

A quick insight into FLOSS licences compatibility issues

EU Informatic Directorate General - EPractice's workshop :
« Open Source: Its place in a cross-border environment »
(7/04/2011 - Brussels, EU Commission, 2011)

Philippe LAURENT

Senior Researcher at the CRID

(Centre de Recherches Informatique et Droit / University of Namur)

Lawyer at the Brussels Bar

(Marx, Van Ranst, Vermeersch & Partners)

MVVP





Compatibility (broad sense)



- **In a broad sense “Compatibility” of FLOSS licences could be described as :**
 - “the characteristic of two (or more) licences according to which the codes distributed under these licences may be put together in order to create a bigger distributable software”
 - + : * this definitions takes into account many of the combination possibilities
 - ⇒ “BSD and GPL are compatible”
 - ⇒ “BSD and Apache are compatible”
 - ⇒ “*LGPL and Mozilla may, in some cases, be compatible*”
 - : * this definition does not take into account the **result of the combination**
 - * It creates a **false idea of reciprocity**, which could lead to legal mistakes



Compatibility (narrow sense)



- In a narrow sense, “Compatibility” of a FLOSS license is commonly understood as

“the characteristic of a licence according to which the code distributed under this licence may be integrated in a bigger software *that will be distributed under another licence*”

Ex.: cfr. the use of the terms “GPL-Compatible” on the FSF website

Adopted definition

⇒ **COMPATIBILITY of FLOSS Licences = usually a ONE WAY ROAD**

BSD is “**GPL-compatible**” **BUT** **GPL** is **NOT** “**BSD-compatible**”

BSD code **can** be added in a software / GPL code **cannot** be added in a software distributed under GPL / distributed under BSD

⇒ **HOWEVER:**

BSD is MIT compatible AND MIT is BSD compatible

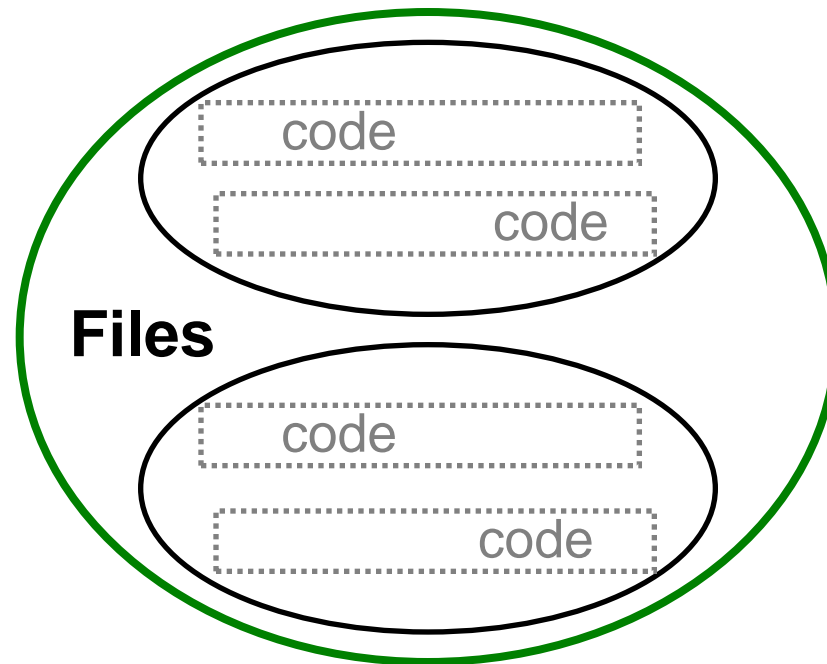


...schematic view...
(= Simplified view!)



Entire software application (“*operational*” code)

