Active Memory Expansion: Overview and Usage Guide

February 9, 2010

By David Hepkin

IBM Systems and Technology Group

Table of Contents

1 Introduction

IBM's POWER7™ systems with AIX® feature Active Memory™ Expansion, a new technology for expanding a system's effective memory capacity. Active Memory Expansion employs memory compression technology to transparently compress in-memory data, allowing more data to be placed into memory and thus expanding the memory capacity of POWER7 systems. Utilizing Active Memory Expansion can improve system utilization and increase a system's throughput.

This paper provides an overview of POWER7's Active Memory Expansion technology, as well as guidance on how to deploy and monitor workloads with Active Memory Expansion.

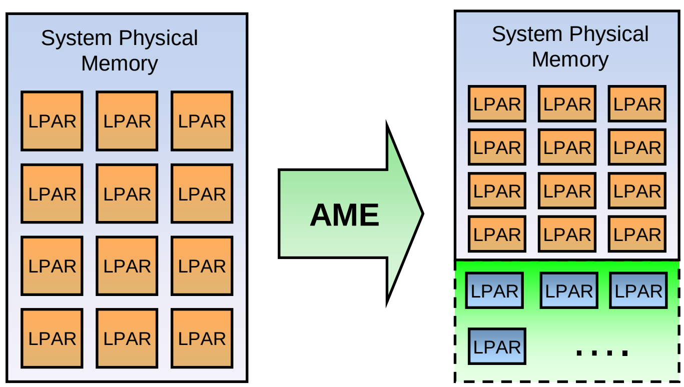
2 Active Memory Expansion Value

Active Memory Expansion increases a system's effective memory capacity. The additional memory capacity made available by Active Memory Expansion enables a system to do more work, leading to an increase in the system's throughput and utilization. Thus, the value of Active Memory Expansion is that it enables a system to do more work by increasing a system's effective memory capacity.

There are multiple ways this additional memory capacity can be used. The following are a few scenarios that demonstrate the value of expanding a system's memory capacity with Active Memory Expansion.

2.1 Higher Consolidation

One way Active Memory Expansion can be used is to enable consolidation of more Logical Partitions (LPAR's) onto a single system.



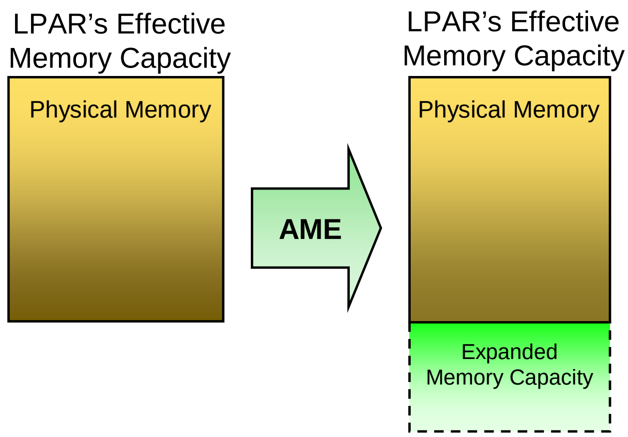
In a memory constrained environment, the number of LPAR's that can be deployed on a system is limited by the system's physical memory size. By enabling Active Memory Expansion for the LPAR's on the system, the physical memory configuration for each LPAR can be reduced, resulting in a net memory savings across all of the existing LPAR's on the system. This memory savings can then be used to deploy more LPAR's onto the system.

Adding more LPAR's to a system means that the system can do more work, increasing the system's utilization and throughput.

2.2 LPAR Throughput Increase

Another way Active Memory Expansion can be used is to increase the effective memory size of a single LPAR, enabling the LPAR to do more work.

In a memory constrained environment, the memory size of a LPAR is constrained by the system's physical memory size. A user may want to increase the workload in a LPAR, but in most cases, increasing the workload in a LPAR increases the LPAR's memory requirements. If the system is out of memory, the LPAR's memory cannot be increased, and thus, the workload in the LPAR is constrained and cannot be increased.



By enabling Active Memory Expansion for the LPAR, the effective memory capacity of the LPAR can be increased without increasing the physical memory required for the LPAR. Additional workload can then be added to the LPAR to take advantage of this effective increase in memory capacity, leading to higher throughput and increased utilization of the LPAR.

3 Active Memory Expansion Overview

Active Memory Expansion relies on compression of in-memory data to increase the amount of data that can be placed into memory and thus expand the effective memory capacity of a POWER7 system. The in-memory data compression is managed by the operating system, and this compression is transparent to applications and users.

Active Memory Expansion is configurable on a per-logical partition (LPAR) basis. Thus, Active Memory Expansion can be selectively enabled for one or more LPAR's on a system.

When Active Memory Expansion is enabled for a LPAR, the operating system will compress a portion of the LPAR's memory and leave the remaining portion of memory uncompressed. This results in memory effectively being broken up into two pools – a compressed pool and an uncompressed pool. The operating system will dynamically vary the amount of memory that is compressed based on the workload and the configuration of the LPAR.

The operating system will move data between the compressed and uncompressed memory pools based on the memory access patterns of applications. When an application needs to access data that is compressed, the operating system will automatically decompress the data and move it from the compressed pool to the uncompressed pool, making it available to the application. When the uncompressed pool is full, the operating system will compress data and move it from the uncompressed pool to the compressed pool. This compression and decompression activity is transparent to the application.

Because Active Memory Expansion relies on memory compression, some additional CPU utilization will be consumed when Active Memory Expansion is in-use. The amount of additional CPU utilization needed for Active Memory Expansion will vary based on the workload and the level of memory expansion being used. The sizing guidelines provided later in this paper give an overview of how to estimate the additional CPU utilization needed for Active Memory Expansion.

3.1 Memory Expansion Factor

When configuring Active Memory Expansion, there is just a single configuration option that must be set for the LPAR – the memory expansion factor. An LPAR's memory expansion factor specifies the target effective memory capacity for the LPAR. This target memory capacity provides an indication to the operating system of how much memory to make available with memory compression.

