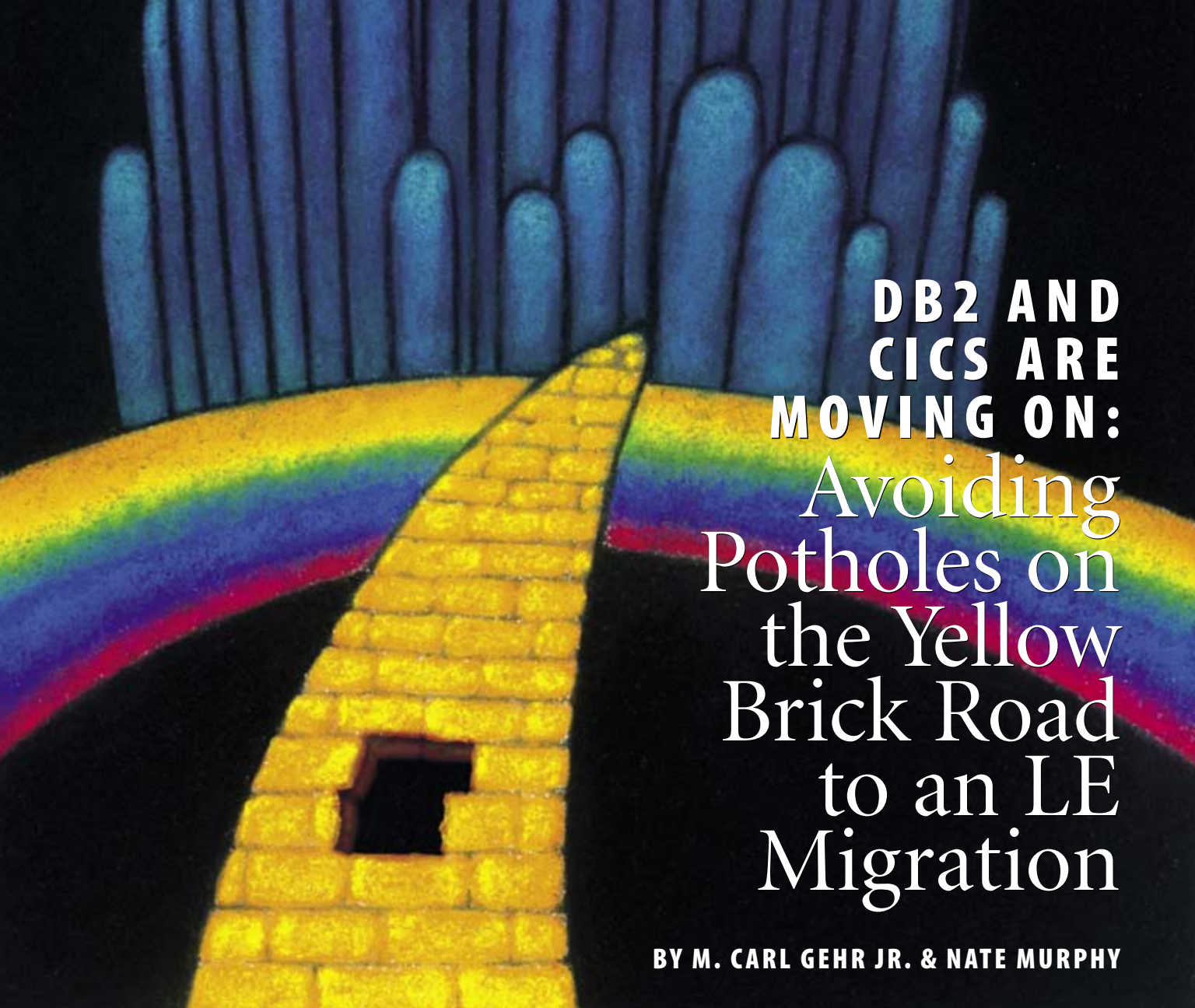
INSIDE 

**DB2 and CICS Are Moving On: Avoiding Potholes on the Yellow Brick Road to an LE Migration**

**The Trusted Enterprise as a Single, Secure Domain**

**MWLC for z/VSE**

**Migrating to Linux on System z**



# DB2 AND CICS ARE MOVING ON: Avoiding Potholes on the Yellow Brick Road to an LE Migration

BY M. CARL GEHR JR. & NATE MURPHY

For many years, IBM mainframe customers have ignored the warning shots of service withdrawal for various language compilers and run-times, especially COBOL and PL/I. “If it ain’t broke, don’t fix it,” became their mantra. However, with the announcement of CICS TS V3.1 and DB2 V8, customers who aren’t already running their applications under the Language Environment (LE) run-time library will need to perform a run-time migration, and probably compiler migrations, too. Starting with CICS TS V3.1, transactions compiled with OS/VS COBOL or earlier won’t execute with any run-time. This is the first time IBM has actually said, “In this environment, it will not run.” That might sound harsh, but IBM gave users adequate notice to avoid a crisis.

The prerequisite languages for DB2 V8 are the “currently supported” COBOL and PL/I compilers. Since these are all LE-conforming compilers, the LE run-time also is required. Not there yet? Then *two* migrations are required!

DB2 V8 has been available since March 2004. The CICS TS V2.2 announcement in December 2002 included the warning that OS/VS COBOL support would be removed in a future release. But, more than four years later, some shops are just realizing the implication of installing CICS TS V3.1, which became available in March 2005.

This is probably a clear indication of things to come regarding LE and LE-conforming compilers as prerequisites to new releases of other IBM software. There’s an old saying in regard to migrations, “If you wait until you must

migrate, you’ve waited too long.”

