<previous description obsolete, deleted>

Virtual memory map with 4 level page tables:

0000000000000000 - 00007fffffffffff (=47 bits) user space, different per mm
hole caused by [48:63] sign extension
ffff800000000000 - ffff80ffffffffff (=40 bits) guard hole
ffff880000000000 - ffffc7ffffffffff (=64 TB) direct mapping of all phys. memory
ffffc80000000000 - ffffc8ffffffffff (=40 bits) hole
ffffc90000000000 - ffffe8ffffffffff (=45 bits) vmalloc/ioremap space
ffffe90000000000 - ffffe9ffffffffff (=40 bits) hole
ffffea0000000000 - ffffeaffffffffff (=40 bits) virtual memory map (1TB)
... unused hole ...
ffffffff80000000 - ffffffffa0000000 (=512 MB) kernel text mapping, from phys 0
ffffffffa0000000 - fffffffffff00000 (=1536 MB) module mapping space

The direct mapping covers all memory in the system up to the highest
memory address (this means in some cases it can also include PCI memory
holes).

vmalloc space is lazily synchronized into the different PML4 pages of
the processes using the page fault handler, with init_level4_pgt as
reference.

Current X86-64 implementations only support 40 bits of address space,
but we support up to 46 bits. This expands into MBZ space in the page tables.

-Andi Kleen, Jul 2004