The memory expansion factor is specified as a multiplier of a LPAR's true memory size:

```
LPAR_expanded_mem_size = LPAR_true_mem_size * LPAR_mem_exp_factor
```

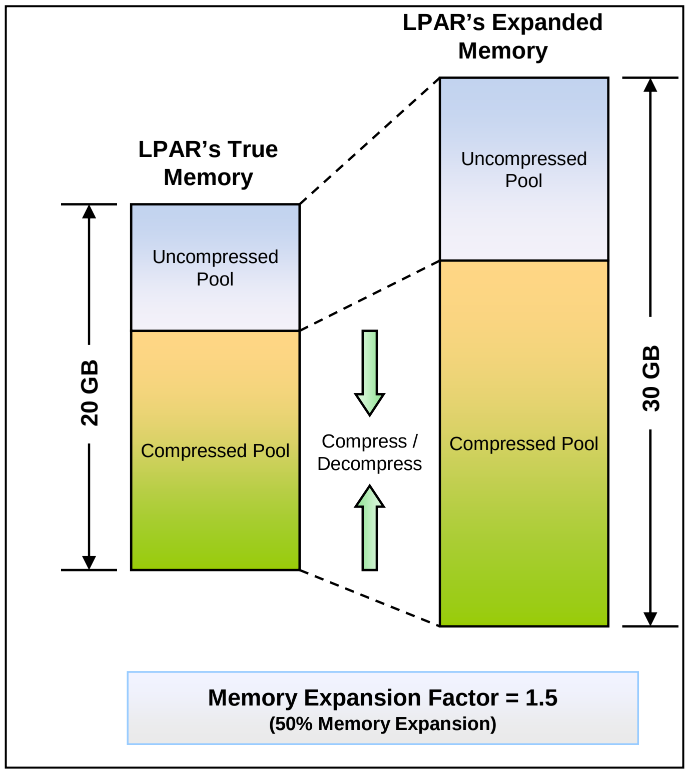
For example, using a memory expansion factor of 2.0 for an LPAR indicates that memory compression should be used to double the LPAR's memory capacity. So, if a LPAR was configured with a memory expansion factor of 2.0 and a memory size of 20GB, the target memory capacity (expanded memory size) for the LPAR would be 40GB:

```
40GB = 20GB * 2.0
```

The operating system would try to compress enough in-memory data to fit 40GB of data into 20GB of memory.

3.2 Active Memory Expansion - Example

To take a more detailed example, consider a workload that has a memory requirement of 30GB. By enabling active memory expansion for the LPAR containing the workload, the operating system can compress a portion of the workload's 30GB of memory and shrink the LPAR to fit in 20GB of physical memory. This results in a net memory savings of 10GB of memory.



In this example, when Active Memory Expansion is enabled for the LPAR, the LPAR would be configured with a physical memory size of 20GB. Since the target memory size of the LPAR is 30GB but only 20GB of physical memory is allocated, the LPAR's memory has been expanded by 50%. Thus, the memory expansion factor for the LPAR would be:

30GB / 20GB = 1.5

When reporting a LPAR's memory size to applications and the user, the operating system will report the expanded memory size. Thus, in this example, the operating system would report that the LPAR has 30GB of memory, even though only 20GB of physical memory was actually configured for the LPAR. Reporting the expanded memory size to applications ensure that applications scale appropriately when applications use memory size in their scaling calculations.

3.3 System Requirements

Active Memory Expansion is supported across all POWER7 systems. In order to use Active Memory Expansion, the following minimum levels of software are required:

1. HMC: V7R7.1.0.0
2. eFW: 7.1
3. AIX: 6.1 TL4 SP2

AIX is currently the only operating system that supports Active Memory Expansion.

4 Planning considerations

Before deploying a workload in the Active Memory Expansion (AME) environment, some initial planning is required to ensure a workload gets the maximum benefit from AME. The benefit of AME to a workload will vary based on the workload's characteristics. Some workloads can get a higher level of memory expansion than other workloads. The key considerations in determining the suitability and benefit of AME for a workload are the following:

1. **Compressibility of in-memory data –** the better a workload's data compresses, the higher the memory expansion that can be achieved with Active Memory Expansion. If a workload's data compresses well, higher levels of memory expansion can be achieved than if the workload's data does not compress well. In general, most data tends to compress well, and the compressibility of a workload's data is not an issue for most workloads. One exception to this is data that is already compressed such as compressed database objects. Data that is already compressed will not compress further with Active Memory Expansion, and thus workloads that deal mostly with compressed data will not see much benefit from AME.
2. **Memory access patterns –** Different workloads have different memory access patterns. Some workloads frequently reference a large amount of memory. Other workloads frequently access only a small portion of memory and infrequently access other portions of memory. In general, workloads of this second type that tend to frequently access only a small portion of memory will perform best with AME.
3. **Type of memory usage –** Active Memory Expansion will not compress file pages that are cached in memory. Thus, a workload that dedicates a large amount of memory to the file cache, such as a file server, will not see much benefit from AME.
4. **Pinned memory usage –** Active Memory Expansion will not compress pinned virtual memory pages. For most workloads, only a very small percentage of memory is pinned, and thus, pinned memory usage does not have much of an effect. However, some applications can be configured to pin most of their virtual memory. Applications configured in this way will have a large pinned memory footprint and will not likely see much benefit from AME.

Since the above attributes for a workload may not be commonly known, an Active Memory Expansion planning tool is provided with AIX to measure these attributes for a workload. The AME planning tool (`amepat`) will monitor an existing workload and provide an indication of whether the workload is a good candidate for AME based on the above factors.

