

Come contribuire al kernel Linux

Rigel Di Scala, Carmelo Sarta

Linux Day 2022



LDMI 2022



Chi siamo

Rigel

Engineering Manager, Red Hat 🎩

Primo computer: Commodore Vic 20 (1984)

Prima distro: Mandrake 7.2 (2001 - 2003)

github.com/zedr



Carmelo

Technical Graduate, Red Hat 🎩

Haven't distro-hopped in ~~42~~ 0 days.

github.com/Laiot



The screenshot captures a Linux desktop environment from the 2003/2004 period. The desktop is populated with several windows:

- Firefox Browser:** Displays the Gentoo Linux website, featuring the Gentoo logo and navigation links like 'News', 'About', 'Social Contract', 'Downloads', 'Forums', 'Lists', 'Screenshots', 'Bugs', 'Store', 'GWN', and 'Get Gentoo! Sponsors'. The main content area highlights 'Gentoo Linux 1.4 for PowerPC now available for pre-order' and 'Our new Gentoo Store is now open accepting pre-orders for Gentoo Linux 1.4 for PowerPC, PowerPC G3, PowerPC G4, and the KDE/Gentoo PowerPC LiveCD'. A 'Gentoo Weekly Newsletter' link is also present.
- Thunderbird Email Client:** Shows an inbox for 'stewart@snek.org' with a 'Welcome to Thunderbird Mail!' message. The message content includes: 'Thunderbird™ Mail is a powerful open-source mail and news client supporting advanced junk mail detection and other useful features... Features: E-mail and News groups, Support for multiple accounts, Junk Mail detection, Message Filters, HTML message support, Address Book... More Information: For frequently asked questions, For Thunderbird™ Mail Downloads, Powered by Gecko.' A chat window for '#Gentoo-Dev' is also visible, showing a conversation about the Gentoo Dev channel and a user named 'Blkdeath'.
- Terminal Window:** Displays system logs, including network connections and system messages. Visible text includes: '1002006027.890 404 192.168.0.21 TOP CLIENT: RE:RESUB:MISS:00:1942:GET:HTTP://www.gentoo.org/images/atom.png - DIRECT:212.162.48.146 image/png', '100550007.441 553 192.168.0.21 TOP CLIENT: RE:CHAT:MISS:00:5301:GET:HTTP://www.gentoo.org/images/netraverse-gentoo.gif - DIRECT:202.112.31.154 image/gif', and various 'Jul 26 15:15:07' system messages related to 'sshd' and 'sshd: disk abort or changed during operation'.
- Taskbar:** Shows icons for 'lirc (4884915)', 'Gentoo Linux - Gentoo', 'VMware Workstation', 'OpenOffice.org 1.1.4', 'X-Chat [2.0.2] - Blkdeath', 'K9 - The CD Krazor', and 'The GIMP'.
- Sticky Note:** A yellow sticky note in the bottom right corner contains the text '2003/2004 (2.4 → 2.6)'.

2003/2004 (2.4 → 2.6)

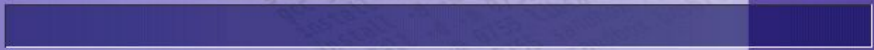




GEANTOO



2004.2



```
raid6: sse2x4 19019 MB/s
raid6: using algorithm sse2x4 (19019 MB/s)
md: raid6 personality registered as nr 8
md: multipath personality registered as nr 7
md: md driver 0.90.0 MAX_MD_DEVS=256, MD_SB_DISKS=27
device-mapper: 4.1.0-ioctl (2003-12-10) initialised: dm@uk.sistina.com
Advanced Linux Sound Architecture Driver Version 1.0.4 (Mon May 17 14:31:44 2004 UTC).
```

ALSA device list:

```
No soundcards found.
NET: Registered protocol family 2
IP: routing cache hash table of 4096 buckets, 32Kbytes
TCP: Hash tables configured (established 32768 bind 32768)
NET: Registered protocol family 1
NET: Registered protocol family 10
IPv6 over IPv4 tunneling driver
NET: Registered protocol family 17
md: Autodetecting RAID arrays.
md: autorun ...
md: ... autorun DONE.
```

```
RAMDISK: Compressed image found at block 0
VFS: Mounted root (ext2 filesystem) readonly.
Freeing unused kernel memory: 184k freed
EXT2-fs warning: checktime reached, running e2fsck is recommended
```

>> Loading modules...

```
:: Scanning for ehci-hcd...ehci-hcd loaded.
:: Scanning for uhci... module not found.
:: Scanning for usb-ohci... module not found.
:: Scanning for hid... module not found.
:: Scanning for usb-storage...usb-storage loaded.
:: Scanning for ieee1394...ieee1394 loaded.
:: Scanning for ohci1394...ohci1394 loaded.
:: Scanning for sbp2...sbp2 loaded.
```

