AIX Version 6.1 tools overview

Handy tools for performance, virtualization, security and development

Authors
Michel Riviere
Ramesh Chitor
ISV Business Strategy and Enablement
March 2008
Version 1.2

Table of contents

Abstract

This white paper provides guidelines for developers and independent software vendors (ISVs) who are interested in using certain unique IBM POWER6 processor and IBM AIX 6 operating-system tools in their applications. The combination of AIX 6 and the POWER6 processor can put system performance and efficiency into a new realm.

Introduction

When it comes to tools, an analogy between sculpture and software draws an easy comparison. An artist does not just start chipping away at a block of white marble with any old chisel. Precision tools are necessary to create a masterpiece. The same is true in the software world. For example, for a database application or a business-intelligence piece of software that resides on top of a customer-relationship management (CRM) application, as the complexity increases, the quality of tools must also increase to produce the desired results. Thus, it is critical to understand the nature of a tool before you pick one for your application.

The IBM® POWER6™ microprocessor introduces several key features. That is why the IBM AIX® Version 6.1 version of the operating system delivers some major functions and tools. This white paper provides details on some of the salient tools that are available in the performance, virtualization and software-development areas.

POWER6 microprocessor

IBM has a long history of leveraging technology to build high-performance, reliable systems. In the past several years, IBM has increasingly applied this expertise to UNIX® servers and the AIX operating system. The IBM POWER6 processor-based IBM System p™ 570 model continues this heritage with industry-leading performance across a broad spectrum of industry-standard benchmarks. See the IBM System p 570 Web site that presents some of these performance details (**ibm.com**/systems/p/hardware/midrange/570m/perfdata.html).

The POWER6 microprocessor is the IBM follow-on to the IBM POWER5™ processor. The POWER6 processor, which was released on June 8, 2007, has the following characteristics:

- Approximately 790 million transistors that are 341 mm² large and are fabricated on a 65 nm processor
- 3.5, 4.2 and 4.7 GHz clock speed
- Dual-core processor
- 128 KB of L1 cache (64 KB data and 64 KB instructions)
- Eight-way, set-associative design
- Two-stage pipeline that supports two independent, 32-bit reads or one 64-bit write per cycle
- 4 MB semishared L2 cache for each core, where the cache is assigned to a specific core, but the other has fast access to it
- 32 MB large L3 cache that is shared by the two cores (the L3 cache is off die, using an 80 GBps bus)

Each core has two integer units, two binary floating-point (BFP) units, and a decimal floating-point (DFP) unit, and is capable of two-way symmetric multithreading (SMT). The BFP unit incorporates many microarchitecture, logic, circuit, latch and integration techniques to achieve a 6-cycle, 13-FO4 pipeline. POWER6 has hardware support for decimal arithmetic and includes the first DFP unit that is integrated in

silicon. More than 50 new floating-point instructions handle the decimal math and conversions between binary and decimal.

There is an AltiVec (single instruction, multiple data, or SIMD) unit on the POWER6 processor, and the processor is fully compliant with the new POWER ISA v.2.03 specification. The POWER6 processor also takes advantage of Virtual Vector Architecture (ViVA-2), which enables the combination of several POWER6 nodes to act as a single vector processor.

The POWER6 processor can connect to up to 31 other processors by using two internode links (50 GBps), and supports up to 1024 virtual partitions. The POWER6 processor comes in multiple chip modules (MCMs), as does the POWER5 processor, with up to four processor dies and associated external L3 cache on a single substrate.

The POER6 processor supports the IBM AIX 5L™ 5.2 TL10, AIX 5L 5.3 TL06 and AIX V6.1 operating-system releases, as well as the Linux® operating-system distributions from Linux partners Red Hat and Novell. Renowned for their computing power, IBM System p models and workstations support user needs across a broad range of applications, including transaction processing, Web publishing, data mining, systems management and others. This family of 1- to 64-core IBM POWER™ processor-based systems is designed to provide leadership features for high performance, availability, scalability and dynamic-resource allocation. Unique IBM virtualization features allow users to process more information on a single server — creating the potential to save on the total cost of system ownership.