One side benefit of migrating to new platforms is that it enhances your ability to implement Service-Oriented Architecture (SOA). The main drivers for SOA adoption are that it links computational resources and promotes their reuse. Enterprise architects believe that SOA can help businesses respond more quickly and cost-effectively to changing market conditions.

DB2 and CICS provide a gold mine of application components that can contribute reusable resources in the development of SOA systems. It shouldn’t come as a surprise that both DB2 and CICS require LE-conforming compilers for full implementation. LE provides an efficient, effective environment for batch and online applications written in COBOL, PL/I, Assembler, and C/C++



without regard to SOA. Postponing or avoiding the migration to LE, the new compilers, DB2 V8/9 and CICS TS V3.1, hampers your ability to implement a robust SOA environment.

### CICS LE Migration Requirements

Over the past several years, CICS has provided a well-planned migration path to an LE-conforming environment, culminating with CICS TS V3.1. The LE migration path began with CICS V2.2, where the CICS translator was changed to no longer support old compilers. The LE run-time was strongly recommended, but not required. To use an older compiler, the translator from an earlier release of CICS could be used to compile programs using old compilers. CICS V2.3 required CICS programs to use the LE run-time library, but OS/VS COBOL programs were allowed to run if they ran under LE. CICS TS V3.1 requires that you convert and re-compile your OS/VS COBOL (and earlier COBOL) programs to at least VS COBOL II, but

the best alternative is to move directly to the Enterprise COBOL V3.4 compiler.

### DB2 LE Migration Requirements

DB2 V8 lets you run in V8 Compatibility Mode (CM) before migrating to V8 New Function Mode (NFM). You must migrate to CM before you can migrate to Enabling New Function Mode (ENFM) and finally to NFM. The run-time must be LE, but here's the catch: To use *any* of the new DB2 V8 functionality requires program changes, so you must re-compile your program with an LE-conforming compiler. You can use either the integrated coprocessor that's part of the enterprise compilers or the DB2 V8 pre-compiler.

