**TITLE: [CICS Transaction Server Dynamic LIBRARY Support]**

**HOST: Hi, and welcome to the 'Did you say Mainframe?!' podcast series. This is where we regularly interview IBM technical experts who can help you to understand important IBM mainframe hardware and software issues. I'm your host Sherrie Abshire.**

**Today we're going to talk about the CICS Transaction Server Dynamic LIBRARY support and how it can assist in 24x7 operations, as well helping to organize your application data sets.**

**Our guest today is Catherine Moxey from the CICS group based in Hursley in the UK. Catherine, it's great to have you here.**

SME: Thank you very much, it's my pleasure.

**HOST: Before we begin, I'd like to mention to our listeners that there is a more detailed presentation on CICS Dynamic Library Support, I'll be giving out more information at the end of this podcast.**

SME: (optionally responds)

**HOST (question 1): Catherine, one of the new features in CICS Transaction Server V3.2 is support for Dynamic LIBRARY Management. Can you tell me more about that and why it's useful?**

SME (answer 1): Yes, I'd be happy to. When CICS runs an application program, it must first load it from a data set, actually either a PDS or PDSE data set, and CICS needs to know about the data set in which the program can be found. Prior to the Dynamic LIBRARY support, the only way to tell CICS about such data sets, was to list them in the DFHRPL concatenation in the CICS startup JCL. If you wanted to introduce a new program to a CICS system, and the data set wasn't already in the DFHRPL concatenation, then you would have to amend the JCL and restart the CICS. With dynamic LIBRARYs, you can add the data set to a new or existing LIBRARY definition, install it, and it is immediately available to CICS.

Another benefit is that this allows you to organize the data sets in a more application-oriented way. DFHRPL is just a serial list of data sets, but each dynamic LIBRARY is a collection of from 1 to 16 logically related data sets, and this can be given a name reflecting their use, such as the application provided by the program artifacts in the data sets.

**HOST (question 2): Can you give me an example of how this support would be used?**

SME (answer 2): Certainly. Let's suppose that you have a new application that you want to introduce to a CICS system. This could be a development or test system that it isn't convenient to shut down and restart, or a production system where that simply wouldn't be practical. Let's suppose that the various programs and related artifacts (such as user exits, or perhaps BMS maps) are in a couple of PDSE (partitioned data set extended) data sets.

All you would have to do is to define a dynamic LIBRARY using CICSPlex SM BAS or RDO, listing those data sets as part of the LIBRARY, make the associated program definitions and so on, and then install the LIBRARY. When the application is run, CICS will search for the programs in data sets that it knows about, which will now include those defined in the new LIBRARY, and therefore CICS will find and load them.

**HOST (question 3): You mentioned that CICS will now search the data sets in the LIBRARY, what determines the order CICS will search for programs to load?**

SME (answer 3): That's a good question. With the DFHRPL concatenation, the order is simple, because it is just the order in which the data sets are listed. With dynamic LIBRARY support, each LIBRARY is in itself like a mini concatenation, so the data sets defined within a LIBRARY (if there is more than one) will be searched in the order they are listed in the definition. But we have also introduced the concept of a "search order" through all LIBRARYs installed in the system, and each LIBRARY has an attribute called its RANKING which indicates where it belongs in the search order relative to other LIBRARYs. RANKING can be a number from 1 to 99 (but not 10 as I'll explain in a moment), and a LIBRARY with a smaller RANKING will be searched first; so, for example, a LIBRARY with RANKING of 20 will be searched before one with a RANKING of 30. Most of the time, the position of a LIBRARY in the search order probably won't matter, because the programs making up the application in one LIBRARY are unlikely to intersect with the programs in another application. But the RANKING allows you to control this.

Suppose, for example, that you want to introduce an updated version of an application. The original version of the application would probably be in a LIBRARY with a RANKING of 50 because that is the default. If you define a LIBRARY with the data sets containing the new version of the application with a RANKING of 40, and then install the LIBRARY and issue NEWCOPY or PHASEIN for the programs, then CICS will load and run the updated application, because it will search in this new LIBRARY first. What's more, if there are any problems with the new version, then you could revert to the previous version by just disabling the new LIBRARY with the RANKING of 40 and reissuing NEWCOPY requests, to go back to running the original version.

