July 2013

ASO (Active System Optimizer) – a quick way to get performance for free
Author Jaqui Lynch

In October 2011 IBM announced an enhancement to AIX called Active System Optimizer (ASO) that would only be available on POWER7 recent AIX levels. The intent of ASO was to take some of the best practices performance tweaks and provide a means to automatically apply those changes to individual LPARs. Some of the tweaks provided by ASO include improvements to memory affinity, specifically associating targeted workloads to a specific set of cores (or a core) and determining if memory can be relocated for higher affinity to cache and cores. ASO needs to run on each LPAR that it is tuning as it is a within LPAR solution that runs in the background.

There is a follow-on LPP to ASO called DSO (Dynamic systems optimizer). This requires a license and also requires installation of the dso.aso fileset and has a prerequisite of AIX v6 TL8 or AIX v7 TL02. DSO comes with AIX Enterprise Edition or as a licensed program product for standard edition. For whole server level tuning there is an additional product called DPO (Dynamic Platform Optimizer). This article focusses on implementing ASO only.

ASO is installed by default in a standard installation – it is part of the bos.aso fileset. If you run 'lslpp –l | grep bos.aso' you can see the levelset that is installed. In order to use ASO there are three prerequisites: AIX v6 TL08 or AIX v7 TL01, bos.aso must be installed and the server must be POWER7/7+. Additionally ASO does not support AMS or LPAR migration. Below we check the prerequisites:

```
# oslevel -s
7100-01-05-1228
```
The above shows we are at AIX v7 TL01 SP5
```
# lslpp -l | grep aso
  bos.aso                 7.1.1.15  COMMITTED  Active System Optimizer
  bos.aso                 7.1.1.15  COMMITTED  Active System Optimizer
# lsconf | grep ^Processor
Processor Type: PowerPC_POWER7
Processor Implementation Mode: POWER 7
Processor Version: PV_7_Compat
Processor Clock Speed: 3000 MHz
```
Based on the above this system can run ASO with no problems.

In order to use ASO you first have to start the subsystem using startsrc –s aso. You can then monitor and control it using the asoo command. The following commands can be very useful:
```
oslevel -s
7100-01-05-1228
```
The above shows we are at AIX v7 TL01 SP5

```
# lssrc -s aso
Subsystem         Group            PID          Status
 aso                                5373980      active
```
This shows the ASO subsystem is started and is running as PID 5373980

```
# ps -ef | grep 5373980
    root  5373980  3604670   0   Jan 02      -  1:14 /usr/sbin/aso
```
Here we can see the program being run is /usr/sbin/aso

asoo –L List the current and reboot values
```
# asoo -L
NAME                 CUR    DEF    BOOT   MIN    MAX    UNIT       TYPE
aso_active             0      0         0         0       1     boolean      D
```

#asoo -op aso_active=1 Permanently turn ASO on (0 is disabled and is the default)

You can also check what is happening in the ASO logs- their location is specified in /etc/syslog.conf and defaults to:

```
aso.notice /var/log/aso/aso.log              rotate size 128k time 7d
```

aso.info /var/log/aso/aso_process.log rotate size 1024k
aso.log records the on/off status and the reasons that ASO goes into hibernation.
aso_process.log records the same information but also includes details of any actions ASO takes as well as the processes it modifies.

My test LPAR is on a 720 and has an entitlement of 0.5, VPs set to 2 and 4GB of memory. lssrad shows that I only have one socket so there is not much to tune with respect to affinity.

```
# lssrad -av
REF1   SRAD       MEM      CPU
0
        0    3699.81      0-11
```

Although the ASO subsystem is running we still need to enable ASO as follows:

```
# asoo -o aso_active=1
Setting aso_active to 1
```

The log now shows:

```
Jun 16 17:32:46 testlpar aso:notice aso[5373980]: ASO enabled by tunable
Jun 16 17:32:46 testlpar aso:notice aso[5373980]: [HIB] Current number of system virtual cpus too low (2 cpus)
Jun 16 17:32:46 testlpar aso:notice aso[5373980]: [HIB] Increase system virtual cpus to at least 3 cpus to run ASO. Hibernating.
```

