

THE BUFFER POOL - AN OVERVIEW OF MAINFRAME SPECIALTY PROCESSORS

By Craig S. Mullins

Anyone who uses an IBM z Series mainframe has probably heard about zIIPs and zAAPs and other specialty processors. But maybe you haven't yet done any real investigation into what they are, what they do, and why they exist. So, with that in mind, let's take a brief journey into the world of specialty processors!

Over the course of the past decade or so, IBM has introduced several different types of specialty processors. The basic idea of a specialty processor is that it sits alongside the main CPUs and specific types of "special" workload are shuttled to the specialty processor to be run there instead of on the primary CPU complex.

Why is this useful or interesting to mainframe customers? Well, the specialty processor workload is not subject to IBM or independent software vendor (ISV) licensing charges... and, as any mainframer knows, software cost can be many multiples more expensive than the hardware cost. Mainframe software contracts are tied to the processor size of the machine on which the software is to be run, and the cost of the software rises as the capacity on the mainframe rises. But, if capacity can be redirected to a specialty processor, then software license charges do not accrue -- at least for that workload.

Another benefit of the specialty processors is that they are significantly cheaper to acquire than standard CPUs. A standard mainframe CP costs around \$250,000, whereas a specialty processor costs less than \$50,000. That is less than one fifth the price of the standard processor.

And specialty processors can be purchased for a one-time charge per engine, which includes no-charge replacement by faster zIIP engines when upgrading to a new machine. So many organizations today are augmenting their mainframes with specialty processors to delay costly upgrades.

But, of course, there is a catch! The specialty processors can only run certain types of workloads - at least this is the case as configured by IBM "out of the box."

There are four types of specialty processors:

- *ICF: Internal Coupling Facility - used for redirecting coupling facility cycles in a data sharing environment*
- *IFL: Integrated Facility for Linux - used for processing zLinux workload on an IBM mainframe*
- *zAAP: Application Assist Processor - used for the Java and XML parsing*
- *zIIP: Integrated Information Processor - used for processing certain, distributed database workloads.*

When you activate any of these processors some percentage of the relevant workload can be redirected off the main CP onto the specialty processor... but not 100% of the workload. Indeed, one

of the frustrating issues, particularly with the zIIP, is that it can be troublesome to understand exactly what is being redirected, exactly when, and exactly how much of the workload is being redirected.

LET'S TALK ABOUT THE ZIIP

The zIIP, being an "information" processor, is particularly interesting to DB2 users. It was introduced by IBM in 2006 and DB2 V8 was the first version to take advantage of the zIIP. Additional DB2 workload can be redirected to the zIIP as of DB2 V9.

In general, distributed DB2 for z/OS workload and XML processing can be redirected to zIIP processors. To be a bit more precise, the following workloads can benefit from zIIPs:

- *Remote DRDA access, including JDBC and ODBC access to DB2, including access across LPARs using Hipersockets*
- *Native SQL stored procedures that are run through DDF (DB2 V9)*
- *BI application query processing utilizing DB2 star-schema parallel query capabilities*
- *XML parsing (DB2 V9) and*
- *Certain IBM DB2 utility processing that performs maintenance on index structures*

Keep in mind, though, that even with zIIPs installed, all of the potential workload will not be redirected to the zIIP - only a percentage of it. Some people refer to the amount of workload that can be redirected as the IBM generosity factor.

So how many zIIPs can you install? The short answer is: you can have one zIIP for each main CP. This is the case for zAAPs, too.

TCBS AND SRBS

To fully comprehend what can and cannot run on a zIIP, we need to discuss TCBS and SRBs. Many DB2 administrators first heard about TCBS and SRBs in an IBM performance class, but not everyone has taken one of those classes. And even for those of us who have, a refresh is probably in order.

For mainframe z/OS programs, code can execute in one of two modes: TCB mode, also known as task mode, or SRB mode. Most programs execute under the control of a task. Each thread is represented by a TCB, or Task Control Block. A program can exploit multiple processors if it is composed of multiple tasks, as most programs are.

An SRB, or Service Request Block, is a control block that represents a routine that performs a particular function or service in a specified address space. SRBs are lightweight and efficient, but are available only to supervisor state software. An SRB is similar to a TCB in that it identifies a unit of work to the system. But an SRB cannot "own" storage areas. SRB routines can obtain, reference, use, and free storage areas, but the areas must be owned by a TCB. SRB mode is used by operating system facilities and vendor programs to perform certain performance-critical functions.

In general, z/OS will dispatch DB2 work in TCB mode if the request is local, or in SRB mode if the request is distributed. These parallel tasks are assigned the same importance as the originating address space.

Preemptable enclaves are used to do the work on behalf of the originating TCB or SRB address space. Enclaves are grouped by common characteristics and service requests and, since they are preemptable, the z/OS dispatcher - and Workload Manager - can interrupt these tasks for more important ones. There are two types of preemptable SRBs: client SRBs and enclave SRBs.

If the DB2 request is distributed DRDA workload, then it will be executed in enclave SRBs. If the request is coming over a local connection, then it will be dispatched between TCBs, client SRBs, and in some cases enclave SRBs (such as for parallel queries and index maintenance).

So what does all of this have to do with specialty processors? To run on a zIIP, the workload must run under an enclave SRB. So, code written to execute under a TCB usually will be unable to execute under an SRB without major changes.

WHAT DOES IT ALL MEAN?

This is the fun part. Here is where we analyze the specialty processor environment. And, of course, all of the following is just my personal opinion. First of all, why is IBM doing this? It seems to me that IBM is promoting specialty processors to extend the life of the mainframe platform.

Now anyone who knows me knows that I am a huge mainframe proponent and have argued vehemently that it is not a dying platform. Nevertheless, many organizations have abandoned the mainframe for modern workloads: Java, XML, web development, etc. By offering inexpensive specialty processors for the mainframe that run this type of workload, IBM is hoping to encourage more modern workloads to be built for, and run on, the mainframe.

I've heard many old-timers grumble about this, saying something like "Well, why don't they just decrease the price of the hardware instead of making me worry about specialty processors that only run certain stuff." Good point - at least from the customer perspective. But from IBM's perspective (again, just my opinion) decreasing prices overall does nothing to protect the large revenue stream IBM earns from mainframe hardware and software.

So the desire is to keep customers running their legacy batch and transaction processing workload on the expensive hardware (because there is little competition), while reducing the price for new workload (because there is stiff competition).

SYNOPSIS

The bottom line, though, is that specialty processors are here to stay and they can be used to help to reduce the cost of mainframe computing. Even more exciting will be to watch and take advantage of new offerings as additional software innovations are introduced by ISVs to help organizations more effectively utilize specialty processors for more and varied types of workloads. So an investment in specialty processors may be able to pay for itself in no time at all!