>> Mounting filesystems...

```
Started device management daemon v1.3.25 for /newroot/dev
>> Attempting to mount CD:- /newroot/dev/cdroms/*
>> Attempting to mount CD:- /newroot/dev/ide/cd/*
```

```
BusyBox v1.00-pre7 (2004.07.18-21:56+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.
```

```
/bin/ash: can't access tty; job control turned off
/ # _
```





Ubuntu



Fedora



Debian



Arch



Gentoo



Linux
From
Scratch

Unix beards

WhoCaresAboutThat 2015



7.c. Default: Manual Configuration

Introduction

Manually configuring a kernel is often seen as the most difficult course every Linux users ever has to go through. Nothing is less true -- after configuring a couple of kernels you don't even remember that it was difficult ;)

However, one thing *is* true: you must know your system when you start configuring a kernel manually. Most information can be gathered by viewing the contents of `/proc/pci` (or by using `lspci` if available). You can also run `lsmod` to see what kernel modules the LiveCD uses (it might provide you with a nice hint on what to enable).

Now go to your kernel source directory and execute `make menuconfig`. This will fire up an ncurses-based configuration menu.

Code Listing 5: Invoking menuconfig

```
# cd /usr/src/linux
# make menuconfig
```

You will be greeted with several configuration sections. We'll first list some options you must activate (otherwise Gentoo will not function, or not function properly without additional tweaks).

Activating Required Options

First of all, activate the use of development and experimental code/drivers. You need this, otherwise some very important code/drivers won't show up:

Code Listing 6: Selecting experimental code/drivers

```
Code maturity level options --->
[*] Prompt for development and/or incomplete code/drivers
```

Make sure that you compile your kernel with the correct processor family:

Code Listing 7: Selecting correct processor family



Introduzione



Obiettivi

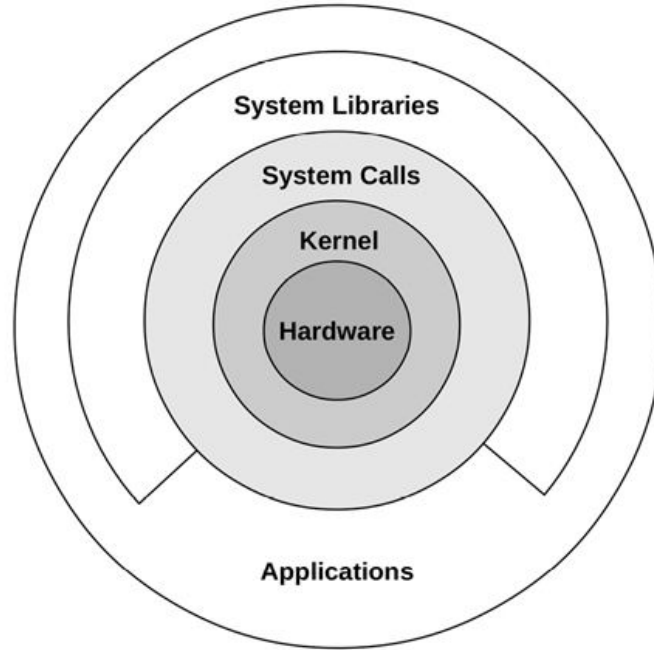
1. Capire dati e trend interessanti del kernel
2. Capire come scaricare e compilare il kernel
3. Capire come creare, verificare ed inviare una patch
4. Capire come scrivere un piccolo modulo in Rust



Il kernel Linux



Le responsabilità del kernel





"The Linux kernel is the largest collaborative software project ever."

Statistics:

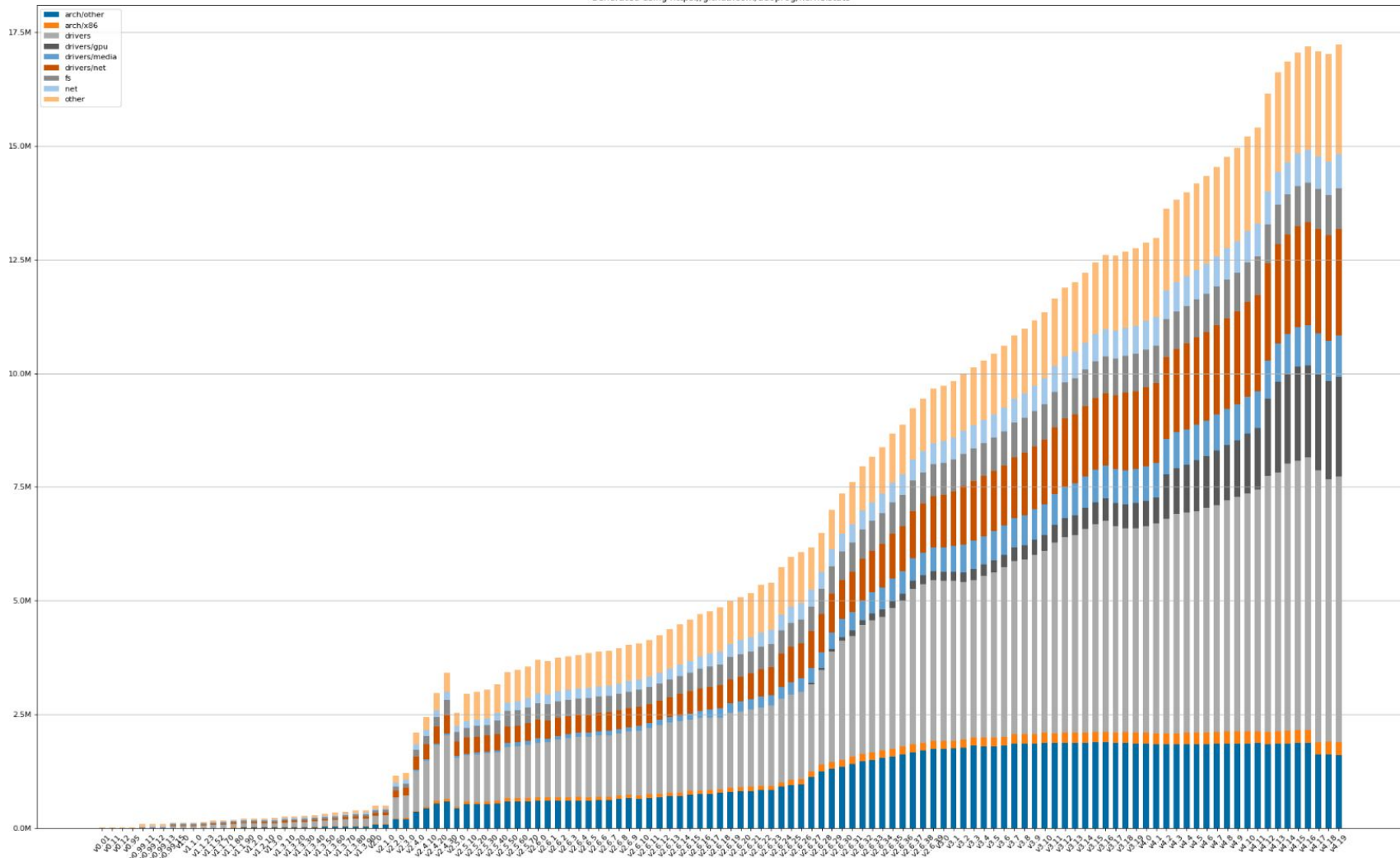
It's 88 files with about ten thousand lines, written by yours truly except for the vsprintf routine which was co-written with Lars Wirzenius.

(...)

Hardware needed for running linux:

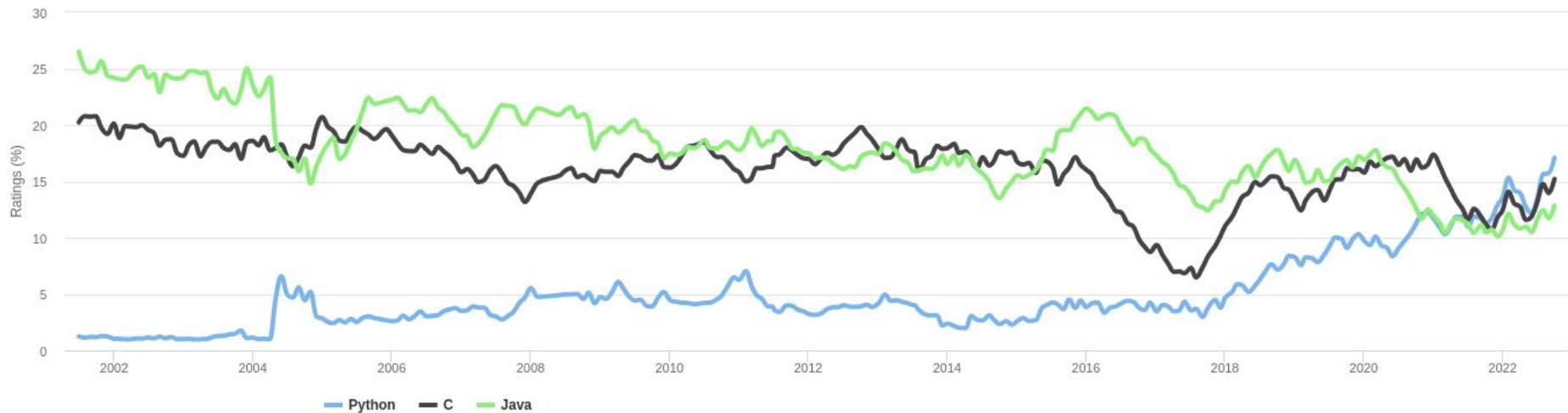
- 386 AT
- VGA/EGA screen
- AT-type harddisk controller (IDE is fine)
- Finnish keyboard (oh, you can use a US keyboard, but not without some practise :-)