2008 WebSphere System z Podcasts – 'Did you say Mainframe?' Incidentally, DFHRPL itself has a RANKING to indicate where it appears in the search order. It is regarded as a special kind of static LIBRARY, which cannot be changed and, obviously, not disabled. DFHRPL has a pre-set RANKING of 10, and you might notice that this means you can, if necessary, install dynamic LIBRARYs that will be searched before DFHRPL. Another thing to note about RANKING is that any number of LIBRARYs can have the same RANKING, but only DFHRPL can have a RANKING of 10.

**HOST (question 4): So, can I tell where my programs are being loaded from?**

SME (answer 4): Yes. One of the additional enhancements we've introduced with Dynamic LIBRARY support is that when you inquire on information about a program, you can now also get information about the LIBRARY and the data set from which it has been loaded (if it has been loaded from a LIBRARY). This also applies if the LIBRARY it has been loaded from is DFHRPL.

The information is available from the INQUIRE PROGRAM SPI, from the CICSPlex SM Program views and base table, from CEMT INQUIRE PROGRAM and from the XPI.

**HOST (question 5): Are there any limitations or restrictions?**

SME (answer 5): The main thing people might need to be aware of is that programs used in Phase 1 PLT processing during CICS startup must be in data sets in the DFHRPL concatenation. This is because of when during startup the dynamic LIBRARYs are installed. Programs used in Phase 2 PLT processing can be in dynamic LIBRARYs. As an aside, while I am mentioning startup, it is worth saying that dynamic LIBRARYs can be installed during startup on cold or initial start by CICSPlex SM BAS install or GRPLIST install, and they are automatically restored from the catalog on a warm or emergency restart.

There are a few other restrictions which shouldn't affect most people, such as the fact that dynamic LIBRARY support assumes the data sets defined in a LIBRARY have a disposition of SHR. If for some reason a data set requires a disposition of OLD, then that would have to go into DFHRPL.

Also, although CICS doesn't place any explicit restrictions on the number of dynamic LIBRARYs you can define or install, if you were to install very large numbers of LIBRARYs then you could run into z/OS system limits.

**HOST (question 6): Ok, hey..finally, is there a choice over whether or not to use dynamic LIBRARY management?**

SME (answer 6): Absolutely. You get complete flexibility over how you use this. You could continue to use DFHRPL in exactly the same way as pre-CICS TS V3.2, and just get the benefits of seeing which data sets programs have been loaded from. Or, you can have some in-house application data sets defined in dynamic LIBRARYs, and everything else in DFHRPL. Or, you could choose to move all of your in-house and vendor applications to dynamic LIBRARYs, and just use DFHRPL for things like CICS itself and other IBM-supplied data sets.

There's a lot of information in the CICS TS V3.2 InfoCenter to get people started on using this facility. For example, I haven't even had a chance to tell you about the CRITICAL attribute for a LIBRARY, or to say very much about enabling and disabling LIBRARYs.

We are seeing a lot of interest in this new capability, and indeed our own Test teams have found it very useful for installing and managing their test programs.

**HOST: Thanks Catherine, that was really interesting.**

SME: Thanks very much for giving me the opportunity to talk about CICS Dynamic Libraries.

**HOST: Well, that wraps up this podcast discussion. To find out more about the detailed presentation on CICS Dynamic Library Support I mentioned earlier, please go to the description for this podcast at:**
**http://www.ibm.com/software/os/systemz/podcasts/websphereonz/**

**There you will also find the URL for registering for the IBM Impact 2008 event which is being held in Las Vegas in April.**

**And for more in-depth information on the capabilities of CICS TS please go to www.ibm.com/cics**

**Join us next time as we talk about another important mainframe topic. For now, this is Sherrie Abshire saying "Thanks for listening".**