

Fieldbus API Working Group

GPL, copyleft and „derivative work“ in the context of UIO (user space) drivers

Carsten Emde



Open Source Automation Development Lab (OSADL) eG



What is Open Source software?

„Open Source“ is a software license.

The rules of Open Source:

Rights: Software may – without restriction - be

- used
- **analysed***
- distributed
- **modified***

Obligation:

- All rights must be preserved when the software is passed on



What is Open Source software?

„Open Source“ is a software license.

The rules of Open Source:

Rights: Software may – without restriction - be

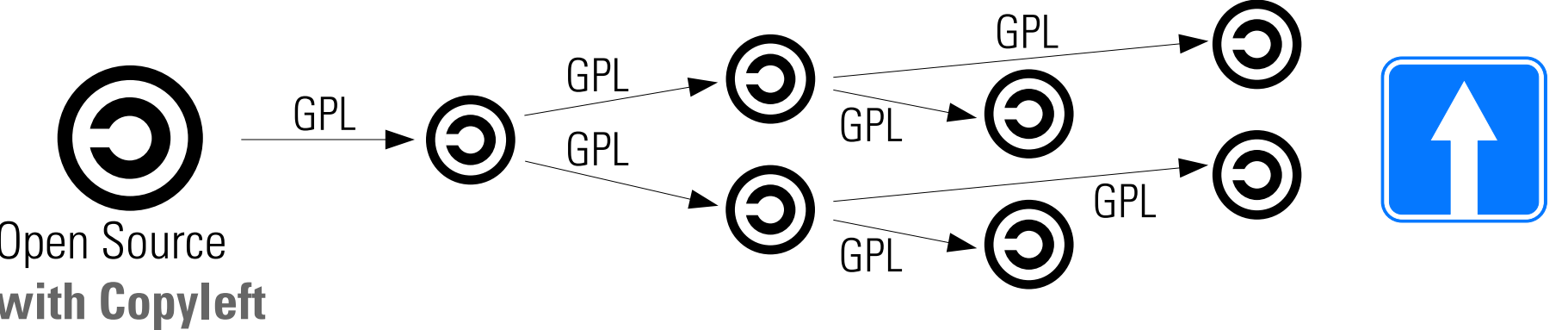
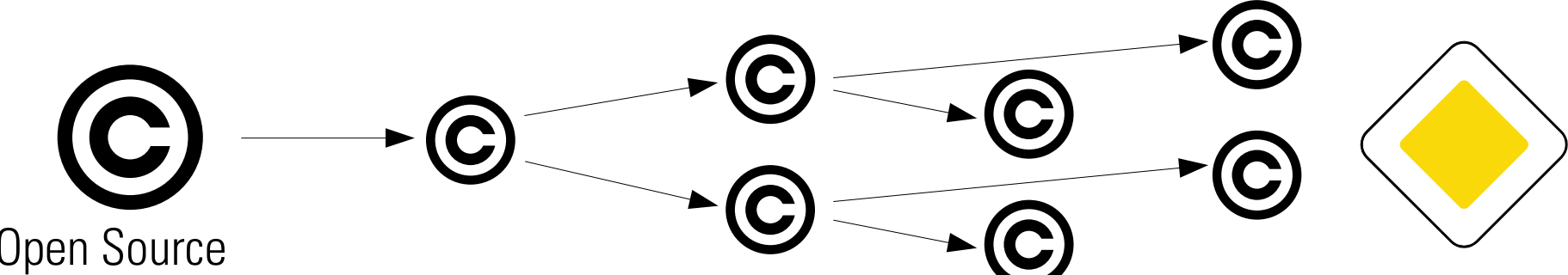
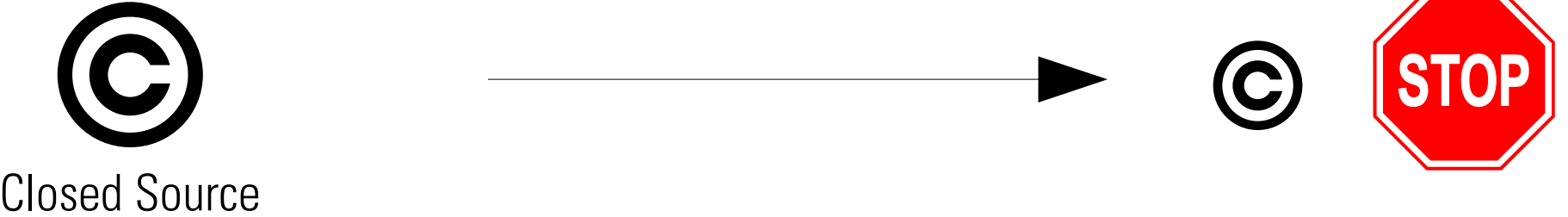
- used
- **analysed***
- distributed
- **modified*** ***requires access to the source code**

Obligation:

- All rights must be preserved when the software is passed on



What is "Copyleft"?