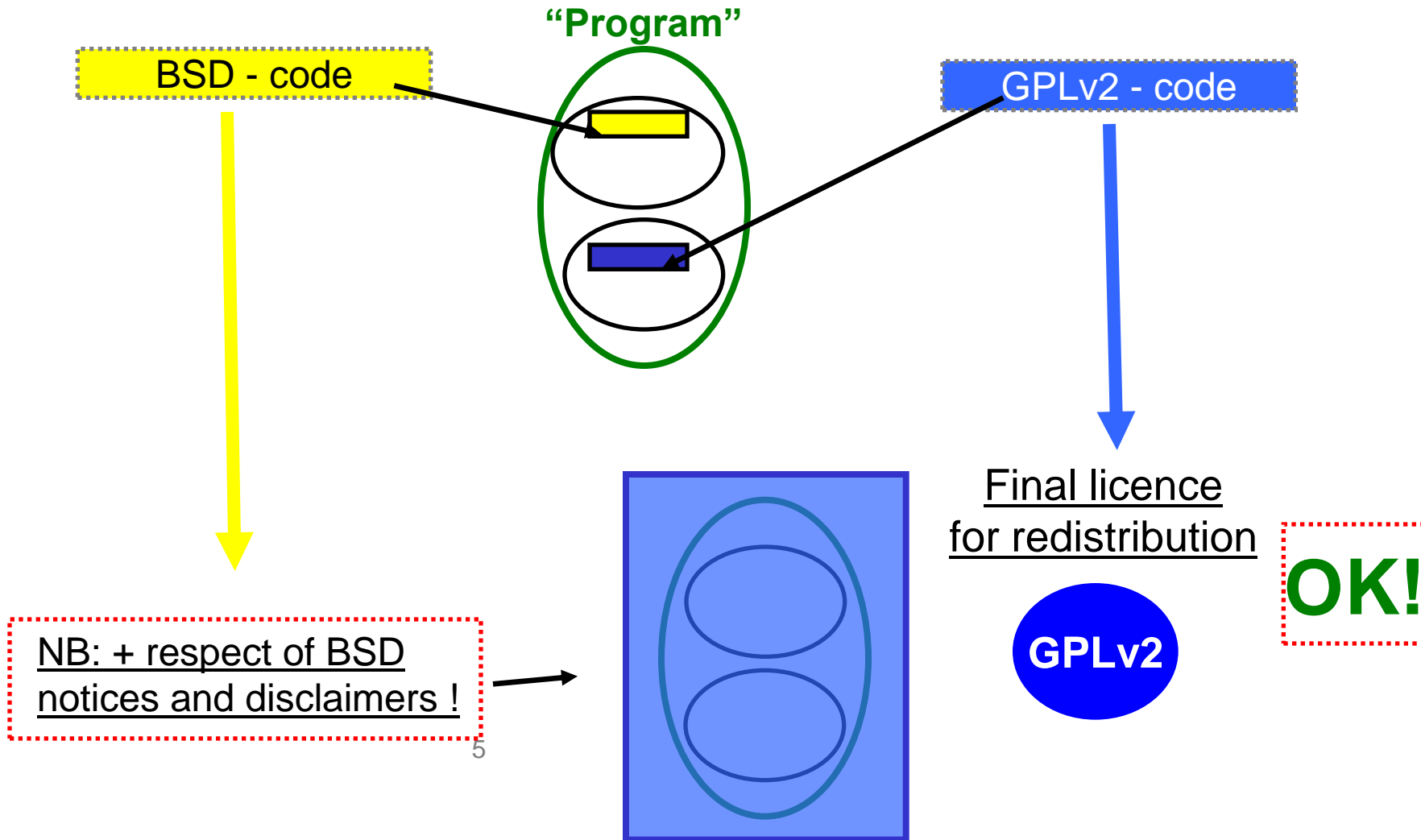
“**Program**” (1 Derivative Work)



NB : the way the codes are used together have an important influence on the result. In our representations, the codes are “merged” in two different files.