In addition to the workload attributes described above, another key consideration when deploying AME is the amount of available CPU resource. Active Memory Expansion requires CPU resource for memory compression and decompression activity. This CPU resource will come from the CPU resource configured to the LPAR. For example, if a workload without AME normally requires 3.75 CPU's of capacity, using AME for the workload may increase the workload's CPU requirements to 4 CPU's – 3.75 CPU's for the workload itself and 0.25 CPU for the AME compression and decompression activity. The amount of CPU resource required for Active Memory Expansion will vary based on the workload and the target amount of memory expansion. The AME planning tool can be used to get an estimate of the CPU resource needed for running a workload with AME.

Thus, before moving a workload into the AME environment, it is recommended that the AME planning tool be used to provide guidance on the initial configuration of the LPAR for the workload.

4.1 Using the AME Planning Tool

The Active Memory Expansion planning tool should be run in conjunction with the workload being evaluated for use with AME. The AME planning tool will monitor a workload's memory usage and data compressibility over a user-configurable period of time. The tool will then generate a report with a list of possible AME configurations for the workload. The tool will include an estimate of the CPU utilization impacts for the different AME configurations. The AME planning tool will also recommend an initial AME configuration to try as a starting point.

The AME planning is available as part of AIX starting with AIX 6.1 TL4 SP2. The name of the tool is `amepat`. It can be launched from the AIX SMIT interface or directly from the AIX command-line. The amepat tool should be run by the root user.

The AME planning tool can be run on all versions of IBM Power Systems™ supported by AIX 6.1. This includes POWER4™, POWER5™, POWER6™, and POWER7 systems. Thus, when moving a workload from an older system to a POWER7 system, the AME planning tool can be run on the older system to size the workload for use on the POWER7 system with AME.

There are two key considerations when running the AME planning tool: the time at which to run the tool and the duration to run the tool. To get the best possible results from the tool, the tool should be run during a workload's peak utilization period. This ensures that the tool captures utilization and memory usage information at a workload's peak. So, if a workload is busiest during the 2-hour period of 9:00 a.m. to 11:00 a.m., the AME planning tool should ideally be run during that 2 hour period to capture utilization information for the workload.

Once the Active Memory Expansion planning tool has been invoked, it will run in the background for the specified duration, periodically gathering performance metrics for the current workload. The tool will also periodically scan the workload's memory to determine the compressibility of the workload's in-memory data.

Once the tool has completed its monitoring period, it will generate a detailed report of its findings. This report includes information about what the tool observed as well as the tool's findings for possible AME configurations for the workload.

To illustrate how the AME planning tool can be used, consider the scenario where a user is moving a workload from a LPAR on a POWER5 system to a LPAR on a POWER7 system. In this scenario, the user is interested in evaluating the possibility of using AME for the workload to reduce the LPAR's memory requirements on the POWER7 system.

The first step is for the user to run the amepat tool during a peak utilization time for the existing workload. In this example, the workload's peak utilization time is between 9:00 a.m. and 11:00 a.m. Thus, the user would start the amepat tool at 9:00 a.m. and run it for 2 hours (120 minutes) as follows:

```
amepat 120
```

This will initiate the AME planning tool and start it monitoring the system for 2 hours. After 2 hours has elapsed, the AME planning tool generates the following report:

```
# amepat 120
Collecting Data for 120 minute(s)...

Command Invoked                : amepat 120
Date/Time of invocation        : Tues Nov  3 9:00:08 CST 2009
Total Monitored time           : 2 hours
Total Samples Collected        : 5

System Configuration:
---------------------
Partition Name                 : p5lpar1
Processor Implementation Mode  : POWER5
Number Of Logical CPUs         : 4
Processor Entitled Capacity    : 2.00
True Memory                    : 8.00 GB
SMT Threads                    : 2
Shared Processor Mode          : Enabled-Capped
Active Memory Sharing          : Disabled
Active Memory Expansion        : Disabled

System Resource Statistics:                Average            Min             Max
---------------------------              -----------        -----------        -----------
CPU Util (Phys. Processors)              0.97 [ 49%]        0.89 [ 45%]        1.05 [ 53%]
Virtual Memory Size (MB)                 7892 [ 96%]        7888 [ 96%]        7897 [ 96%]
True Memory In-Use (MB)                  8053 [ 98%]        8049 [ 98%]        8058 [ 98%]
Pinned Memory (MB)                        665 [  8%]         665 [  8%]         665 [  8%]
File Cache Size (MB)                      147 [  2%]         147 [  2%]         147 [  2%]
Available Memory (MB)                     133 [  2%]         129 [  2%]         138 [  2%]

Active Memory Expansion Modeled Statistics:
-------------------------------------------
Modeled Expanded Memory Size   :   8.00 GB
Average Compression Ratio      :   1.99

Expansion    Modeled True      Modeled              CPU Usage
Factor       Memory Size       Memory Gain          Estimate
---------    --------------    -----------------    -----------
  1.01             8.00 GB       0.00 KB [  0%]    0.00 [  0%]
  1.11             7.25 GB     768.00 MB [ 11%]    0.00 [  0%]
  1.21             6.75 GB       1.25 GB [ 21%]    0.13 [  7%]
  1.31             6.25 GB       1.75 GB [ 31%]    0.28 [ 14%]
  1.51             5.50 GB       2.50 GB [ 51%]    0.42 [ 21%]

Active Memory Expansion Recommendation:
---------------------------------------
The recommended AME configuration for this workload is to configure the LPAR
with a memory size of 6.25 GB and to configure a memory expansion factor
of 1.31.  This will result in a memory gain of 31%.  With this configuration,
the estimated CPU usage due to AME is approximately 0.28 physical processors,
and the estimated overall peak CPU resource required for the LPAR
is 1.33 physical processors.
```

The first two sections of the report, "System Configuration" and "System Resource Statistics," are provided for informational purposes. The "System Configuration" section provides an indication of the LPAR configuration in which the tool was run, and the "System Resource Statistics" section reports the minimum, maximum, and average utilization for both memory and CPU during the monitored period.