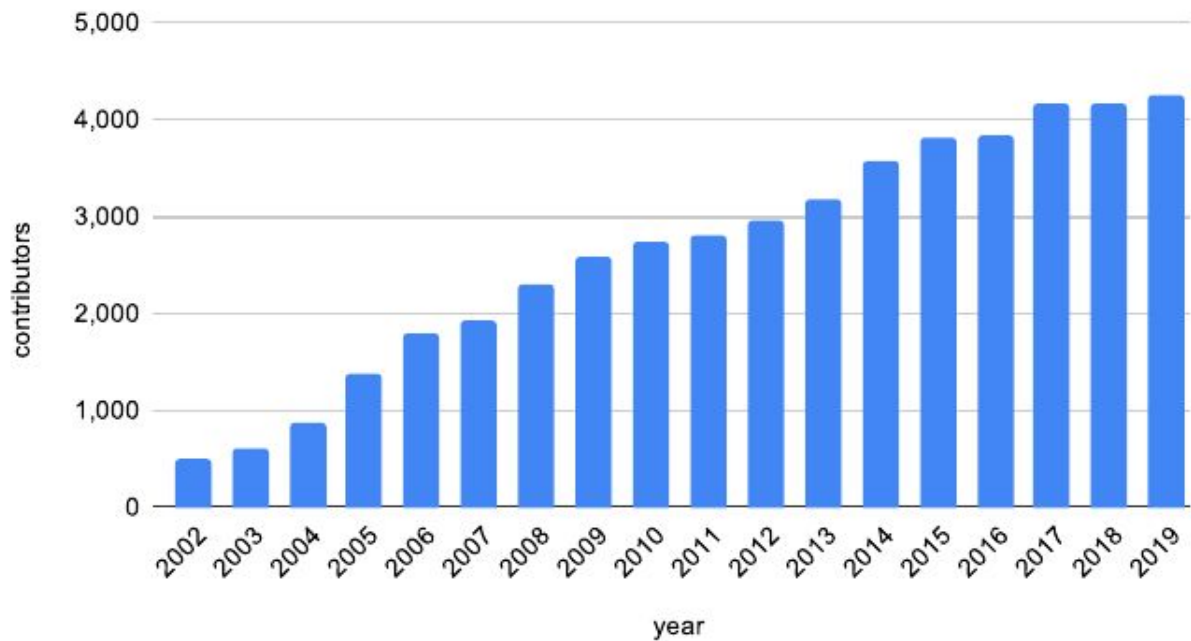
TIOBE Programming Community Index

Source: www.tiobe.com



Rating basato sul tipo e numero di ricerche sul web

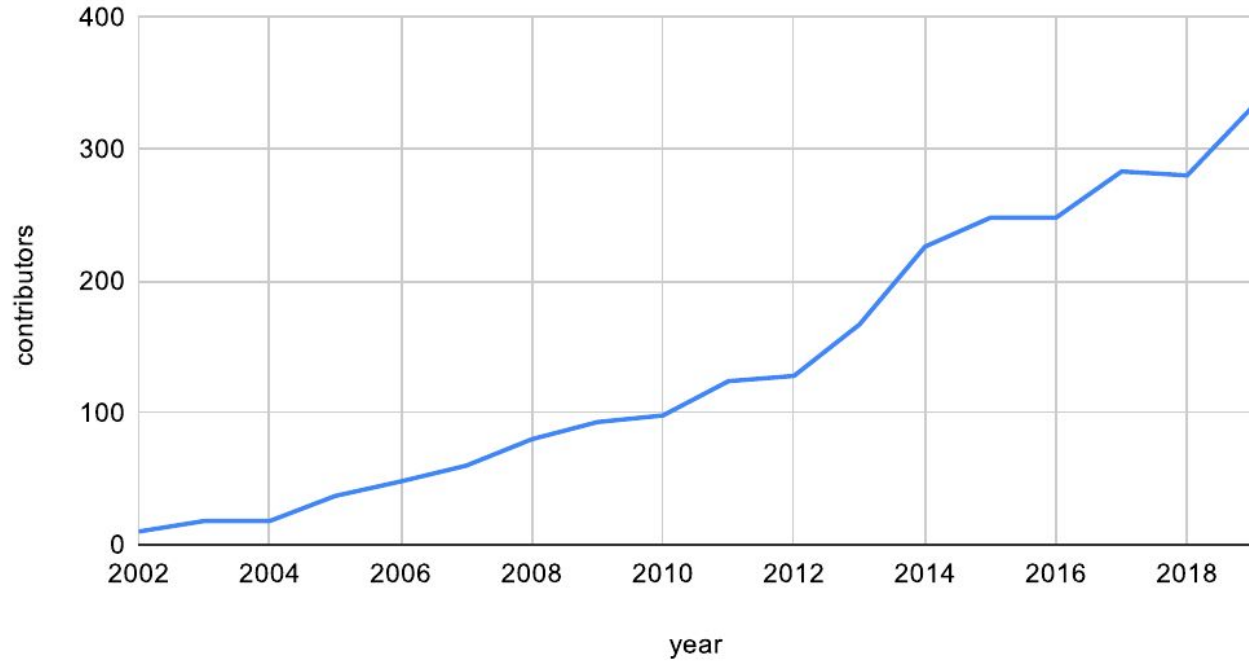




Contributors to the Linux Kernel per Year

Fonte: Linux Foundation (2020)





