What: /sys/devices/system/cpu/
Date: pre-git history
Contact: Linux kernel mailing list <linux-kernel@vger.kernel.org>
Description:
A collection of both global and individual CPU attributes

Individual CPU attributes are contained in subdirectories named by the kernel's logical CPU number, e.g.:

/sys/devices/system/cpu/cpu#/

What: /sys/devices/system/cpu/sched_mc_power_savings
/sys/devices/system/cpu/sched_smt_power_savings
Date: June 2006
Contact: Linux kernel mailing list <linux-kernel@vger.kernel.org>
Description: Discover and adjust the kernel's multi-core scheduler support.

Possible values are:

0 - No power saving load balance (default value)
1 - Fill one thread/core/package first for long running threads
2 - Also bias task wakeups to semi-idle cpu package for power savings

sched_mc_power_savings is dependent upon SCHED_MC, which is itself architecture dependent.

sched_smt_power_savings is dependent upon SCHED_SMT, which is itself architecture dependent.

The two files are independent of each other. It is possible that one file may be present without the other.

Introduced by git commit 5c45bf27.

What: /sys/devices/system/cpu/kernel_max
/sys/devices/system/cpu/offline
/sys/devices/system/cpu/online
/sys/devices/system/cpu/possible
/sys/devices/system/cpu/present
Date: December 2008
Contact: Linux kernel mailing list <linux-kernel@vger.kernel.org>
Description: CPU topology files that describe kernel limits related to hotplug. Briefly:

kernel_max: the maximum cpu index allowed by the kernel configuration.

offline: cpus that are not online because they have been HOTPLUGGED off or exceed the limit of cpus allowed by the kernel configuration (kernel_max above).

online: cpus that are online and being scheduled.

possible: cpus that have been allocated resources and can be

brought online if they are present.

present: cpus that have been identified as being present in the system.

See Documentation/cputopology.txt for more information.

What: /sys/devices/system/cpu/probe
/sys/devices/system/cpu/release
Date: November 2009
Contact: Linux kernel mailing list <linux-kernel@vger.kernel.org>
Description: Dynamic addition and removal of CPU's. This is not hotplug removal, this is meant complete removal/addition of the CPU from the system.

probe: writes to this file will dynamically add a CPU to the system. Information written to the file to add CPU's is architecture specific.

release: writes to this file dynamically remove a CPU from the system. Information writtento the file to remove CPU's is architecture specific.

What: /sys/devices/system/cpu/cpu#/node
Date: October 2009
Contact: Linux memory management mailing list <linux-mm@kvack.org>
Description: Discover NUMA node a CPU belongs to

When CONFIG_NUMA is enabled, a symbolic link that points to the corresponding NUMA node directory.

For example, the following symlink is created for cpu42 in NUMA node 2:

/sys/devices/system/cpu/cpu42/node2 -> ../../node/node2

What: /sys/devices/system/cpu/cpu#/node
Date: October 2009
Contact: Linux memory management mailing list <linux-mm@kvack.org>
Description: Discover NUMA node a CPU belongs to

When CONFIG_NUMA is enabled, a symbolic link that points to the corresponding NUMA node directory.

For example, the following symlink is created for cpu42 in NUMA node 2:

/sys/devices/system/cpu/cpu42/node2 -> ../../node/node2

What: /sys/devices/system/cpu/cpu#/topology/core_id
/sys/devices/system/cpu/cpu#/topology/core_siblings
/sys/devices/system/cpu/cpu#/topology/core_siblings_list
/sys/devices/system/cpu/cpu#/topology/physical_package_id

/sys/devices/system/cpu/cpu#/topology/thread_siblings
/sys/devices/system/cpu/cpu#/topology/thread_siblings_list
Date: December 2008
Contact: Linux kernel mailing list <linux-kernel@vger.kernel.org>
Description: CPU topology files that describe a logical CPU's relationship to other cores and threads in the same physical package.

One cpu# directory is created per logical CPU in the system, e.g. /sys/devices/system/cpu/cpu42/.

Briefly, the files above are:

core_id: the CPU core ID of cpu#. Typically it is the hardware platform's identifier (rather than the kernel's). The actual value is architecture and platform dependent.

core_siblings: internal kernel map of cpu#'s hardware threads within the same physical_package_id.

core_siblings_list: human-readable list of the logical CPU numbers within the same physical_package_id as cpu#.

physical_package_id: physical package id of cpu#. Typically corresponds to a physical socket number, but the actual value is architecture and platform dependent.

thread_siblings: internel kernel map of cpu#'s hardware threads within the same core as cpu#

thread_siblings_list: human-readable list of cpu#'s hardware threads within the same core as cpu#

See Documentation/cputopology.txt for more information.

What: /sys/devices/system/cpu/cpuidle/current_driver
/sys/devices/system/cpu/cpuidle/current_governer_ro
Date: September 2007
Contact: Linux kernel mailing list <linux-kernel@vger.kernel.org>
Description: Discover cpuidle policy and mechanism

Various CPUs today support multiple idle levels that are differentiated by varying exit latencies and power consumption during idle.

Idle policy (governor) is differentiated from idle mechanism (driver)

current_driver: displays current idle mechanism

current_governor_ro: displays current idle policy

See files in Documentation/cpuidle/ for more information.

What: /sys/devices/system/cpu/cpu#/cpufreq/*

Date: pre-git history
Contact: cpufreq@vger.kernel.org
Description: Discover and change clock speed of CPUs

Clock scaling allows you to change the clock speed of the CPUs on the fly. This is a nice method to save battery power, because the lower the clock speed, the less power the CPU consumes.

There are many knobs to tweak in this directory.

See files in Documentation/cpu-freq/ for more information.

In particular, read Documentation/cpu-freq/user-guide.txt to learn how to control the knobs.

What: /sys/devices/system/cpu/cpu*/cache/index*/cache_disable_X
Date: August 2008
KernelVersion: 2.6.27
Contact: mark.langsdorf@amd.com
Description: These files exist in every cpu's cache index directories. There are currently 2 cache_disable_# files in each directory. Reading from these files on a supported processor will return that cache disable index value for that processor and node. Writing to one of these files will cause the specificed cache index to be disabled.

Currently, only AMD Family 10h Processors support cache index disable, and only for their L3 caches. See the BIOS and Kernel Developer's Guide at

http://www.amd.com/us-en/assets/content_type/white_papers_and_tech_docs/31116-Public-GH-BKDG_3.20_2-4-09.pdf
for formatting information and other details on the cache index disable.
Users: joachim.deguara@amd.com