The next section of the report, "AME Modeled Statistics," provides an indication of the Active Memory Expansion configuration possibilities for the workload. This section starts with the "modeled expanded memory size." By default, the AME planning tool assumes that the user wants the expanded memory size of the AME LPAR to be equal to the current LPAR's memory size. So, in this example, the current LPAR is configured with 8GB of memory, and the AME planning tool assumes that the goal is to continue to provide a target expanded memory capacity of 8GB through the use of Active Memory Expansion. Alternatively, a specific target expanded memory size can be specified to the AME planning tool with the "-t" command-line option.

Following the "Average Compression Ratio" is a table that lists some possible AME configurations for the workload. Each row corresponds to a different AME configuration for the workload. The configurations are listed from the least amount of memory expansion (smallest memory expansion factor) to the highest amount of memory expansion. Each row provides a LPAR configuration that will result in the target expanded memory size, which is 8GB in this example. Here is a summary of the meaning of the different columns:

Column Name	Description
Expansion Factor	This is the value of the memory expansion factor that would need to be configured to achieve the target expanded memory size. This provides an indication of how much memory expansion the operating system should perform.
Modeled True Memory Size	This is the amount of memory to configure for the LPAR.
Modeled Memory Gain	This is the amount of additional memory capacity that will be made available to the LPAR by using AME. The percentage memory gain is relative to the LPAR's modeled true memory size.
CPU Usage Estimate	This is an estimate of the additional CPU resource that will be consumed due to memory compression activity for the current AME configuration. This is intended as an initial guideline. The actual CPU usage consumed for memory compression may be lower or higher than this initial estimate.

To take an example from the above sample output, the third row in the table indicates that 1.25GB of memory can be saved by running the workload in an LPAR with 6.75GB of physical memory and a memory expansion factor of 1.21. When running the workload in a LPAR with this configuration, the estimated additional CPU usage due to memory compression is 0.13 CPU's, which is 7% of the LPAR's currently configured CPU capacity.

The final section of this report, "AME Recommendation," provides a recommendation for an initial AME configuration to try. The AME planning tool will select one of the configurations reported in the "AME Modeled Statistics" table as a recommendation. In the above sample output, the AME planning tool suggests configuring the LPAR with 6.25GB of memory and a memory expansion factor of 1.31. It is important to note that any of the configurations listed in the above table are valid and can be used for the LPAR. The recommendation just provides a suggestion for an initial configuration to try.

The AME planning tool supports a number of additional options that can be used customize the report generated by the tool. Please see the documentation for the AME planning tool for more information on these additional options.

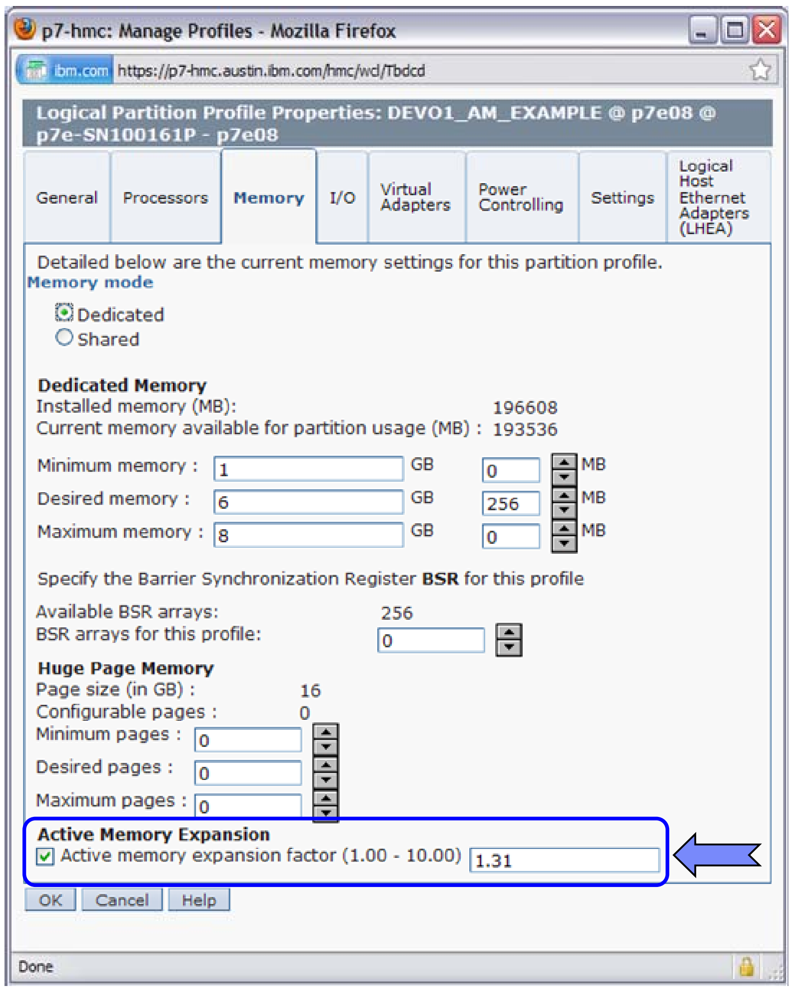
5 Deployment

After running the AME planning tool on the target workload that will use AME, the next step is to configure Active Memory Expansion for a LPAR and deploy the workload into the LPAR.

Configuring Active Memory Expansion for a LPAR is very simple. The only step necessary to configure AME for a LPAR is to specify the memory expansion factor for the LPAR. The memory expansion factor is a LPAR profile attribute that can be configured via the HMC. In the HMC graphical user interface, the AME configuration for a LPAR is located on the memory configuration panel for a LPAR's profile. At the bottom of the memory configuration panel, an "Active Memory Expansion" configuration section is provided. To enable Active Memory Expansion, select the active memory expansion check box and specify the memory expansion factor for the LPAR.