Estimated Female Contributors to the Linux Kernel per Year

Fonte: Linux Foundation (2020)

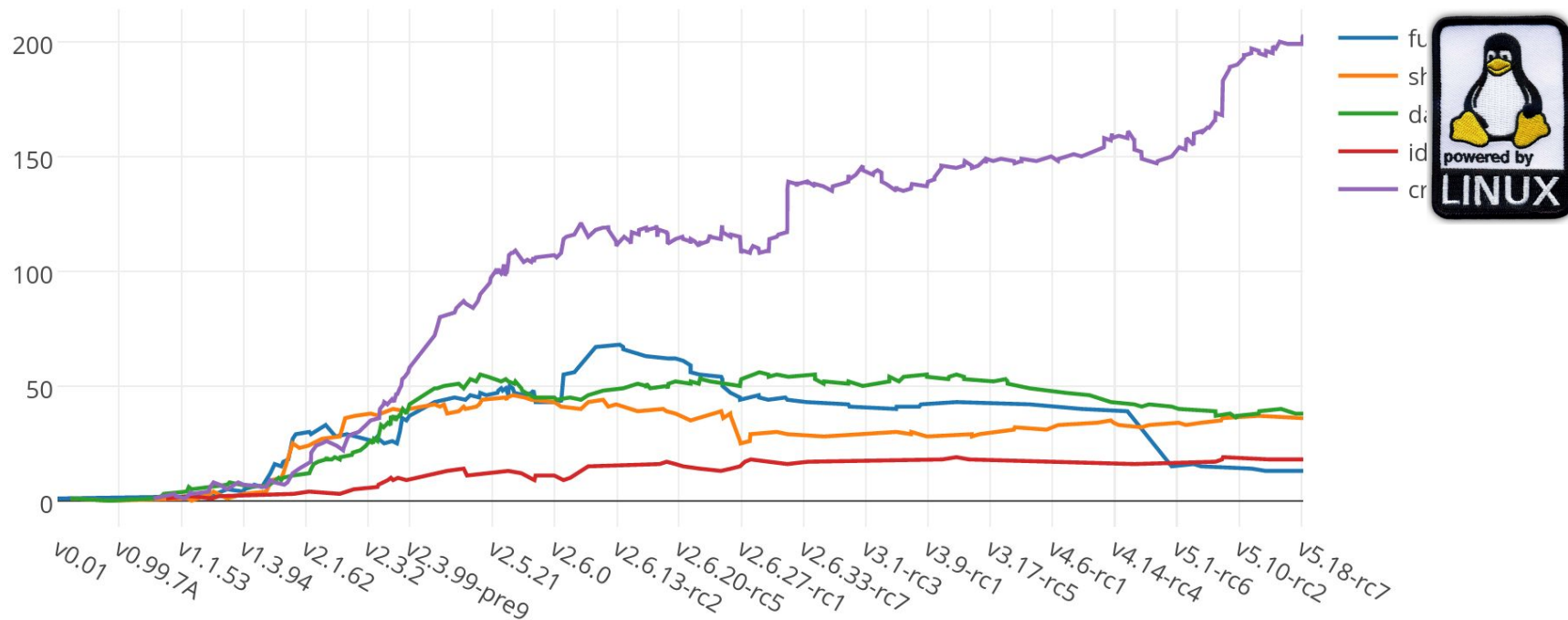


Organization	# of Commits 2007-2019	%
None	93,225	11.95%
Intel	78,068	10.01%
Red Hat	69,443	8.90%
(Unknown)	31,919	4.09%
IBM	29,538	3.79%
SUSE	27,239	3.49%
Linaro	24,740	3.17%
(Consultant)	23,081	2.96%
Google	21,779	2.79%
Samsung	20,160	2.58%

Fonte: Linux Foundation (2020)



Occurrences of "\$^%' words in the Linux kernel source code over time



Fonte: <https://www.vidarholen.net/contents/wordcount/>



Come compilare il kernel





Greg Kroah-Hartman
Linux Kernel Developer and Maintainer



Pre-requisiti (minimi)

GNU C

Clang/LLVM (optional)

Rust (optional)

bindgen (optional)