Understanding the specific compiler requirements requires some interpretation of several terms. In the DB2 V8 announcement, IBM indicated the requirements for a "supported compiler" as the minimum level. At that time, the supported COBOL and PL/I compilers were:

- Enterprise COBOL for z/OS and OS/390 V3.2 or later
- IBM COBOL for OS/390 and VM V2.2
- Enterprise PL/I for z/OS and OS/390 V3.2 or later
- IBM PL/I for MVS and VM V1.1.

These supported compilers were those IBM still designated as "in service." Using the in-service guideline, the only COBOL compiler that's supported as of May 1, 2007, is Enterprise COBOL V3.4. Supported PL/I compilers are PL/I for MVS V1.1.1 and Enterprise PL/I V3.5 and V3.6. However, PL/I for MVS will be out of service after September 2009, so the target for any compiler migration to use with DB2 should be the latest Enterprise compilers; this will avoid an additional near-term migration. It's important that, in project planning, you ensure your LE migration team is in agreement on "supported levels" of compilers and other prerequisites. You also should verify that IBM Level 2 will respond to your calls and agree that your DB2 system is running in an IBM-supported environment.

IMS lacks any mandatory LE prerequisites that exclude the use of older run-times and compilers, but there are significant benefits to using LE for IMS applications and there are IMS considerations when moving applications to LE and the new compilers.

So, how do you get to LE and an LE-conforming compiler? The LE migration approach you select will depend on your unique requirements. There's no right or wrong way to determine when you need to re-compile your programs with an LE-conforming compiler, but not understanding what you need to do and how to do it can lead to wasted effort and possible delays in upgrading your applications to the new compilers and to the latest IBM middleware.

Let's consider some approaches to migrations and how some DB2 customers are working to become fully LE-conforming. Each approach has pros and cons. If you want the full benefits of DB2 V8, re-compile all your programs with a supported LE-conforming compiler. If the DB2 V8 functions aren't important to you yet, you can continue to run DB2 V8 in CM. Your choice to re-compile selected applications can be based on your need to use new DB2 V8 functions or when normal maintenance requires recompilation. By re-compiling with LE-conforming compilers whenever any change is required, you can

CSECTS CREATED BY COBOL	71
CSECTS USING ANSI V2 OR V3 COMPILER	2
CSECTS USING COBOL V4 COMPILER	0
CSECTS USING OS/VS COBOL COMPILER	27
CSECTS CREATED BY VS COBOL II	33
CSECTS CREATED BY VS COBOL II RLSE 1.0	0
CSECTS CREATED BY VS COBOL II RLSE 1.1	0
CSECTS CREATED BY VS COBOL II RLSE 2.0	6
CSECTS CREATED BY VS COBOL II RLSE 3.0	13
CSECTS CREATED BY VS COBOL II RLSE 3.E	6
CSECTS CREATED BY VS COBOL II RLSE 3.1	0
CSECTS CREATED BY VS COBOL II RLSE 3.2	0
CSECTS CREATED BY VS COBOL II RLSE 4.0	8
CSECTS CREATED BY COBOL/370 RLSE 1.0	1
CSECTS CREATED BY COBOL/370 RLSE 1.1	1
CSECTS CREATED BY COBOL FOR MVS V1 R2.0	1
CSECTS CREATED BY COBOL FOR MVS V1 R2.1	0
CSECTS CREATED BY COBOL FOR 390 V2 R1.0	2
CSECTS CREATED BY COBOL FOR 390 V2 R1.1	0
CSECTS CREATED BY COBOL FOR 390 V2 R1.2	0
CSECTS CREATED BY COBOL FOR 390 V2 R1.3	0
CSECTS CREATED BY COBOL FOR 390 V2 R2.0	0
CSECTS CREATED BY COBOL FOR 390 V2 R2.1	0
CSECTS CREATED BY COBOL FOR 390 V2 R2.2	0
CSECTS CREATED BY IBM ENT COBOL V3 R1.0	3
CSECTS CREATED BY IBM ENT COBOL V3 R1.1	0
CSECTS CREATED BY IBM ENT COBOL V3 R2.0	0
CSECTS CREATED BY IBM ENT COBOL V3 R2.1	0
CSECTS CREATED BY IBM ENT COBOL V3 R3.0	0
CSECTS CREATED BY IBM ENT COBOL V3 R3.1	0
CSECTS CREATED BY IBM ENT COBOL V3 R4.0	0
CSECTS CREATED BY IBM ENT COBOL V3 R4.1	0

Figure 1: What Is Your COBOL Application Inventory?

