

# Behaviour of the VM on a embedded linux

---

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2008-08/msg10026.html>

---

- *From:* "Wappler Marcel" <[Marcel.Wappler@xxxxxxxxxxxxx](mailto:Marcel.Wappler@xxxxxxxxxxxxx)>
  - *Date:* Fri, 22 Aug 2008 14:31:43 +0200
- 

Hi Folks,

I'm trying to figure out whats going on an embedded system I have to deal with. It's running a 2.6.24.7 kernel on 32 MBytes of RAM.

There is no swapping. There are some daemons and shells running and – a big monolithic c++ application.

The application runs a lot of pthreads on different real time priority levels. It looks like the application consumes a huge amount of real memory in contrast to the assumption, that large code size is no problem due to paging out pages with unused code.

I'm not so familiar with the VM internals of the Linux kernel – I only have some general ideas about systems using MMUs. So my assumption was, that even if a application consists of a lot of code, almost all memory pages containing executable code can be paged out in favour of pages which are needed to store data (stacks, heap, slab). I tried to figuring this out through reading in different books like "understanding the linux kernel" but those parts I was not able to catch anywhere.

Now there is the idea, that pages which contain code of pthreads running at real time priority cannot be paged out because of the real time demand.

Is this true?

Thanks a lot,  
Marcel

PS: please CC me personally on replies  
PPS: I'm hapy about every plausible idea

—

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to [majordomo@xxxxxxxxxxxxxxx](mailto:majordomo@xxxxxxxxxxxxxxx)  
More majordomo info at <http://vger.kernel.org/majordomo-info.html>  
Please read the FAQ at <http://www.tux.org/lkml/>