GNU make

bash

binutils

flex

bison

pahole

util-linux

kmod

e2fsprogs

jfsutils

reiserfsprogs

xfsprogs

squashfs-tools

btrfs-progs

pcmciautils

quota-tools

PPP

nfs-utils

procps

udev

grub

mcelog

iptables

openssl & libcrypto

bc

cpio

Fonte: Documentation/process/changes.rst (4.18)



Pre-requisiti (per la demo)

- Git
- Make
- gcc

Su Fedora:

```
$ sudo dnf install -y \  
    git gcc make bc elfutils-libelf-devel \  
    openssl-devel flex bison ncurses-devel
```

Oppure, semplicemente:

```
$ sudo dnf group install -y "Development Tools"
```



Dove scaricare i sorgenti del kernel



index : kernel/git/torvalds/linux.git

Linux kernel source tree

[about](#) [summary](#) [refs](#) [log](#) [tree](#) [commit](#) [diff](#) [stats](#)

Branch	Commit message
master	Merge tag 'mm-hotfixes-stable-2022-10-11' of git://git.kernel.org/pub/scm/lin...

Tag	Download
v6.0	linux-6.0.tar.gz
v6.0-rc7	linux-6.0-rc7.tar.gz
v6.0-rc6	linux-6.0-rc6.tar.gz
v6.0-rc5	linux-6.0-rc5.tar.gz
v6.0-rc4	linux-6.0-rc4.tar.gz
v6.0-rc3	linux-6.0-rc3.tar.gz
v6.0-rc2	linux-6.0-rc2.tar.gz
v6.0-rc1	linux-6.0-rc1.tar.gz
v5.19	linux-5.19.tar.gz
v5.19-rc8	linux-5.19-rc8.tar.gz
[...]	

[git://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git](https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git)



Partire da zero

↓ Pulisci i file di build (e.g. .ko) ma non la configurazione

```
$ make clean
```

↓ Pulisci i file di configurazione (incluso .config, occhio!)

```
$ make mrproper
```

↓ Boom 

```
$ make distclean
```



Creare una configurazione

```
$ ARCH=x86_64 make defconfig
```

oppure

```
$ cp /boot/config-... .config && yes "" | make oldconfig
```

infine

```
$ make menuconfig
```





Compilare il kernel





Intervallo



Come aderire alle linee guida per C



Regole base

1. Indentazione con tab a 8 caratteri
2. Ogni linea ha un massimo di 80 caratteri
3. I blocchi e le parentesi graffe seguono lo stile K&R
4. Usare in modo giudizioso gli spazi ed i commenti
5. I nomi sono espressivi e usano solo lettere minuscole (eccetto le costanti, macro e gli enum)
6. Non_t usare_t typedef per_t struct_t e puntatori_t
7. Funzioni brevi che fanno (bene) una cosa sola
8. Usare goto solo in alcuni casi
9. Ritornare 0 o 1? vedi predicati vs. comandi
10. Usare le macro in include/linux/kernel.h

Fonte:

<https://www.kernel.org/doc/html/latest/process/coding-style.html>



Ed in particolare...

```
if (cond) {
```

```
    do_something();
```

```
} else {
```

```
    do_something_else();
```

```
}
```



```
if (cond)
```

```
    do_something();
```

```
else
```

```
    do_something_else();
```

```
oooops();
```

"Do not unnecessarily use braces where a single statement will do."

<https://www.kernel.org/doc/html/v4.10/process/coding-style.html#placing-braces-and-spaces>



Come trovare qualcosa da fare



Visita Kernel Janitors

<https://kernelnewbies.org/KernelJanitors/ToDo>

- * `printk()` calls should include appropriate `KERN_*` constant (of course only at beginning of lines). Also consider using `dev_printk()` and friends.
- * `pr_debug()` from `kernel.h` could replace a lot of `DPRINTK` and similar macros.
- * same for `pr_info()`
- * Lots of unnecessary casts (mostly `void*` pointers) in drivers can be removed.

Example:

```
- struct netdev_private *np = (struct netdev_private *)dev->priv;  
+ struct netdev_private *np = dev->priv;
```

use `netdev_priv()` for network devices
See [/VoidPointerConvs](#) for more details.

- * `ARRAY_SIZE` macro has duplicates, remove them.

(2007/10/27 - the 4 remaining duplicate defs have good reasons for existing. Someone please verify agreement and then remove this todo item)

- * put `BIT` macro into `kernel.h` (?) and remove it from drivers
- * Purely cosmetic, but far nicer to read.

```
- for (list = ymf_devs.next; list != &ymf_devs; list = list->next) {  
+ list_for_each(list, &ymf_devs) {
```

- * Convert comments to C99 initializers:

```
- /* 10 */ DECLARE_PII_X_DEV("ICH2"),  
+ [10] = DECLARE_PII_X_DEV("ICH2")
```