By avoiding the potholes in your LE migration plan, you position yourself to fully understand what must occur, explain why it's required, and decide when it needs to be completed.

minimize the testing required, since presumably, you'll test the changes you make even if they don't involve new CICS, DB2 or compiler functions. Normally, this also will provide the other benefits (i.e., performance, improved debugging or testing, resource utilization improvements, etc.) associated with the new compilers.

Additional questions to consider are:

- What price does a customer pay for running old, unsupported COBOL and PL/I compilers?
- What is the exposure if something fails?
- What if some business need requires a program to be changed? What are the implications?

Any of these could dictate an emergency migration. There also may be an unexpected performance penalty because the latest software levels aren't being utilized.

For example, a large Northeastern utility is approaching its migration as a staged conversion by using the new LE-conforming compilers only when normal program development requires a change. For them, this avoids the risk of an extended outage in the middle of the night. They want to avoid unexpected problems that could force programmers to tackle the unfamiliar task of quickly converting old PL/I and COBOL syntax to the required syntax for the enterprise compilers. This always seems to occur while the clock is ticking, the tight batch window is closing, and managers are asking why the outage is taking so long. For this utility, a clear understanding of its inventory and the relationships of all components in its applications is critical.

#### **The Yellow Brick Road**

Migrating before you're forced to is clearly a wise IT strategy, but maybe it's too late for that strategy. If you're already past the point where you can take a more leisurely view of migration, you may be facing increased technical risk or cost. You may have to extend your total migration schedule and may not be able to exploit the new functionality and performance features of the technology as quickly as you'd like.

So, what about users who now face an immediate LE migration to move to CICS TS V3.1 or DB2 V8? For them, some approaches are better than others; under pressure, none are perfect.

### Three LE Migration Potholes

**Migrating entire portfolio:** A mass migration has been used in a few cases, but you significantly increase your risks by attempting to migrate your entire portfolio simultaneously. Change management strategies almost universally emphasize making only one change at a time to allow for less complicated testing, backout, and recovery processes.

**Emulation:** Attempting to migrate by emulating an environment IBM no longer supports also raises the risk of

failure, and may hamper performance. This Band-Aid method hides the real wound and only delays the inevitable updates that really should occur.

**One step at a time:** In this approach, a CICS or DB2 customer would migrate the run-times and re-compile one program or application at a time with the LE-conforming compilers when required. Clearly, many customers will take this approach because of such factors as time, resources, cost, etc. The exposure here is that you must under-

stand all the inter-relationships and cross-talk among your applications to avoid introducing errors in the applications not yet migrated. This is probably the best approach as long as good planning and proper management techniques are followed. Certainly, it's the way to go when not under the duress of other pending migrations. You can reduce your risks by following the guidance below.

### The IBM Migration Guides

To understand the key phases of a migration project, refer to:

- *The IBM COBOL Migration Guide* (GC27-1409-05)
- *IBM PL/I Migration Guide* (GC27-1458-05)
- *IBM LE Run-Time Migration Guide* (GA22-7565-07).

These resources identify almost all the possible migration issues that might arise. If you can determine which of the several hundred pages in each guide apply to your situation, an LE or compiler migration will no longer be a major technical challenge.

Remember, too, that migration is no longer a one-time activity. It will be an ongoing part of each new release of the operating system and each new compiler release. Keeping up is more important than ever and several small steps are easier than one large one.

