

Re: [PATCH x86] [5/16] Replace hard coded reservations in x86-64 early boot code with dynamic table II

## Re: [PATCH x86] [5/16] Replace hard coded reservations in x86-64 early boot code with dynamic table II

---

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2008-01/msg01353.html>

---

- *From:* Andi Kleen <ak@xxxxxxx>
  - *Date:* Fri, 4 Jan 2008 16:03:58 +0100
- 

```
ERROR: use tabs not spaces
#92: FILE: arch/x86/kernel/e820_64.c:89:
+ ^I}$
```

Can't you just convert them using a script on new if you care so much about those?

Ok I can write such a script too, but then every new contributor would need too which will be a collective waste of time.

```
ERROR: use tabs not spaces
#135: FILE: arch/x86/kernel/e820_64.c:107:
+ ^I}$
```

total: 2 errors, 0 warnings, 308 lines checked

i'm not sure how you manage patches – if you have some 'refresh patch' command then you might want to stick a call to scripts/checkpatch.pl in there. I have such a check included in my quilt refresh shortcut.

#3:

```
+ int i;
+ struct early_res *r;
+ for (i = 0; i < MAX_EARLY_RES && early_res[i].end; i++) {
+ r = &early_res[i];
```

newline needed after the variables.

That's an Ingo only rule I disagree with and is actually not in CodingStyle  
When checkpatch.pl warns about it it is wrong and I would lobby for removing it.

Re: [PATCH x86] [5/16] Replace hard coded reservations in x86-64 early boot code with dynamic table II

Re: [PATCH x86] [5/16] Replace hard coded reservations in x86-64 early boot code with dynamic table II

#4:

```
void __init early_res_to_bootmem(void)
{
    int i;
    for (i = 0; i < MAX_EARLY_RES && early_res[i].end; i++) {
        struct early_res *r = &early_res[i];
        reserve_bootmem_generic(r->start, r->end - r->start);
    }
}
```

the variables definition here is totally misaligned.

Yes I fixed that now.

– style cleanliness is an admittedly secondary concept but it strengthens the most important, primary aspect of any code: readability and maintainability.

In a minor way -- it is far less important than let's say good high level comments or clear code flow.

e.g. this patch imho is a major improvement in code clarity and logic and makes previously quite fragile code much simpler and more approachable and easier changeable (I can say this because I was responsible for the original mess -- only excuse is that it has grown over time). And then when people don't see these advantages and just talk about tabs<-->spaces that is quite annoying.

– sloppy style is also often a statistical indicator for sloppy quality: code written in rush, not reviewed and not read well enough. I'm not implying anything like that about this patch of yours, but it's a general policy and you'd sure agree that writing 100% clean code is not that much harder than just writing good code.

It depends. e.g. I personally consider the newline after variables rule ugly for small functions and writing code for what I consider ugly style is harder. Also it was always my personal style to put a variable near its first use in the same paragraph (imho that makes it clearer) so

```
int foo;
for (foo = ...; foo ; foo) {
}
```

makes perfect sense to me. And I don't really see a reason to change

Re: [PATCH x86] [5/16] Replace hard coded reservations in x86-64 early boot code with dynamic table II

Re: [PATCH x86] [5/16] Replace hard coded reservations in x86-64 early boot code with dynamic table II

that.

I think all the cases you complained about were like this. Which means you didn't actually look at the code logic, just at some abstract rule that had nothing to do with the code flow. Which is bad.

I guess it's ok to add these newlines for larger functions and on those you should have paragraphs anyways to separate logically separate code blocks; but again that's not really something that can

Same for tabs versus spaces. That is something no human should need to care about. And it is something a machine can handle fine too anyways.

– cleanup is generally pushed back to patch submitters – this distributes better as it removes load from maintainers (and subsequent patch developers).

When it makes sense. e.g. for trailing white space at least Andrew (and I, don't know about you) always just removed those automatically on merge. No need to push a patch back for such a silly thing. Tabs versus Spaces is similar — something a machine can do fine without any human assistance.

If you really feel strongly about those the right way is to remove it from checkpatch.pl and just let the major maintainers run a script to convert those on all new lines automatically.

Also I really dislike the recent flurry of checkpatch.pl only patches on linux-kernel. Adding the whole file mode to it was a really bad idea. Style is only a minor ingredient to the real goal of good linux code and when people convert whole files everybody who has outstanding patching against these files is forced to redo them which is a large waste of time. Fixing style when something is changed anyways is ok on the other hand because then other patches will reject anyways. To borrow one of your favourite expressions it is pissing on the work of people who actually do non trivial changes to the tree.

I wish that disease would stop.

– it's also a matter of visual taste. People are more likely to fix and improve clean `_looking_` code than messy looking code.

Even more likely they are to improve code which is logically clean. The best coding style doesn't help if that's not the case. Ok it's not totally unimportant, but definitely not as much as you claim it to be. Also some minor coding style variants in the tree are not the end of the world for anybody.

Re: [PATCH x86] [5/16] Replace hard coded reservations in x86-64 early boot code with dynamic table II

Re: [PATCH x86] [5/16] Replace hard coded reservations in x86-64 early boot code with dynamic table II

-Andi

--

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

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

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

Re: [PATCH x86] [5/16] Replace hard coded reservations in x86-64 early boot code with dynamic table II