System p products are designed for smaller and midsized businesses as well as large enterprises that are using UNIX platforms. These systems use AIX, which is the IBM open UNIX operating system, and also support thousands of Linux applications. System p models are powered by IBM leadership technology, including the POWER6 microprocessor, currently the world's fastest chip, which is built using IBM state-of-the-art 65-nanometer process technology. At 4.7 GHz, the dual-core POWER6 processor doubles the speed of the previous-generation POWER5 processor and uses nearly the same amount of electricity to run and cool it. This means that it is possible to use the new processor to either increase performance by 100 percent or cut power consumption virtually in half.

The POWER6 processor provides for hardware-partition migration, known as *live partition migration (LPM)*, which is totally different from *live application mobility (LAM*), which is an AIX 6 feature. AIX V6.1 fully exploits the POWER6 processor. Some of the POWER6 processor features that AIX 6 delivers include kernel exploitation of Storage Keys and automatic page-size optimization.

AIX V6.1 runs on systems that are based on the IBM POWER4™ processor, the IBM PowerPC® 970 processor, the IBM POWER5 processor and the latest generation of POWER processor, POWER6. Most of the new features of AIX V6.1 are available on the earlier POWER platforms, but the greatest capability is delivered on systems that are based on the new POWER6 processors. The AIX operating system is designed for the IBM System p and IBM System i™ product lines, as well as IBM BladeCenter® blades that are based on IBM Power Architecture® technology and IBM IntelliStation® POWER workstations.

New AIX V6.1 features

AIX Version V6.1 is the next IBM generation of the well-proven, scalable, open standards-based UNIX® operating system. It includes a large set of new features for virtualization, security, availability and manageability — designed to make AIX 6 even more flexible, secure and available than previous versions. It is also built to fully leverage POWER6 technology and virtualization to help deliver superior performance, increase system usage and efficiency, provide for easy administration, and reduce total costs. See the following highlights of what is new in AIX Version 6.1.

Enhanced virtualization features

Workload partitions: AIX 6 introduces a new, software-based, virtualization approach called *workload partitions (WPARs)* that complements the existing IBM systems' logical partitions (LPARs) by reducing the number of operating-system images that system administrators must manage when consolidating workloads. WPARs enable the consolidation of multiple applications inside of a single running instance of AIX 6.

Adding WPARs to an operating-system instance provides a new level of system-virtualization capability. WPARs allow you to checkpoint and restart the execution of applications without modifying application code. WPAR technology is a software-based partitioning solution and has no dependency on any hardware features. The part of the AIX 6 operating system that hosts the WPARs is called the *global environment*.

An important feature of WPARs is their ability to be relocated from one LPAR to another LPAR, irrespective of whether the LPAR is being located within the same physical server or not. For most applications, WPARs appear as a booted instance of the AIX operating system. There are two types of WPARs: system WPARs and application WPARs.

- A system WPAR has a dedicated, writable file system and can share global environments in /usr and /opt in read-only mode. A system WPAR is like a typical AIX operating-system environment.
- An application WPAR is reserved for an application or a group of applications that you can start by using only one command on an AIX command-line interface. Such a command is passed as an argument to the AIX wparexec command that, in turn, creates an application WPAR. An application WPAR cannot run system-services daemons, such as inetd. Also, it is not possible to remotely log in to an application WPAR.

You can separately administer each WPAR from other WPARs in the system, including separate security and root-level user processes. WPARs obtain a regulated portion of the system resources that are available to the instance of AIX 6 and share the AIX 6 resources, such as kernel and I/O resources.

Live application mobility: It is possible to move WPARs from one system to another without restarting the application or causing significant disruption to the application user. You enable this capability, called *live application mobility (LAM)*, through a separately offered licensed-program product, IBM Workload Partitions Manager for AIX. (**Note:** This program product is discussed later in this white paper.) Both system WPARs and application WPARs are supported under LAM. LAM is available in POWER4, POWER5 and POWER6 processor-based systems. (**Note:** The POWER6 processor also provides a hardware-partition migration process that is known as *live partition migration* [*LPM*]. LPM is totally different from LAM, which is a software-based AIX 6 feature. This is discussed later in this white paper.)