It's important to understand that what's often referred to as an LE migration actually consists of two separate migrations—the run-time migration and the compiler migration(s). Each has two individual steps.

Before you start, you should understand the concepts, functions, and features LE provides. In this educational effort, you'll see where the compilers and run-time fit into the process. The next phase involves assessment of the effort and planning the tasks to be performed. Both the run-time and compiler migrations require a comprehensive inventory of the languages and run-times your applications use. The assessment is what you do with that inventory.

The run-time migration consists of two sub-steps: First, you start running with the new dynamic run-time library (SCEERUN). Then, once that's stable, you start link-editing with the new run-time library static components (SCEELKED).

Once you have a stable run-time

**** MODULES WITH COBOL TO MIGRATE TO ENTERPRISE COBOL ****							
LOAD MODULE	PRE-				LE CONF		
	COBOL	ANSV4	ANSV4	OS/V5	COBOL	COBOL	LIBR
	CSECTS	COBOL	COBOL	COBOL	II	W/CMPR2	ID
AEDATECK	AEDATECK			X			TEST
	JGS95000				X		
AE110XLL	AE110					CMPR2	TEST
AE6992A	AE6992AB				X		TEST
	AE6992A				Re12		
	AEPR2				CMPR2		
AE998RB	CS998			X			TEST
BCBS01	CPF1453			LVL(1)			TEST
	GREGEDIT	X					
CCAXP1	PMMREPTA		X				TEST
**** MODULES WITH COBOL TO MIGRATE TO ENTERPRISE COBOL: SUMMARY ****							
TOTAL MODULES ANALYZED							21
TOTAL CSECTS ANALYZED							177
TOTAL OF ALL MODULES WITH COBOL CSECTS							12
TOTAL OF ALL COBOL CSECTS							34
TOTALS FOR MODULES AND CSECTS LISTED IN REPORT:							
NON LE CONFORMING COBOL NEEDING MIGRATION TO ENTERPRISE COBOL							
TOTAL MODULES							11
TOTAL CSECTS							34
CSECTS CREATED BY PRE-ANSV4 COBOL							2
CSECTS CREATED BY ANSV4 COBOL							5
CSECTS CREATED BY OSV5 COBOL WITH LANGLVL(1)							3
CSECTS CREATED BY OSV5 COBOL WITH LANGLVL(2)							4
CSECTS CREATED BY COBOL II RELEASE 2 OR EARLIER							1
CSECTS CREATED BY COBOL II WITH CMPR2							1
CSECTS CREATED BY COBOL II WITH NOCMPR2							12
LE CONFORMING COBOL NEEDING MIGRATION TO ENTERPRISE COBOL							
TOTAL MODULES							1
TOTAL CSECTS CREATED BY LE CONFORMING COBOL WITH CMPR2							6

Figure 2: Which Application Load Modules Have COBOL to Migrate?

environment, you can begin compiler migration by re-compiling with the new LE-conforming compilers. It isn't necessary to complete the entire run-time migration before starting the compiler migration. Once LE is stable for an application, the compiler migration can begin in parallel with the continuing migration of the run-times for unrelated applications.

### What Tools Are Available?

At the SHARE User Group ([www.share.org](http://www.share.org)), almost all LE migration presentations recommend the use of automated tools whenever possible. Tools are usually written as repeatable processes that look for specific conditions or issues that are known troublemakers. The quality and maturity of the tool are important to your success. Many of the premier tools have been available for more than a decade and were battle tested during Y2K.

John Scull, an LE/compiler migration specialist with the IBM Global Services organization, is responsible for MVS Program Products support for IBM internal locations. His 2006 SHARE presentations, Session 8231, "An LE Migration From A to Z," is a step-by-step approach to LE migration. His most recent SHARE presentation has the same session number, but is titled "DB2, CICS and IMS Are Moving On—Will Your Program Languages Keep Up?" This presentation addresses migrations as they apply to DB2 and CICS. All of his sessions are available to SHARE members on the SHARE Website. John believes in using the best automated tools available to perform assessments and to manage the progress of a migration. He will start an assessment project only when the customer has installed an appropriate set of LE migration tools.