As described in the previous section, the AME planning tool is a good source for determining the initial value for a LPAR's memory expansion factor. Going back to the AME planning tool example in the previous section, the tool suggested configuring the LPAR with 6.25GB of memory and using a memory expansion factor of 1.31. This will enable a total expanded memory size of 8GB for the LPAR. The HMC screen snapshot shows this example configuration.

That's it. The LPAR is now configured to use AME. The next time the LPAR is activated with this profile, AME will be enabled, and memory compression will be used to achieve the expanded memory size configured for the LPAR.



After activating the AME-enabled LPAR, the next step is to run the target workload in the LPAR and monitor the performance of the workload with AME enabled.

6 Monitoring AME

After deploying a workload in a LPAR with Active Memory Expansion enabled, the next step is to monitor the performance of the workload and, if necessary, adjust the Active Memory Expansion configuration based on the performance results of the workload.

This monitoring and tuning step is critical in deploying Active Memory Expansion. The CPU utilization estimates provided by the AME planning tool only give a general expectation of the CPU usage due to memory compression. The actual CPU usage due to Active Memory Expansion may be lower or higher depending on the workload. Thus, after deploying a workload in the AME environment for the first time, it is always best practice to monitor the workload for a period of time to ensure that AME and the workload are performing as expected. The remainder of this section provides guidelines on how to monitor a workload in the AME environment.

6.1 AME Performance Monitoring

Monitoring a workload in the AME environment is not much different than monitoring a workload without AME. The standard AIX and third-party monitoring tools will work as usual in the AME environment. However, due to the memory compression activity that happens when AME is enabled, there are two key performance metrics that should be monitored to determine how efficiently AME is running:

1. CPU utilization
2. Expanded memory deficit

6.1.1 CPU Utilization

When AME is enabled for a LPAR, CPU resource will be consumed for the compression and decompression activity that must be performed. Thus, the CPU utilization of a workload will usually be higher when running in the AME environment than when the workload is running without AME. It is important that the CPU utilization of the workload be monitored to ensure there is adequate CPU capacity available to the workload. If the workload runs out of CPU capacity, the performance of the workload will suffer.

The standard tools for monitoring a LPAR's CPU utilization can also be used in the AME environment. This includes tools like vmstat, lparstat, topas, and others.

In addition to monitoring the overall CPU utilization of the LPAR, the CPU usage specific to memory compression activity can also be monitored. This gives a view of the CPU utilization impact due to running in the AME environment. Both the lparstat and amepat commands can be used to monitor the CPU use for Active Memory Expansion. See the section on performance tools for more details.

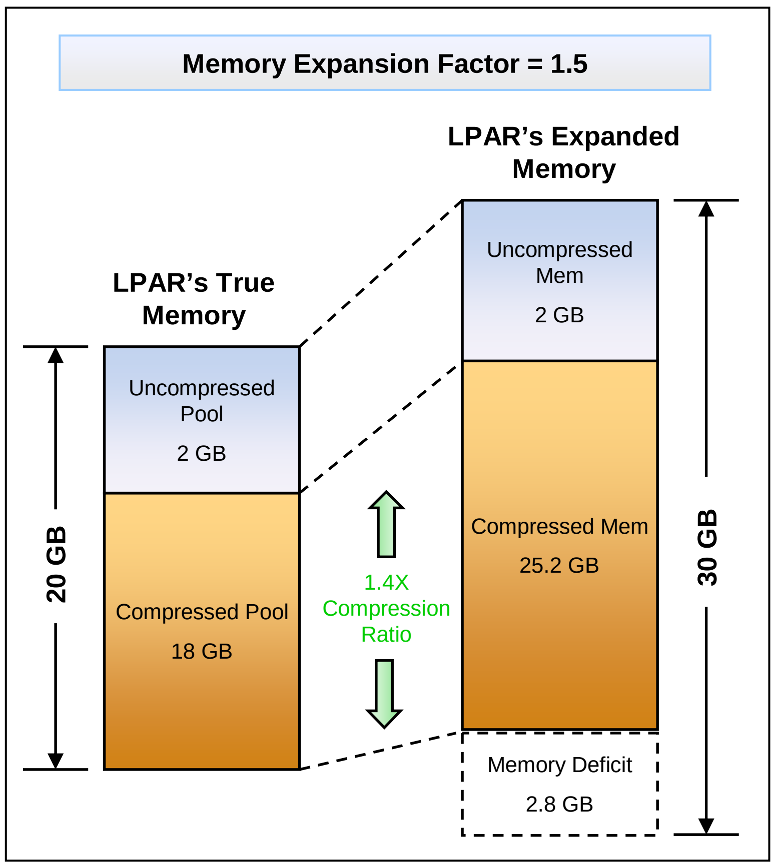
6.1.2 Expanded Memory Deficit

When configuring the memory expansion factor for a LPAR, it's possible that a memory expansion factor may be chosen that is too large and cannot be achieved based on the compressibility of the workload. When the memory expansion factor for a LPAR is too large, a memory expansion deficit will form, indicating that the LPAR cannot achieve its memory expansion factor target.

To illustrate the concept of an expanded memory deficit, consider the following example:

A LPAR is configured with a memory size of 20GB and a memory expansion factor of 1.5. This results in a total target expanded memory size of 30GB. However, the workload running in the LPAR does not compress well, and the workload's data only compresses by a ratio of 1.4 to 1. In this case, it is impossible for the workload to achieve the targeted memory expansion factor of 1.5.

Because the workload's data only compresses by a ratio of 1.4 to 1, the maximum expanded memory size that can be achieved is 27.2GB. Since the target expanded memory size is 30GB, there is a short-fall of 2.8GB of expanded memory. This 2.8GB short-fall is referred to as a memory deficit.



The effect of a memory deficit is the same as the effect of configuring a LPAR with too little memory. When a memory deficit occurs, the operating system will not be able to achieve the expanded memory target configured for the LPAR, and the operating system may have to resort to paging out virtual memory pages to paging space. Thus, in the above example, if the workload used more than 27.2GB of memory, the operating system would start paging out virtual memory pages to paging space.

