Documenting Linux MM for fun and for ... fun

Mike Rapoport
<rppt@linux.ibm.com>

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 688386.
Q: How do I map a page into a process address space?

A: It should be documented somewhere
   ... hmm, read the source
For the reader

- Reading docs is easier than reading the code
- Guides are better than bad code examples
- Looks professional ;-)

For an experienced writer

- Make sure code matches the intention
- Revisiting after week, month, year
- Help others
Even more value

For a newbie writer

- Better understanding of the code
- Helps to find the itch to scratch
- Help others
MM documentation a while ago

- Good coverage for MM-related `sysctl`'s and `/proc` bits
- Limited API description
- Hodgepodge of random documents
User visible interfaces

- sysctl’s and other bits in /proc are documented
  - Documentation/sysctl/vm.txt
  - Documentation/filesystems/proc.txt
  - Difficult for an unprepared reader

- /sysfs is very partially covered
  - Documentation/ABI/testing/sysfs-kernel-mm-{hugepages, ksm, swap}

- Additional descriptions buried in Documentation/vm
Internal APIs

● A section in Documentation/core-api/kernel-api.rst
  ○ First appeared in 2.4.0-test2pre1
  ○ Handful of updates since then
    ■ Several additions of documented sources
    ■ Fixes for file renames
    ■ ReST conversion

● A paragraph in Documentation/kernel-hacking/hacking.rst
  ○ The last update was in 2005

● More information buried in Documentation/vm
• Somewhat exists
  ○ 40% of mm files have kernel-doc comments
  ○ Roughly 10-15% of functions are documented
  ○ Some comments just lack the second * in /**
• Comments are not always consistent with the code
• Most of the existing kernel-doc is not linked
• Omnium-Gatherum
  ○ Random notes
  ○ User space interface descriptions
  ○ Design documents
  ○ Internal API user guides

• User guides are mixed with implementation details
• Hard to identify outdated and irrelevant parts
Start small

- White space
- Spelling, grammar, punctuation
- Comments formatting
- Add several kernel-doc comments
- Make comments and code agree
  - Gets broken again from time to time
- **ReSTify** Documentation/vm
- **Split user space interface descriptions to** Documentation/admin-guide
- **Move “Memory Management” API reference to a dedicated file**
  - Link GFP flags documentation
- **Brief concepts overview**
  - Make life easier for an unaware reader
- **Memory allocation guide**
  - Allocations in kernel are slightly more complex than malloc(size)
- **Description of boot time memory management**
  - bootmem, nobootmem, memblock
Challenges

- Stay focused
- Overwhelming amount of information
- “The best is the enemy of the good”
Current state

- User and admin guides

- API reference

- Internal documentaion
• Drop irrelevant content
• Move more parts to `admin-guide` or `core-api`
• Extend implementation descriptions
• Improve structure
  ○ Design docs
  ○ Guides
  ○ Debug aids
  ○ Notes
• Elaborate description of design and implementation
• kernel-doc additions
  ◦ prioritize by importance/usage frequency
• “Understanding the Linux Virtual Memory Manager” ReST version?
• **Request to verify docs consistency in** submitting-patches.rst
  ○ `kernel-doc -none -v`
  ○ `make W=1`

• **Teach** checkpatch.pl **about** kernel-doc

• **Add** `make *docs` **to LKP and other bots**
  ○ Thank you for the patch! Perhaps something to improve:
    warning: Function parameter or member 'ctx' not described in 'follow_page_mask'

• **Pay more attention to docs at review time**
Please, Document!
Thank you!