The IBM migration guides can help you identify some of the desirable functions you might want to look for in LE migration tools, including the ability to:

- Handle application inventory verification and validation
- Identify LE run-time and compilation issues in machine-readable format
- Identify incompatibilities in language usage
- Convert the incompatibilities in COBOL syntax to valid 1985 standard COBOL syntax
- Convert old CICS command syntax to the syntax required for current COBOL compilers.

A compiler migration may or may not require language syntax changes. Your inventory and assessment of that inventory will point you in the correct direction for this process. Unfortunately, the second sub-step of compiler migration, exploiting new functions, is often ignored. There are several good reasons to move to the new compilers besides the ability to work with the new CICS and DB2. You should try to understand these new features to take advantage of them.

No two LE assessments or migrations are identical because each customer has a different inventory of compilers, applications, and attributes. Most attribute and compiler information is best obtained from the load modules of production applications. Unfortunately, in some cases, this is the only reliable source for this information. IBM provides several z/OS tools that can help identify some of this information. You may have to use multiple tools to evaluate a single module. But, if reading hex dumps isn't your

```

**** FIND CICS V3.1 (OR LATER) MIGRATION ISSUES: OS/V5 COBOL ****

      LINK OR
LOAD  CONTAINS MR  COMPILER  COMPILER  COMPILER  LIBR
MODULE CSECT  TYPE DATE      DATE      TIME      ID
-----
ACTUL001          03002          2002
          CALCPFB3 VS  99021  01/21/99  12:34:45
          QUERY01  VS  95032  02/01/95  21:32:32
ACTUL002          03002          2002
          LOOKUP01 VS  99012  01/12/99  03:14:17
ACTUL003          03002          2002
          RECALC01 VS  99002  01/02/99  03:04:27
-----
**** FIND CICS V3.1 (OR LATER) MIGRATION ISSUES: OS/V5 COBOL ****

TOTAL MODULES ANALYZED          8,476
TOTAL CSECTS ANALYZED          12,954

MODULES WITH CICS AND OS/V5 COBOL          3
CSECTS CREATED BY OS/V5 COBOL IN MODULES WITH CICS          4

```

Figure 3: What CICS Transactions Contain OS/V5 COBOL?

```

**** FIND DB2 V8 MIGRATION ISSUES: UNSUPPORTED COBOL COMPONENTS ****

      LINK OR          LINK OR
LOAD  CONTAINS MR  COMPILER  COMPILER  COMPILER  CODE  LIBR
MODULE CSECT  TYPE DATE      DATE      TIME      PAGE  ID
-----
LOADMOD1          03002          22:44:16          2001
          CALCPFB3 VS  99021  01/21/99  12:34:45
          PRESORT1 C2  95032  02/01/95  21:32:32
LOADMOD2          03002          08:16:44          2001
          PASTHIST C3  99012  01/12/99  03:14:17  01140

NOTE 1: ONLY THE COBOL CSECTS THAT ARE UNSUPPORTED BY DB2 V8,
        ARE LISTED IN THIS REPORT.
NOTE 2: ON THIS REPORT MRTYPE=C3 DENOTES ONLY:
        ENTERPRISE COBOL PRE-V3.2.0.
NOTE 3: OS/V5 COBOL DOES NOT RECORD THE COMPILER DATE MM/DD/YY UNDER
        CERTAIN CIRCUMSTANCES.

```

Figure 4: Find COBOL Components Not Supported With DB2 V8 or Later

cup of tea, you probably want to find other alternatives.