To get an indication of whether a workload can achieve its expanded memory size, the operating system reports a memory deficit metric. This is a "hole" in the expanded memory size that can't be achieved. If this deficit is zero, the target memory expansion factor can be achieved, and the LPAR's memory expansion factor is configured correctly.

If the expanded memory deficit metric is non-zero, this means that the workload will fall short of achieving its expanded memory size by the size of the deficit. To eliminate a memory deficit, the LPAR's memory expansion factor should be reduced. However, reducing the memory expansion factor will reduce the LPAR's expanded memory size. Thus, to keep the LPAR's expanded memory size the same, the memory expansion factor must be reduced, and more memory must be added to the LPAR. Both the LPAR's memory size and memory expansion factor can be changed dynamically.

In general, memory deficits are rare and can easily be avoided by following the best practices described in this white paper for planning the deployment of a workload in an AME environment.

6.2 Basic Monitoring of Active Memory Expansion

For basic monitoring of the AME environment, the amepat tool can be used. The amepat tool can be used to easily get a summary of the AME configuration and performance metrics for an AME-enabled LPAR. When the amepat tool is run without any options, it will report a snapshot of the LPAR's configuration and AME performance metrics.

When using the amepat tool to get a snapshot, the snapshot will include various CPU utilization metrics, including how much CPU has been used for AME. It is important to note that the CPU utilization metrics reported by the amepat tool are overall average utilization metrics since the LPAR was last booted. In order to do fine-grained monitoring of CPU utilization over specific intervals, other commands like lparstat or vmstat can be used. These commands are described further in the advanced monitoring section.

Here is an example of an AME planning tool snapshot report:

```
# amepat

Command Invoked                : amepat
Date/Time of invocation        : Tue Nov  3 17:35:21 CST 2009
Total Monitored time           : NA
Total Samples Collected        : NA

System Configuration:
---------------------
Partition Name                 : p7e08
Processor Implementation Mode  : POWER7
Number Of Logical CPUs         : 8
Processor Entitled Capacity    : 2.00
True Memory                    : 6.25 GB
SMT Threads                    : 4
Shared Processor Mode          : Enabled-Capped
Active Memory Sharing          : Disabled
Active Memory Expansion        : Enabled
Target Expanded Memory Size    : 8.00 GB
Target Memory Expansion factor : 1.31


System Resource Statistics:               Current
---------------------------          ----------------
CPU Util (Phys. Processors)               1.56 [ 78%]
Virtual Memory Size (MB)                  7943 [ 97%]
True Memory In-Use (MB)                   6395 [100%]
Pinned Memory (MB)                         616 [ 10%]
File Cache Size (MB)                         7 [  0%]
Available Memory (MB)                      256 [  3%]

AME Statistics:                           Current
---------------                      ----------------
AME CPU Usage (Phy. Proc Units)           0.11 [  6%]
Compressed Memory (MB)                    3440 [ 42%]
Compression Ratio                         2.11
```

There are several interesting metrics to note. The following table summarizes some of the key configuration attributes and performance metrics reported by the amepat command:

Attribute	Description
True Memory	The actual amount of memory currently configured to the LPAR
Target Expanded Memory Size	The target amount of expanded memory for the LPAR; this is a function of the LPAR's true memory size and the memory expansion factor configured for the LPAR.
Target Memory Expansion Factor	The value of the memory expansion factor that is currently configured for the LPAR
CPU Util (Phys. Procesors)	The per-second average total CPU usage for the LPAR since the LPAR was booted; this includes the "AME CPU usage" time. The total CPU usage is reported in units of physical processors.
AME CPU Usage (Phy. Proc Units)	The per-second average CPU usage for memory compression activity since the LPAR was booted; this is reported in units of physical processors
Compressed Memory (MB)	The current amount of memory that is compressed in units of megabytes
Memory Deficit Size	If the memory expansion factor configured for the LPAR is too large and the LPAR is experiencing a memory deficit, the size of the memory deficit will be reported. This indicates that more memory should be added to the LPAR and that the memory expansion factor should be reduced.

6.3 Fine-Tuning the Active Memory Expansion Configuration

After doing some initial monitoring of a workload in the Active Memory Expansion environment, the performance metrics may indicate that the workload could benefit from some fine-tuning of the LPAR's Active Memory Expansion configuration.

For example, if the AME CPU usage is very low when AME is enabled, the LPAR may be capable of achieving higher levels of memory expansion. Alternatively, if the AME CPU usage is very high when AME is enabled, the current AME configuration for the LPAR may be too aggressive, and thus, a lower level of memory expansion should be used to reduce the CPU usage associated with AME.

To help with this fine-tuning, the AME planning tool can again be used to provide further guidance on alternative AME configurations. Just as the AME planning tool can be used to provide recommended AME configurations when AME is disabled, the AME planning tool can be used when AME is enabled to provide a list of alternate AME configurations and their estimated CPU usage.

To use the AME planning tool to provide a list of alternative AME configurations, amepat should be run in the AME environment in the same manner that amepat was run during the initial planning phases. Specifically amepat should be run in conjunction with the target workload during the workload's peak utilization period. amepat will then provide a list of alternate AME configurations that can be used.

To take an example of how amepat can be used to fine-tune the AME configuration for a workload, consider the example workload that was described previously in the planning section. When the AME planning tool was initially run on this sample workload, the tool recommended configuring the LPAR with 6.25GB of memory and a memory expansion factor of 1.31. For this configuration, the AME planning tool estimated that the CPU usage due to Active Memory Expansion would be 0.28 CPU's.