Security features

Role-based access control: This provides improved security and manageability by allowing administrators to grant authorization for management of specific AIX resources to users other than root users — by associating those resources with a role that is then associated with a particular system user. You can also use role-based access control (RBAC) to associate specific management privileges with programs, which can reduce the need to run those programs either under the root user or through setuid.

Trusted AIX: This extends the AIX 6 security capabilities by integrating compartmentalized, multilevel security into the base operating system. You implement Trusted AIX as an installation option to meet critical government and private-industry security requirements.

Encrypting file system: The IBM Journaled File System Extended (JFS2) adds even greater data security with the ability to encrypt file-system data. You can select from a number of encryption algorithms. You can back up the encrypted data in encrypted format, reducing the risk of compromised data if the backup media is lost or stolen. The encrypting file system can even prevent the compromise of data to root-level users.

AIX Security Expert LDAP integration: AIX Security Expert was introduced with the Technology Level 5 update to the AIX V5.3 operating system, and provides clients with the capability to manage more than 300 system-security settings from a single interface. AIX Security Expert has been enhanced in AIX 6 with an option to store security templates directly in a Lightweight Directory Protocol (LDAP) directory — simplifying the implementation of a consistent security policy across an entire enterprise.

Secure by Default installation option: The AIX 6 installation process offers a new option, *Secure by Default*, that enables only the minimal number of system and network services to provide the maximum amount of security. Secure by Default works best when you use it in conjunction with AIX Security Expert to tightly control the security configuration of each system.

These security features are discussed in greater detail later in this white paper.

Availability and manageability features

Kernel support for Storage Keys: This AIX 6 feature brings mainframe-inspired reliability capabilities to the UNIX environment for the first time. Enabled by POWER6, Storage Keys reduce the number of intermittent outages associated with undetected memory overlays in the AIX kernel. Applications can also use the Storage Key feature to increase the reliability of large, complex programs that run under AIX 5.3 or AIX 6.

Concurrent AIX kernel update: This feature provides a new capability to deliver some kernel updates as interim fixes that do not require a system reboot to be put into effect. This can reduce the number of unplanned outages that might otherwise be required to maintain a secure, reliable system.

Dynamic tracing: AIX 6 provides a new dynamic-tracing capability to simplify the task of debugging complex system or application code. This facility is introduced through a new tracing command, *probevue*, that allows you to dynamically insert trace breakpoints in existing code without having to recompile the code. (**Note:** The ProbeVue facility is discussed in detail later in this white paper.)

Enhanced software first-failure data capture (FFDC): A key innovation to improve the reliability, availability and serviceability of AIX is the introduction of FFDC technology. This concept is borrowed from the mainframe world; it gathers diagnostic data about problems as they occur, dramatically reducing the need to recreate the problem (and impacting performance and availability) to generate diagnostic information.

Systems-management and enhanced virtualization tools

For managing and maintaining your operating-system configuration, AIX offers a number of choices, from the command line and menu-driven System Management Interface Tool (SMIT), to a variety of remote management solutions, such as *IBM Systems Director Console for AIX* (*Console*) and *IBM Web-based System Manager*. *Workload Manager* provides a policy-based approach to managing the use of resources by applications that run inside the AIX operating system. Workload Manager has been part of AIX since AIX V4.3.3. *IBM Workload Partitions Manager for AIX* is now available to facilitate management of WPARs.

IBM Systems Director Console for AIX

The Console is a new management interface that allows administrators to manage AIX V6.1 remotely through a browser. It provides responsive Web access to common systems-management tools, such as SMIT. The System Director Console for AIX is a highly scalable set of multihost tools and is included with AIX 6. It does not require any other Web server or other software.