Ideally, to complete an assessment, you want a comprehensive set of flexible machine-readable records and straight-forward exception and analysis reports that identify migration inhibitors in your production application inventory. These flexible machine-readable and report formats also should be able to be used as input to the tool you use to manage your source code changes and testing processes.

Mellon Financial, a global financial institution based in Pittsburgh, PA, migrated to LE using this process to minimize their resource requirements, reduce costs, and accelerate the completion of their LE migration project plan. It all began with a complete understanding of the inventory of languages, compilers, and attributes currently in use.

Shawn Klingensmith, a vice president in Mellon's Technology Delivery Department, indicates that, "By combining output from source inventory tools (such as Micro Focus Revolve) with load module inventory tools (such as Edge Portfolio Analyzer), we were able to create a repeatable monitoring process that identifies:

- Where elements are used or embed-

ded in other elements such as programs, JCL, and online transactions

- The compiler versions and compiler options used to create elements
- Obsolete and duplicate elements, which significantly reduced the conversion scope and inventory.

Once your inventory is completely validated, you can move forward in a comprehensive manner to ensure the integrity of your systems. Because you have a clear picture of the issues, you also can understand where a change in one area might impact other applications or parts of the same application. In some cases, this global view may point you to a more efficient approach to the migration. The typical approaches concentrate on a bottom-up migration, while starting with those areas that have the highest probability of success.

So far, the effort we've described has emphasized planning. Success hinges on a clear understanding of the work to be done.

### Your Migration Team

Make sure everyone involved with the migration is trained on LE, especially those reviewing the LE compilation changes. Make sure the LE program or

application migration issues have been previously identified and are implemented when the changes are applied.

The actual migration process (physically making the changes) requires participation of all the stewards of the application development and maintenance process. Personnel from application development and support, plus subsystem and z/OS installers, must be involved.

Summary and detailed analysis reports should be produced to satisfy the technical requirements of all participants. For instance, all issues could be broken down into three high-level categories:

- LE run-time issues
- LE-conforming compiler issues
- Middleware issues.

Each of these categories can be further subdivided by specific technologies such as CICS, DB2, and by languages. The effort should be tailored to develop the LE project plan, and the information provided to all participants so they can complete their tasks in a timely manner.

Figures 1 through 5 show sample reports that can be used in project assessment, planning, migration, and management.

### LE Migration Complexity

An LE migration and conversion is neither as simple nor as complex as it may appear. For example, the conversion of old source syntax from an outdated OS/VS COBOL syntax that used the 1968 standard language to the 1985 level standard, while also having to deal with changes to the CICS command-level language can appear to be daunting. By applying appropriate tools such as the IBM COBOL and CICS Command-Level Conversion Aid (CCCA) tool to the process, it's typical to find that the tool can handle more than 90 percent of the work. It isn't unusual for CCCA to address 100 percent of the source language changes.

CCCA isn't the only such tool; you can use several others for this portion of the migration process. The key point is that the use of appropriate tools can simplify the task from one that may initially seem a major effort into a purely mechanical task that just requires careful monitoring and some reasonable testing. On the other hand, attempting to perform this task manually can take an extended time and is quite error-prone.

```

**** GLOBAL CSECT CROSS REFERENCE ****

CSECT      MODULE/LIBR
-----
HRT1851    ARS0000 /2001  ARU0000 /2001  DER0000 /2001  HRY0000 /2001
           HRT0000 /2002  KR00000 /2002
LNE9485    DER0000 /2001  LNE0000 /2001  LNX0000 /2002  CFG0000 /2002
           MIN0000 /2002  NFG0000 /2002  NQR0000 /2002  PIP0000 /2002
           LNE0000 /2002

**** GLOBAL CSECT CROSS REFERENCE ****

MRSUM LIBRARY          VOLUME  LIBR
-----
BFEL.PRODLIB          USER01  2001
KDQR.PRODLIB          VOL023  2002

**** GLOBAL CSECT CROSS REFERENCE SUMMARY REPORT ****

TOTAL MACHINE RECORDS          67
TOTAL MRSUM RECORDS            2

CSECTS PROCESSED                18
UNIQUE CSECT NAMES PROCESSED   2

NOTE: ALL CSECTS FOR A MODULE ARE PROCESSED,
      EXCEPT RUN-TIMES AND INTERFACES

```

Figure 5: Where Are CSECTs Used in All Load Libraries?