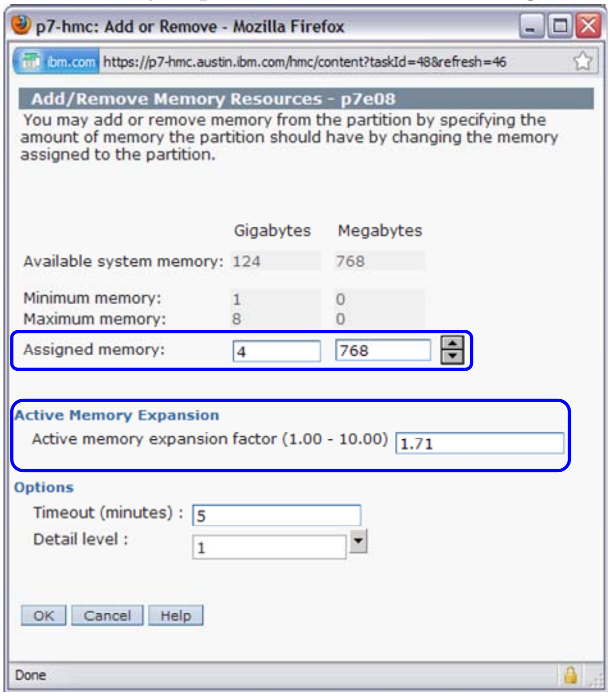
Now that the workload has been deployed in the Active Memory Expansion environment, the amepat snapshot report indicates that the actual CPU usage due to Active Memory Expansion is only .11. This is less than the original target of 0.28 CPU's. Thus, a higher level of memory expansion may be achievable for the original target of 0.28 CPU's. To evaluate alternate AME configurations, the amepat tool is run in the AME environment for two hours (120 minutes):

```
amepat 120
```

The AME planning tool then generates the following report:

```
# amepat 120
Collecting Data for 120 minute(s)...

Command Invoked                : amepat 120
Date/Time of invocation        : Tue Nov  3 15:36:36 CST 2009
Total Monitored time           : 2 hours
Total Samples Collected        : 8

System Configuration:
---------------------
Partition Name                 : p7e08
Processor Implementation Mode  : POWER7
Number Of Logical CPUs         : 8
Processor Entitled Capacity    : 2.00
True Memory                    : 6.25 GB
SMT Threads                    : 4
Shared Processor Mode          : Enabled-Capped
Active Memory Sharing          : Disabled
Active Memory Expansion        : Enabled
Target Expanded Memory Size    : 8.00 GB
Target Memory Expanson factor  : 1.31


System Resource Statistics:               Average           Min             Max
---------------------------             -----------       -----------     -----------
CPU Util (Phys. Processors)             1.56 [ 78%]       1.52 [ 76%]     1.60 [ 80%]
Virtual Memory Size (MB)                7943 [ 97%]       7943 [ 97%]     7943 [ 97%]
True Memory In-Use (MB)                 6395 [100%]       6395 [100%]     6396 [100%]
Pinned Memory (MB)                       616 [ 10%]        616 [ 10%]      616 [ 10%]
File Cache Size (MB)                       7 [  0%]          7 [  0%]        8 [  0%]
Available Memory (MB)                    256 [  3%]        255 [  3%]      259 [  3%]

AME Statistics:                           Average           Min             Max
---------------                         -----------       -----------     -----------
AME CPU Usage (Phy. Proc Units)         0.11 [  6%]       0.10 [  5%]     0.11 [  6%]
Compressed Memory (MB)                  3440 [ 42%]       3394 [ 41%]     3451 [ 42%]
Compression Ratio                       2.10              2.09            2.10

Active Memory Expansion Modeled Statistics:
-------------------------------------------
Modeled Expanded Memory Size   :   8.00 GB

Expansion    Modeled True      Modeled              CPU Usage
Factor       Memory Size       Memory Gain          Estimate
---------    --------------    -----------------    -----------
  1.05              7.75 GB     256.00 MB [  5%]     0.00 [  0%]
  1.31              6.25 GB       1.75 GB [ 31%]     0.11 [  6%] << CURRENT CONFIG
  1.51              5.50 GB       2.50 GB [ 51%]     0.18 [  9%]
  1.71              4.75 GB       3.25 GB [ 71%]     0.28 [ 14%]
  1.91              4.25 GB       3.75 GB [ 91%]     0.76 [ 38%]

Active Memory Expansion Recommendation:
---------------------------------------
The recommended AME configuration for this workload is to configure the LPAR
with a memory size of 4.75 GB and to configure a memory expansion factor
of 1.71.  This will result in a memory gain of 71%.  With this configuration,
the estimated CPU usage due to AME is approximately 0.28 physical processors,
and the estimated overall peak CPU resource required for the LPAR
is 1.73 physical processors.
```

As can be seen in this report, the tool recommends configuring the LPAR with a memory expansion factor of 1.71 and a memory size of 4.75GB. This will save an additional 1.5 GB of memory. The tool estimates that the AME CPU usage for this configuration is only .28 CPU's.



To enact this recommended configuration, the LPAR's configuration can be changed dynamically using DLPAR. To change the LPAR's configuration, the HMC GUI can be used. By selecting "memory" under the DLPAR menu option, the HMC will present a panel providing options on adding and removing memory. To enact the recommended changes, the LPAR's memory size is changed to 4.75 GB, and the LPAR's memory expansion factor is changed to 1.71.

7 Advanced Monitoring

The amepat tool provides a simplified way of monitoring Active Memory Expansion. Additional tooling is provided with AIX for more detailed performance monitoring. These additional tools can be used to monitor the effectiveness and efficiency of AME. This section provides an overview of additional tooling that can be used in the Active Memory Expansion environment.

Here is a summary of options that can be used with the AIX performance tools to monitor AME statistics:

Tool	Option	Description
vmstat	-c	Provides compression and decompression statistics
lparstat	-c	Provides an indication of the CPU utilization for AME compression and decompression activity
svmon	-O summary=ame	Provides a summary view of memory usage broken down into compressed and uncompressed memory
topas		The default topas screen will show memory compression statistics when run in the AME environment

For more information on the output of a specific tool, please consult the tool's documentation in the AIX infocenter.