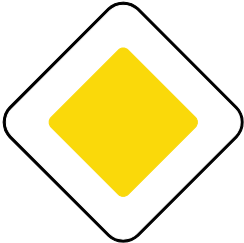
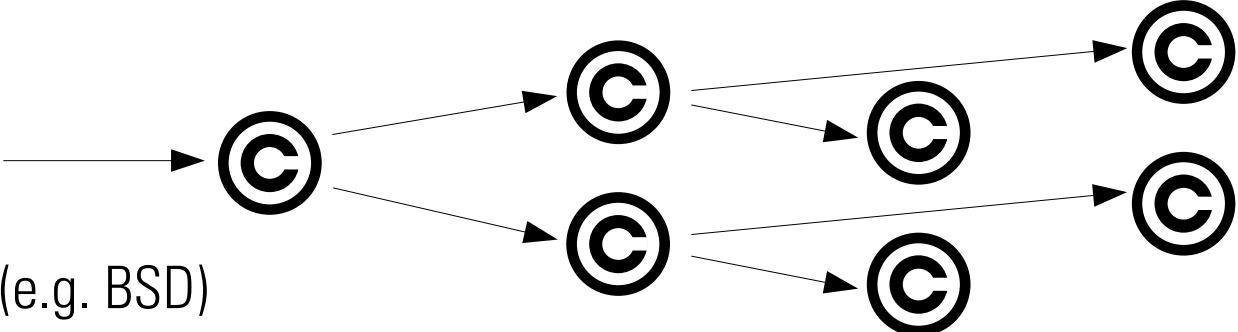
What is "Copyleft"?



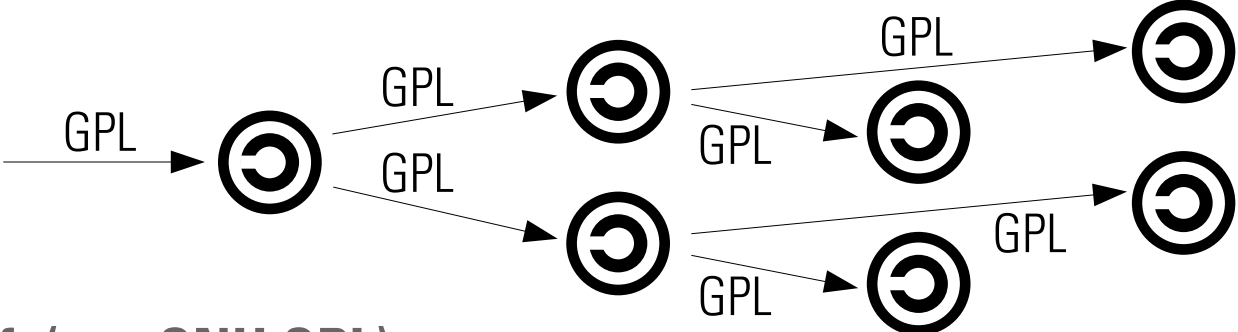
Closed Source



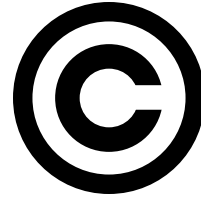
Open Source (e.g. BSD)



Open Source with Copyleft (e.g. GNU GPL)



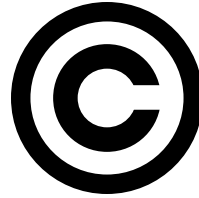
What is “derivative work”?



- **The primary author of a work may reserve the rights to make copy of his work.**
- **This applies not only to the original work but also to “derivative” work – such as, for example, a musical that is written based on the libretto of an opera.**



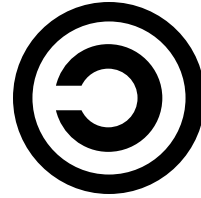
What are the rules of a “derivative work”?



- **The closer a work follows another work, the more likely it is “derivative”.**
- **The more generally known and available terms and facts are used *in parallel* to the original work, the less likely it is “derivative”.**
- **The final decision whether an individual work is “derivative work” or not is made in court.**



How is software concerned?



- **The Linux kernel is licensed under the GPL v2 only. Thus, any “derivative” work must also be licensed under the GPL v2 only.**
- **A normal Linux driver that runs in the Linux kernel space is considered “derivative” work.**
- **A normal Linux application that runs in user space and makes kernel calls according to the POSIX standard is not considered “derivative” work.**



What is a user space (UIO) driver ?

- **Written by Thomas Gleixner, Benedikt Spranger, Hans-Jürgen Koch (Linutronix) and others**
- **Rationale:**

“For many types of devices, creating a Linux kernel driver is overkill. All that is needed is some way to handle an interrupt and provide access to the memory space of the device. The logic of controlling the device does not necessarily have to be within the kernel, as the device does not need to take advantage of **any other resources that the kernel provides.**”

To address this situation, the user space I/O system (UIO) was designed. For typical industrial I/O cards, only a very small kernel module is needed. The main part of the driver will run in user space. **This simplifies the development** and reduces the risk of serious bugs within a kernel module.”



What are the characteristics of a user space (UIO) driver ?

- **A new class of devices, intended for industrial I/O cards**
- **Not intended for any existing class of devices**
- **Standardized kernel API can be used for a wide variety of industrial I/O cards – not just one controller**
- **Can, in principle, be used similarly in other operating systems**
- **Is part of the mainline Linux kernel (since 2.6.22)**
- **Was not primarily designed to circumvent GPL restrictions**



Conclusion

- **The kernel part of a UIO driver must be Open Source and licensed under the GPL v2 only.**
- **The user space part of a UIO driver should preferably also be Open Source and licensed under the GPL v2 only. If a manufacturer, by some reason, decides to release the user space part of a UIO driver as a proprietary binary module, this is very probably not a violation of the GPL.**
- **It is upon the customers to accept or not to accept proprietary binary user space modules.**



The End