It turns out that ASO needs at least 3 VPs to run so keep this in mind when planning where and when to use ASO. At this point ASO was still enabled so I disabled it again and went and DLPAR'd in another VP to the LPAR.

```
# lsdev -C | grep proc
proc0      Available 00-00      Processor
proc4      Available 00-04      Processor
proc8      Available 00-08      Processor
# asoo -o aso_active=1
Setting aso_active to 1
```

The log now shows:

```
Jun 16 17:39:08 testlpar aso:notice aso[5373980]: ASO enabled by tunable
```

Unfortunately the system I was testing on was not very busy so ASO basically decided to hibernate:

```
LOG
Jun 16 17:40:53 testlpar aso:notice aso[5373980]: [HIB] Used entitlement per unfolded vCPU is below threshold (1% of a core).
Jun 16 17:40:53 testlpar aso:notice aso[5373980]: [HIB] ASO will hibernate until used entitlement is at least 30% of a core per unfolded vCPU
```

On a busy system, ASO now takes some time to profile and analyze the running workloads to dynamically tune the system for those workloads. Now that it is running it needs no further interaction. It will look at AIX kernel data about processes and threads and at the POWER7 hardware performance counters. Based on what it sees there ASO makes decisions on tuning. It will try to improve cache and memory affinity by moving workloads to cores that help them improve their affinity needs. It takes into account WLM and WPAR resource sets as well as SRAD constraints. It does this by looking at the workload over minutes and then setting dispatcher SRAD and RSET rules in order to change the CPU that a workload is running on. All changes made are logged and tracked with ASO logging both the expect and actual gain seen due to the changes it made.

By default ASO is set to enable cache affinity optimization, memory affinity optimization, large page optimization and memory prefetch. These are restricted tunables and should only be modified at the request of IBM support.

As mentioned above, ASO looks for suitable workloads that it can optimize and tries to improve the cache and memory performance by using one of three types of optimization – cache affinity, aggressive cache affinity and memory affinity. For cache affinity ASO is trying to reduce chip to chip cache movement. For a workload that has been assigned cores across two sockets ASO will try to place the workload so that it fits within a socket using an RSET (CPU Resource set). For aggressive cache affinity ASO tries to reduce the

number of chips being used and it may compress the workload onto fewer cores in order to reduce chips. Lastly, for memory affinity ASO tries to ensure that memory being used is local rather than near or far. This may involve migrating pages for workloads from their current DIMMs to DIMMS closer to the core being used in order to reduce remote memory traffic. It uses CPU and Memory RSETs to do this.

Finally, in the unlikely event that you experience problems with ASO the perfpmr.sh scripts do not collect the ASO logs. You will need to set ASO into debug mode
asoo –o debug_level=3
After that you will need to recreate the problem and then reset ASO to normal logging
asoo –o debug_level=0
Then you can forward the aso_debug.out file to IBM support.

Summary
ASO can provide great benefits for tuning and it is free with AIX. It monitors the workloads on a busy system and dynamically moves workloads around to gain the best affinity within the LPAR. It helps administrators avoid having to make WLM and RSET changes to attain affinity and it is very easy to use. It can also be turned off immediately if the performance results are not what is expected. It is particularly useful for multi-threaded long running processes that are running on larger systems, especially the 770 and above which have multiple nodes, although any of the multi-socket systems can also benefit from it. It is recommended that administrators test out ASO on test LPARs to see if it potentially offers benefits for performance.

References

- **PowerVM User Group – look at sessions 23 and 26 on ASO, DSO and DPO**
- **http://tinyurl.com/newUK-PowerVM-VUG**
- **asoo Command**
  http://pic.dhe.ibm.com/infocenter/aix/v7r1/index.jsp?topic=%2Fcom.ibm.aix.cmds%2Fdoc%2Faixcmds1%2Fasoo.htm
- **SG24-8080 – Power Systems Performance Guide – Implementing and Optimizing**
  http://www.redbooks.ibm.com/redbooks/pdfs/sg248080.pdf
- **SG24-8079 – Power 7 and 7+ Optimization and Tuning Guide**
  http://www.redbooks.ibm.com/redbooks/pdfs/sg248079.pdf
- **Redbook Tip on Maximizing the Value of P7 and P7+ through Tuning and Optimization**
  http://www.redbooks.ibm.com/technotes/tips0956.pdf