8 Active Memory Expansion & Active Memory Sharing

A complementary technology to Active Memory Expansion is PowerVM™ Active Memory Sharing. Active Memory Sharing is an existing memory virtualization technology supported on Power servers. While both Active Memory Expansion and Active Memory Sharing are memory virtualization technologies, Active Memory Expansion provides a different set of capabilities than Active Memory Sharing.

Active Memory Expansion provides the capability to expand the effective memory capacity of a system beyond its physical memory capacity. Active Memory Expansion enables this expansion of memory capacity on a per-LPAR basis by packing more data into a LPAR's memory through the use of in-memory data compression.

In contrast to Active Memory Expansion, Active Memory Sharing provides the capability to make better use of a system's existing physical memory by improving the utilization of memory between multiple LPAR's. With Active Memory Sharing, the hypervisor will intelligently move memory between LPAR's based on the memory needs of individual LPAR's. By grouping LPAR's together whose workloads peak at different times, the total amount of memory capacity needed across all of the LPAR's can be reduced with Active Memory Sharing by having the LPAR's share a pool of physical memory.

Thus, unlike Active Memory Expansion, Active Memory Sharing does not provide the capability to expand the memory capacity of an LPAR. Instead, Active Memory Sharing makes more efficient use of a system's existing physical memory by sharing it across LPAR's.

Because Active Memory Expansion and Active Memory Sharing provide different capabilities, they can be used independently or together. For example, AME can be enabled for a set of LPAR's on a machine to expand the LPAR's memory capacities, enabling the LPAR's to do more work. Additionally, if some of these LPAR's have varying workloads that peak at different times, Active Memory Sharing can also be applied to those LPAR's in addition to AME, enabling better utilization of the system's existing physical memory.

8.1 Using AME with AMS

Active Memory Expansion and Active Memory Sharing can be used together. This section highlights some important considerations when using Active Memory Expansion with Active Memory Sharing.

When using Active Memory Expansion with Active Memory Sharing, it is important to understand the relationship between the AME and AMS memory configuration attributes.

When configuring a shared memory (AMS) LPAR, the memory size of the LPAR is referred to as the LPAR's *logical* memory size. The actual amount of physical memory allocated to the LPAR will vary at run-time, based on the hypervisor memory allocation policies and the LPAR's workload. For example, if a shared memory LPAR is configured with a 4GB memory size (referred to as a 4GB *logical* memory size), the actual amount of physical memory allocated to the LPAR will vary dynamically between 0 and 4GB.

When using Active Memory Expansion with a shared memory LPAR, the memory expansion factor configured for the LPAR will apply to the LPAR's logical memory size. Thus, going back to the previous example of a shared memory LPAR with a logical memory size of 4GB, configuring a memory expansion factor of 2.0 for the LPAR would result in an expanded memory size of 8GB. Thus, in this example, the operating system will pack 8GB of data into 4GB of logical memory. Because AMS is also enabled for the LPAR, the actual amount of logical memory that is located in physical memory will vary between 0 and 4GB, but this does not have an effect on the LPAR's expanded memory size. The LPAR's expanded memory size will stay at 8GB regardless of the amount of physical memory allocated to the LPAR at any given time.

8.1.1 Page Loaning

When using AME with AMS, AIX provides an additional feature to optimize the performance of the two technologies working together. Namely, AIX will employ memory compression as part of its Active Memory Sharing collaborative memory management.

Page loaning is a feature that enables the hypervisor and operating system to collaborate on memory management in the Active Memory Sharing environment. When the hypervisor needs to take physical memory away from a LPAR, the hypervisor can use page loaning to ask the operating system to release memory back to the hypervisor, thus reducing the LPAR's memory footprint. In the case of AIX, there is a vmo tunable, ams_loan_policy, that controls the aggressiveness of the page loaning that AIX will perform in response to hypervisor requests.

Normally, AIX will satisfy page loaning requests by paging out pages from memory to disk. In the case of a LPAR where both Active Memory Expansion and AMS are enabled, AIX will rely on memory compression to satisfy page loaning requests. Thus, rather than page out pages from memory to disk in response to a page loaning request, AIX will compress pages in response to loaning requests. Since memory compression is significantly faster than paging to and from disks, this capability can improve the performance of memory over-commitment in the AMS environment.

9 Conclusion

Active Memory Expansion with AIX on IBM's POWER7 systems can increase the utilization and throughput of systems by expanding their effective memory capacity. By following the best practices in this paper and using the tools provided by IBM, deploying a workload into the Active Memory Expansion environment can be done quickly and easily.

IBM Corporation
Systems and Technology Group
Route 100
Somers, New York 10589

Produced in the United States of America
February 2010

This document was developed for products and/or services offered in the United States. IBM may not offer the products, features, or services discussed in this document in other countries.

The information may be subject to change without notice. Consult your local IBM business contact for information on the products, features and services available in your area.

All statements regarding IBM future directions and intent are subject to change or withdrawal without notice and represent goals and objectives only.

IBM hardware products are manufactured from new parts, or new and used parts. In some cases, the hardware product may not be new and may have been previously installed. Regardless, our warranty terms apply.

Information concerning non-IBM products was obtained from the suppliers of these products or other public sources. Questions on the capabilities of the non-IBM products should be addressed with those suppliers.

All performance information was determined in a controlled environment. Actual results may vary. Performance information is provided "AS IS" and no warranties or guarantees are expressed or implied by IBM. Buyers should consult other sources of information, including system benchmarks, to evaluate the performance of a system they are considering buying.

When referring to storage capacity, 1 TB equals total GB divided by 1000; accessible capacity may be less.

The IBM home page on the Internet can be found at: http://www.ibm.com.

The IBM Power Systems home page on the Internet can be found at: http://www.ibm.com/systems/power/

POW03037-USEN-00