

Re: [PATCH 1/2] [REPOST] mm: show node to memory section relationship with symlinks in sysfs

Re: [PATCH 1/2] [REPOST] mm: show node to memory section relationship with symlinks in sysfs

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2008-10/msg05246.html>

- *From:* Yasunori Goto <y-goto@xxxxxxxxxxxxxxxx>
 - *Date:* Tue, 14 Oct 2008 20:54:21 +0900
-

On Fri, Oct 10, 2008 at 04:32:30PM -0700, Andrew Morton wrote:

On Fri, 10 Oct 2008 16:18:44 -0700
Gary Hade <garyhade@xxxxxxxx> wrote:

On Fri, Oct 10, 2008 at 02:59:50PM -0700, Andrew Morton
wrote:

On Fri, 10 Oct 2008 14:33:57 -0700
Gary Hade <garyhade@xxxxxxxx>
wrote:

On Fri, Oct 10, 2008 at
12:42:39PM -0700, Andrew
Morton wrote:

On Thu, 9
Oct 2008
12:21:15
-0700
Gary Hade
<garyhade@xxxxxxxx>
wrote:

Show
node
to
memory
section
relationship
with
symlinks
in

Re: [PATCH 1/2] [REPOST] mm: show node to memory section relationship with symlinks in sysfs

sysfs

Add

/sys/devices/system/node/nodeX/memoryY

symlinks

for

all

the

memory

sections

located

on

nodeX.

For

example:

/sys/devices/system/node/node1/memory135

->

../memory/memory135

indicates

that

memory

section

135

resides

on

node1.

I'm not
seeing here

a
description
of why the
kernel
needs this
feature.

Why is it
useful?

How will it
be used?

What value
does it have
to

our users?

Sorry, I should have
included that. In our case, it
is another
small step towards eventual
total node removal. We will

Re: [PATCH 1/2] [REPOST] mm: show node to memory section relationship with symlinks in sysfs 2

Re: [PATCH 1/2] [REPOST] mm: show node to memory section relationship with symlinks in sysfs

need to know which memory sections to offline for whatever node is targeted for removal. However, I suspect that exposing the node to section information to user-level could be useful for other purposes. For example, I have been thinking that using memory hotremove functionality to modify the amount of available memory on specific nodes without having to physically add/remove DIMMs might be useful to those that test application or benchmark performance on a multi-node system in various memory configurations.

hm, OK, thanks. It does sound a bit thin, and if we merge this then not only do we get a porkier kernel,

Would you feel the same about the size increase if patch 2/2 (include memory section subtree in sysfs with only sparsemem enabled) was withdrawn?

Without patch 2/2 the size increase for non-Sparsemem or Sparsemem wo/memory hotplug kernels is extremely small. Even for memory hotplug enabled kernels there is only a little extra code in ./drivers/base/node.o which only gets linked into NUMA enabled kernels. I can gather some numbers if necessary.

Size is probably a minor issue on memory-hotpluggable machines.

Re: [PATCH 1/2] [REPOST] mm: show node to memory section relationship with symlinks in sysfs 3

Re: [PATCH 1/2] [REPOST] mm: show node to memory section relationship with symlinks in sysfs

we also get a new userspace interface which we're then locked into.

True.

That's a bigger issue. The later we leave this sort of thing, the more information we have.

I understand your concerns about adding possibly frivolous interfaces but in this case we are simply eliminating a very obvious hole in the existing set of memory hot-add/remove interfaces. In general, it makes absolutely no sense to provide a resource add/remove mechanism without telling the user where the resource is physically located. i.e. providing the `_maximum_` possible amount of location information available for the add/remove controllable resource. This is especially critical for large multi-node systems and for resources that can impact application or overall system performance.

The kernel already exports node location information for CPUs (e.g. `/sys/devices/system/node/node0/cpu0 -> ../../cpu/cpu0`) and PCI devices (e.g. `/devices/pci0000:00/0000:00:00.0/numa_node`). Why should memory be treated any differently?

The memory hot-add/remove interfaces include physical device files (e.g. `/sys/devices/system/memory/memory0/phys_device`) which are not yet fully implemented. When systems that support removable memory modules force this interface to mature, node location information will become even more critical. This feature will not be very useful on multi-node systems if the user does not know what node a specific memory module is installed in. It may be possible to encode the node ID into the string provided by the `phys_device` file but a more general node to memory section association as provided by this patch is better since it can be used for other purposes.

Sorry for late response.

Our Fujitsu box can hot-add a node. This means a user/script has to find which memory sections and cpus belong to added node when node hot-add is executed.

Current my hotplug script is very poor. It onlines all offlined cpus and memories. However if user offlined one memory section intentionally due to memory error message, the script can't understand it is intended, and hot-add the error section. I think this is one of reason why this link is necessary.

Re: [PATCH 1/2] [REPOST] mm: show node to memory section relationship with symlinks in sysfs 4

Re: [PATCH 1/2] [REPOST] mm: show node to memory section relationship with symlinks in sysfs

I think not only node id, but I would like to show `_PXM` value of ACPI to specify physical position of the node. Because node id is decided by OS at boot time (and hot-add time) to make it consecutive.

(This is historical inheritance when there is no macro like `for_each_online_cpus()`.)

If a system has 2 nodes whose `_PXM` values are 0 and 3, and boot up, then kernel make node id 0 and 1 for them, and when hot-add a node whose `_PXM` value is 1, then new node id will be 2.

```
_PXM 0 1 3  
node id 0 2 1
```

When user reboot the system, then node id will be followings.
User will be confused by this.

```
_PXM 0 1 3  
node id 0 1 2
```

Current kernel may allow "node id = `_PXM`", because `for_each_xxx_node()` works well now. But I'm not sure....

Bye.

--

Yasunori Goto

--

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to `majordomo@xxxxxxxxxxxxxxxxxx`

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>

Re: [PATCH 1/2] [REPOST] mm: show node to memory section relationship with symlinks in sysfs 5