IBM Web-based System Manager

IBM Web-based System Manager uses a management console that is capable of administering multiple AIX operating-system hosts from AIX, PC or Linux remote clients. The new plug-in architecture makes it easier to extend the application suite. In addition, Web-based System Manager supports dynamic monitoring of system events through its integration of Resource Monitoring and Control (RMC), which was developed for management of IBM UNIX systems. Starting with AIX V6.1, you can also access Web-based System Manager through IBM Systems Director Console for AIX.

Workload Manager

Workload Manager helps you manage multiple applications, workloads and resources by allowing you to control the amount of resources, such as processor capacity or memory, that an application can use. In AIX V6.1, Workload Manager is the primary infrastructure for regulating the use of resources for WPARs. When used in conjunction with Workload Manager, WPARs enable you to easily consolidate multiple workloads in a single AIX instance. Workload Manager also allows you to enforce business priorities on your workloads to help ensure that the resource requirements of less-critical jobs in the system do not have an impact on the performance of critical applications. You can easily apply individually tailored resource-allocation profiles, which allows you to spend less time on routing workload tasks.

IBM Workload Partitions Manager for AIX

With the addition of WPARs in AIX V6.1, a new licensed program product, *IBM Workload Partitions Manager for AIX*, facilitates management of WPARs on multiple systems. You can create, clone, start, stop and monitor WPARs on multiple systems from a single interface. Workload Partitions Manager also enables live application mobility, which allows you to move a WPAR from one system to another without restarting the application. LAM help reduce application downtime, improves workload balancing across systems, and provides energy savings by moving workloads from an underused system and then shutting it down.

Performance tools

This section presents the most popular features and new tools that are available for AIX V6.1. These tools allow you to get the most out of your System p platform and help you control, monitor and diagnose application performance. They also help you improve the reliability and availability of your system.

AIX tracing tools overview

AIX V6.1 includes several tracing facilities.

AIX system trace is the main trace facility on AIX; it is designed for tracing inside the kernel and kernel extensions. However, this facility also supports user-defined tracing in application code. It is based on compiled-in static-trace hooks and is only enabled when needed. By default, all trace hooks are enabled when tracing is turned on. However, there are options to enable only a set of trace hooks or to disable some specific trace hooks. Both user and kernel tracing share the same system buffers. So, the application-level trace data is copied to the system buffer.

Lightweight memory trace (LMT) allows you to trace only key AIX kernel events and is not available in user mode. LMT is also based on compiled-in static trace hooks. It is enabled by default, but it uses a lightweight mechanism to record trace data, so its impact on performance is minimal. Trace data is sent to per-processor buffers and stays in memory until overwritten. There are commands to extract the traced data, and you use the same tools as AIX system-trace tools to display the traced data. Alternatively, you can display it with the AIX kdb command or extract it from a system dump.

Truss enables tracing of all system calls and, optionally, all library calls that are run by a specific process. Thus, traced events are limited to system subroutine calls. Trace output consists of parameters that pass into, and the values that return from, each system (and library) call. This is directly sent to standard error mechanisms for that process. There are no systemwide buffers and no mechanism to save the trace data.

Component trace (CT) is a new facility that became available in AIX 5.3 TL06. You can use the component tracing facility as an additional filter on AIX system trace. You can also use it exclusively to provide in-memory tracing. You can direct it to use either system-wide LMT buffers or component-specific buffers to save the trace data. Its primary purpose, similar to LMT, is to collect FFDC data for debugging purposes.

POSIX trace AIX V6 implements the POSIX trace system in order to trace user applications. These facilities allow a process to select a set of trace-event types to activate a trace stream of the selected trace events as they occur in the flow of execution and also to retrieve the recorded trace events. Similar to system trace, POSIX trace also depends on precompiled in-trace hooks in the application that is being instrumented.

AIX POSIX trace in AIX 6.1

With AIX V6.1, the process of managing AIX POSIX trace streams and events involves a daemon that is named *posixtrace*. It is the only process the operating system has to implement. Because posixtrace creates a trace stream for all processes and records all events, posixtrace belongs to the root user. The posixtrace daemon runs as root.