Guarda i TODO

Specialmente in drivers/staging



Trova piccole cose da sistemare

Prova con checkpatch:

```
$ ./scripts/checkpatch.pl -f drivers/ ... /*.c
```



RAI



**RODI GARGANICO
(Foggia) Vista Dal Porto**

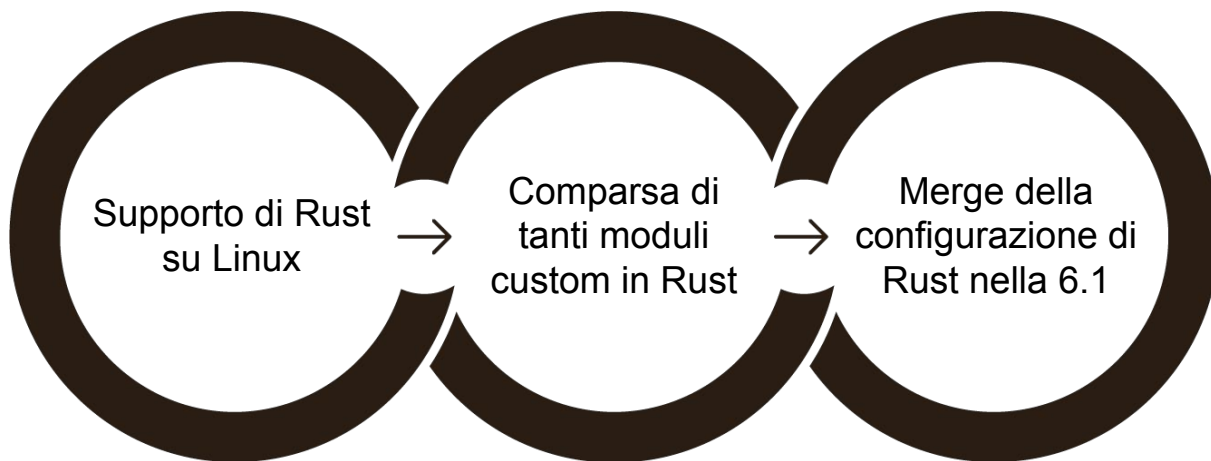
Secondo Intervallo



Come scrivere moduli in Rust



Rust finalmente nel Kernel!



Esempio di modulo in Rust

```
use kernel::prelude::*;

module! {
    type: HelloKernel,
    name: b"hello-kernel",
    author: b"Carmelo Sarta",
    description: b"Rust minimal hello world",
    license: b"GPL",
}

struct HelloKernel {
    words: Vec<String>,
}
```

```
impl kernel::Module for HelloKernel {
    fn init(_module: &'static ThisModule) -> Result<Self> {
        pr_info!("Rust minimal sample (init)\n");
        pr_info!("Am I built-in? {}\n", !cfg!(MODULE));

        let mut words = Vec::new();
        words.try_push("Hello")?;
        words.try_push("there,")?;
        words.try_push("Kernel!")?;

        Ok(HelloKernel { words.try_to_owned()? })
    }
}

impl Drop for HelloKernel {
    fn drop(&mut self) {
        pr_info!("My phrase is {:?}\n", self.words.join("-"));
        pr_info!("hello-kernel (exit)\n");
    }
}
```



Fundamental Steps

- Clonare la repo del kernel Linux
- Aggiungere il vostro fantastico modulo scritto in Rust
- Aggiungere un'appropriata configurazione al Kconfig file e al Makefile del kernel
- Abilitare il modulo nel kernel dal menù di configurazione
- Finalmente buildare il kernel con il vostro modulo dentro!

```
.config - Linux/x86 6.0.0 Kernel Configuration

Linux/x86 6.0.0 Kernel Configuration
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty
submenus ----). Highlighted letters are hotkeys. Pressing <Y>
includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to
exit, <?> for Help, </> for Search. Legend: [*] built-in [ ]

^(-)
-- Enable the block layer --->
  Executable file formats --->
  Memory Management options --->
  [*] Networking support --->
    Device Drivers --->
    File systems --->
    Security options --->
  -- Cryptographic API --->
    Library routines --->
  | Kernel hacking --->

<Select> <Exit> <Help> <Save> <Load>
```



Come creare una patch



1. Crea un branch

```
$ git checkout -b test-linux-day
```



2. Prova a ricompilare il modulo

Abilitare il modulo in `make menuconfig`

Ricompila il kernel (per le nuove dipendenze)

```
$ make prepare && make
```

In seguito...

```
$ make modules
```

Oppure anche...

```
$ make M=drivers/staging/...
```

> "prepare: things we need to do before we recursively start building the kernel"



3. Crea il commit con un messaggio idoneo

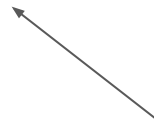
```
$ git commit -m "Staging: <nome_modulo>: <cosa fa>"
```

anzi...

```
$ git commit --amend
```

aggiungere:

```
Signed-off-by: Autore <autore@example.com>
```



Developer's Certificate of Origin



4. Creare la patch

```
$ git format-patch master..test-linux-day
```



Every program attempts to expand until it can read mail.