Frequently, an LE and compiler migration project may take a year or more to complete, depending on the resources allocated; it also may be much shorter, depending on your starting point. More than one tool is often required to efficiently complete the migration. You should first acquire tools that help you complete the assessment phase of the LE migration. Once you have a good understanding of your inventory, you can then make a more intelligent decision about the additional tools that may be required.

The COBOL, PL/I and LE migration guides provide some complexity charts and guidelines to help you size the effort. The more complex the inventory, the more likely tools should be considered to assist in your migration process.

### Summary

A rather famous saying is, "If you don't know where you're going, it matters not which path you take." It also can be said that, "If you don't know where you're going, how do you know when you're there?" And there's a Yogi Berra variation, "You got to be careful if you don't know where you're going, because

you might not get there."

All three variations apply to an LE migration that commences without a clear understanding of the current environment, how you got there, and where you need to be.

In the past, many companies have invested millions of dollars on CICS and DB2 applications. Migration to DB2 V8 and CICS TS V3.1 provide opportunities for exploiting the interoperability of those legacy applications in the LE environment, using the new System z specialty processors, and full XML implementation.

Make sure your migration project has the full support of your IT management team. When they understand the value proposition, the importance to the business, and then communicate both to those involved in the migration project, the odds of a successful project significantly improve.

By avoiding the potholes in your LE migration plan, you position yourself to fully understand what must occur, explain why it's required, and decide when it needs to be completed.

A clear understanding of your application portfolio will yield the

necessary insight on where you are and the effort required to get you where you need to be. Once you understand where you're going and how you intend to get there, an LE migration, a compiler migration, or a DB2 or CICS migration become a matter of executing your plan. **Z**

### About the Authors

M. Carl Gehr Jr. is the lead developer and principal consultant for Edge Information Group and has more than 45 years of experience in information systems, including 32 years as an IBM Systems Engineer. He participated in the GUIDE/SHARE Languages Futures Task Force from 1979 to 1984. This group defined most of the basic requirements that led to Language Environment. Currently, he's co-project manager of the SHARE LE and Languages Project.

Email: [cgehr@edge-information.com](mailto:cgehr@edge-information.com)  
Website: [www.edge-information.com](http://www.edge-information.com)

Nate Murphy is president of Nate Murphy International and the New York DB2 users group, TRIDEX. He was a founding member of the IDUG BOD and has more than 40 years of IT experience in IBM mainframe operational efficiencies. He sells the Edge Portfolio Analyzer product and provides an LE Assessments service.

Voice: 856-234-2353  
Email: [nmurphy@natemurphy.net](mailto:nmurphy@natemurphy.net)  
Website [www.natemurphy.net](http://www.natemurphy.net)

# SIMPLIFY LOAD MODULE MANAGEMENT



## Use the Edge Portfolio Analyzer To:

- ◆ Identify Language Impediments to DB2 & CICS Upgrades
- ◆ Plan Your LE Migration
- ◆ Guide Implementation of New Compilers
- ◆ Troubleshoot and Debug Programs
- ◆ Populate Change Management Database

## Release 6.03 Now Available Supporting:

- Operating Systems through z/OS 1.8
- IBM Enterprise COBOL through V3.4.1
- IBM Enterprise PL/I through V3.5
- Identification of COBOL SEARCH ALL in Load Modules
- *And much, much more....*

Get Your FREE  
General Information Manual  
GO TO [www.edge-information.com](http://www.edge-information.com)  
or CALL 847-297-2020