1. The posixtrace daemon is started by the first library load through the associated library-initialization routine mechanism. This mechanism is implemented through the -binitfini binder option. Thus, the libposixtrace.a library is linked with the option -binitfini:posix_trace_libinit.
2. This posix_trace_libinit routine binds a dedicated socket to the file /var/adm/ras/.pxt_sock and listens for a connection from the instrumented code that is linked with the libposixtrace.a library.
3. Another file, /var/adm/ras/.start_lock, is used as a lock file to prevent several starts of the posixtrace daemon.
4. When the main daemon thread determines that no thread is left, it closes the socket, unlocks and unlinks /var/adm/ras/.pxt_sock, and then exits.

AIX V6.1 dynamic tracing or ProbeVue

Another new tracing facility tool in AIX V6.1 is ProbeVue. It allows users to dynamically specify trace points and also to dynamically provide the actions that should run at the specified trace points. The tracing specifications are written in the VUE programming language, which is the primary interface for the user to specify or control dynamic tracing on AIX. VUE shares several similarities with the C programming language and shell script, but also has unique features.

ProvbeVue provides a complimentary tracing tool in addition to the AIX V6.1 static-tracing tools. You can use ProbeVue for performance analysis as well as for debugging problems. It runs safely on production systems and provides protection against errors in the instrumentation code. ProbeVue does not require modification or recompilation of the user application that is to be traced.

With the ProbeVue dynamic-tracing facility, you can investigate a production system: ProbeVue captures the execution data without installing dedicated instrumented versions of applications or the kernel that, otherwise, would require you to interrupt the service for the application relaunch or server reboot. Additionally, ProbeVue helps to find the root cause of troubles that happen only on long-running jobs where unexpected accumulation of data, queue overflows and other defects in the application or kernel are revealed only after many days or months of usage.

ProbeVue performs privileged operations that are restricted to privileged users. The correct authorization and privileges level must be granted if such operations are triggered when running a VUE script. Only the root user or users with *aix.ras.probevue.manage* authorization can update ProbeVue parameters and view ProbeVue sessions. See the topic *Privileged command database* in the IBM Systems Information Center (http://publib.boulder.ibm.com/infocenter/systems/index.jsp?topic=/com.ibm.aix.security/doc/security/rbac_priv_cmd_db.htm&tocNode=int_38195).

When you specify the probevue command with a vue script, the command enables the tracing that is specified in the script and produces the tracing output. AIX V6.1 supports the following probe types:

- User-function tracing (UFT) probes for entry or exit of user-written function code

  To set a probe to monitor entry into a function helloworld() for a process with PID 12345, use the following probe:
  `@@uft:12345:*:helloworld:entry`

- System-call entry and exit probes (or syscall probes)

  To set a probe for a system call on entry and exit, use the following syntax:
  `@@syscall:`*`PID`*`:`*`SystemCallName`*`:[entry/exit]`

- Probes that fire at specific-time intervals (or interval probes)

  To fire every 100 milliseconds, use the following interval probe:
  `@@interval:*:clock:100`

Figure 1 shows a very simple example of ProveVue usage.

```
#!/usr/bin/probevue
@@BEGIN
{ /* block to be interpreted at the beginning of the execution of the script */
int count;
int total;
count =0;
total = 0;
}

@@syscall:*:open:entry    /* trap entry to open system call for any process ID */ {
   count++;
   total++;
}

@@interval:*:clock:1000                                                    /* Probe
fired every 1000 milliseconds */
{
   printf("Number of open system since the last count = %d\n", count);
   count=0;
}

@@END  /* Block executed when the script is terminated via Control-C */
{
    printf("\nTotal number of file opened = %d\n",total);
}
```

Figure 1. Example of ProveVue

Figure 2 shows the output from the previous example.

```
# ./readcount.e
Number of open system since the last count = 0
Number of open system since the last count = 15
Number of open system since the last count = 2
Number of open system since the last count = 0
Number of open system since the last count = 10
^C
Total number of file opened = 27
```

Figure 2. ProveVue output