Those programs which cannot so expand are replaced by ones which can.



5. Installare e configurare il client email di Git

```
$ dnf install -y git-email
```

```
$ git config --global sendemail.smtppass 11n00xd4y
```

```
$ git config --global sendemail.smtpuser zedr@zedr.com
```

```
$ git config --global sendemail.smtpencryption tls
```

```
$ git config --global sendemail.smtpserver mail.example.com
```

```
$ git config --global sendemail.serverport 465
```



6. Identificare i maintainer

```
$ ./scripts/get_maintainer.pl 0001-mia.patch
```

```
Greg Kroah-Hartman <gregkh@linuxfoundation.org>  
(supporter:STAGING SUBSYSTEM, commit_signer:4/5=80%)
```

```
linux-staging@lists.linux.dev (open list:STAGING SUBSYSTEM)
```

```
linux-kernel@vger.kernel.org (open list)
```



7. Inviare la patch

```
$ git send-email \  
    --smtp-debug \  
    --to gregkh@linuxfoundation.org \  
    --cc linux-staging@lists.linux.dev \  
    --cc linux-kernel@vger.kernel.org \  
    0001-mia.patch
```



Il risultato...



From: **Greg KH** <gregkh@linuxfoundation.org>
Date: Thu, 13 Oct 2022, 07:56
Subject: Re: [PATCH] Staging: rtl8192e: fix a brace style issue
To: Rigel Di Scala <zedr@zedr.com>

On Wed, Oct 12, 2022 at 09:16:12PM +0000, Rigel Di Scala wrote:
> Fixed a coding style issue affecting a conditional if statement.
>
> Signed-off-by: Rigel Di Scala <zedr@zedr.com>
> ---
> drivers/staging/rtl8192e/rtl819x_BAProc.c | 5 ++---
> 1 file changed, 2 insertions(+), 3 deletions(-)
(...)

Hi,

This is the friendly semi-automated patch-bot of Greg Kroah-Hartman. You have sent him a patch that has triggered this response.

Right now, the development tree you have sent a patch for is "closed" due to the timing of the merge window. Don't worry, the patch(es) you have sent are not lost, and will be looked at after the merge window is over (after the -rc1 kernel is released by Linus).

So thank you for your patience and your patches will be reviewed at this later time, you do not have to do anything further, this is just a short note to let you know the patch status and so you don't worry they didn't make it through.

thanks,

greg k-h's patch email bot

Prima email automatica da Greg KH



From: **Philipp Hortmann** <philipp.g.hortmann@gmail.com>
Date: Fri, 14 Oct 2022, 18:56
Subject: Re: [PATCH] Staging: rtl8192e: fix a brace style issue
To: Rigel Di Scala <zedr@zedr.com>, <gregkh@linuxfoundation.org>
Cc: <linux-staging@lists.linux.dev

On 10/12/22 23:16, Rigel Di Scala wrote:

> Fixed a coding style issue affecting a conditional if statement.
>
> Signed-off-by: Rigel Di Scala <zedr@zedr.com>
> ---
> drivers/staging/rtl8192e/
> rtl819x_BAProc.c | 5 ++---
> 1 file changed, 2 insertions(+), 3 deletions(-)
(...)

Tested-by: Philipp Hortmann <philipp.g.hortmann@gmail.com>

Seconda email automatica da PH



From: **Greg KH** <gregkh@linuxfoundation.org>
Date: Thu, 20 Oct 2022, 17:35
Subject: patch "Staging: rtl8192e: fix a brace style issue" added to staging-testing
To: <zedr@zedr.com>, <gregkh@linuxfoundation.org>, <philipp.g.hortmann@gmail.com>

This is a note to let you know that I've just added the patch titled

Staging: rtl8192e: fix a brace style issue

to my staging git tree which can be found at
[git://git.kernel.org/pub/scm/linux/kernel/git/gregkh/staging.git](https://git.kernel.org/pub/scm/linux/kernel/git/gregkh/staging.git)
in the staging-testing branch.

The patch will show up in the next release of the [linux-next](#) tree
(usually sometime within the next 24 hours during the week.)

The patch will be merged to the staging-next branch sometime soon,
after it passes testing, and the merge window is open.

If you have any questions about this process, please let me know.

From 06b764cf60877bdd6604c173963501e16236ccdf Mon Sep 17 00:00:00 2001
From: Rigel Di Scala <zedr@zedr.com>
Date: Wed, 12 Oct 2022 21:16:12 +0000
Subject: Staging: rtl8192e: fix a brace style issue

Fixed a coding style issue affecting a conditional if statement.

Signed-off-by: Rigel Di Scala <zedr@zedr.com>
Tested-by: Philipp Hortmann <philipp.g.hortmann@gmail.com>
Link: <https://lore.kernel.org/r/20221012211612.75871-1-zedr@zedr.com>
Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>



Infine, altra email da Greg KH - patch accettata!