EX. : BSD is GPLv2 compatible





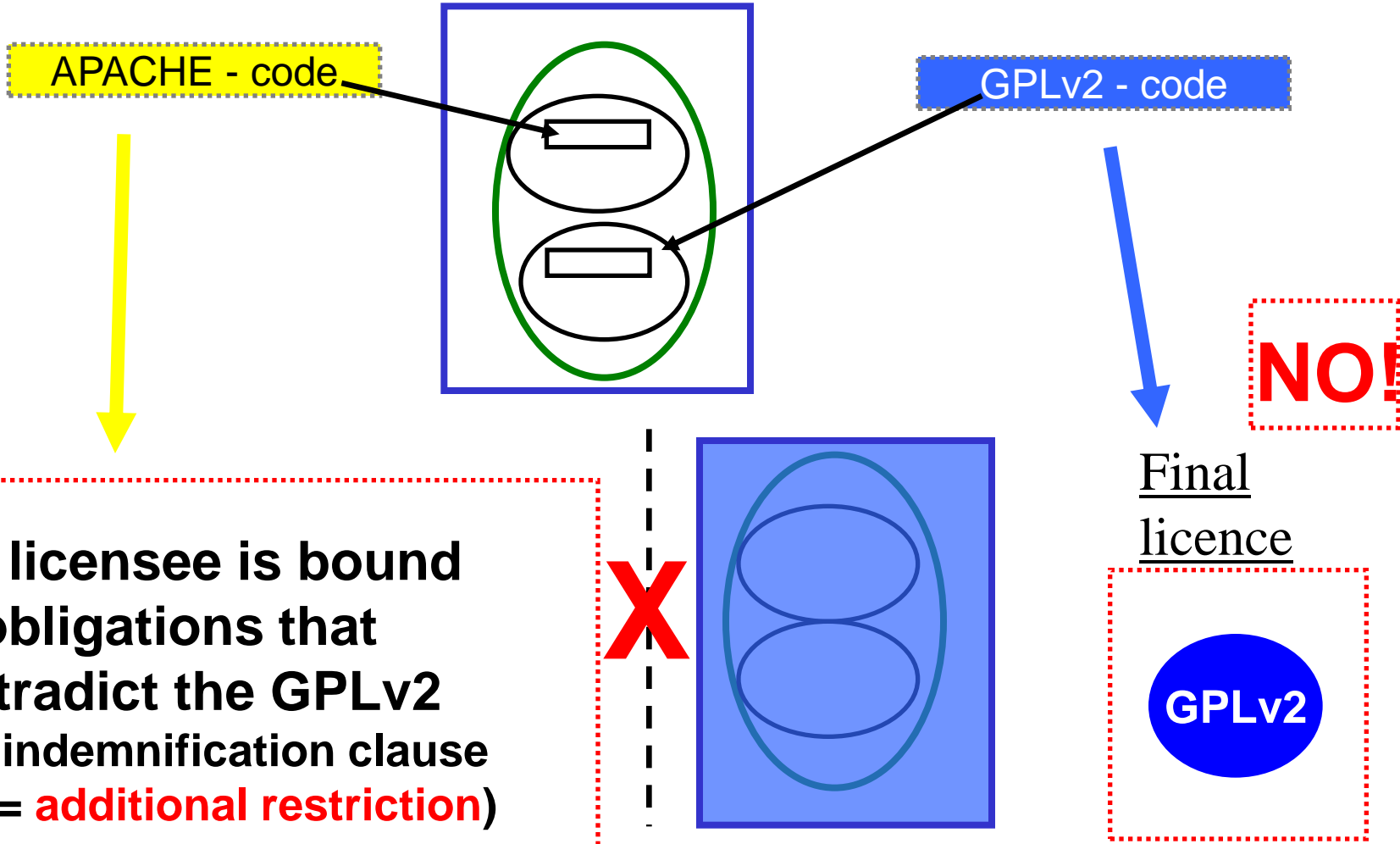
Incompatibility



- **Incompatibility** is due to contradictory obligations provided in the different licences under which two codes to be merged are distributed.
- It can be due to clauses of a multitude of kinds.
 - ⇒ Basically : anything that puts the licensee in a position where he cannot fulfill at the same time any and all his/her obligations under both licences.



Ex. : the Apache licence is not GPLv2 compatible

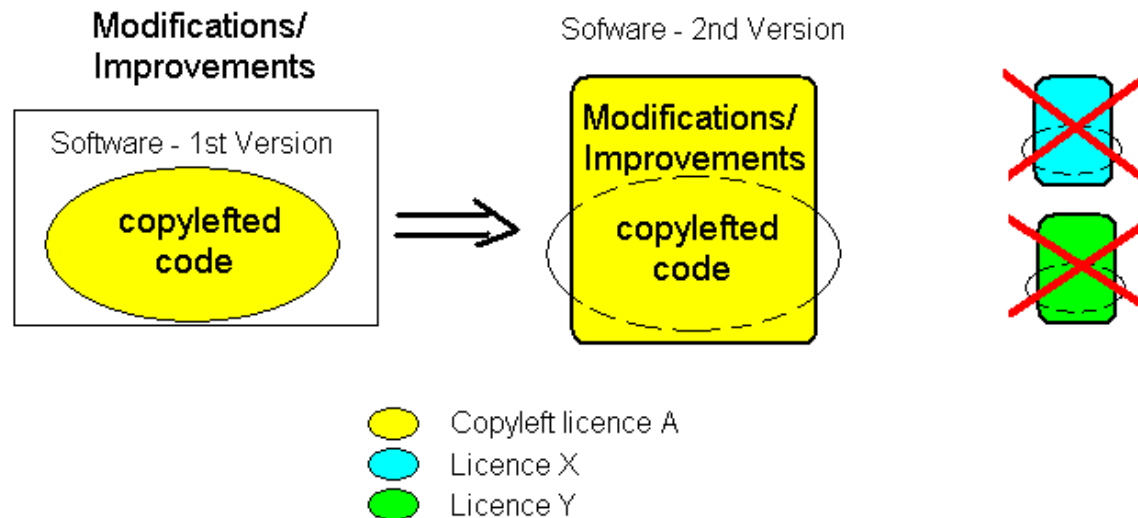




Copyleft is the main source of compatibility problems



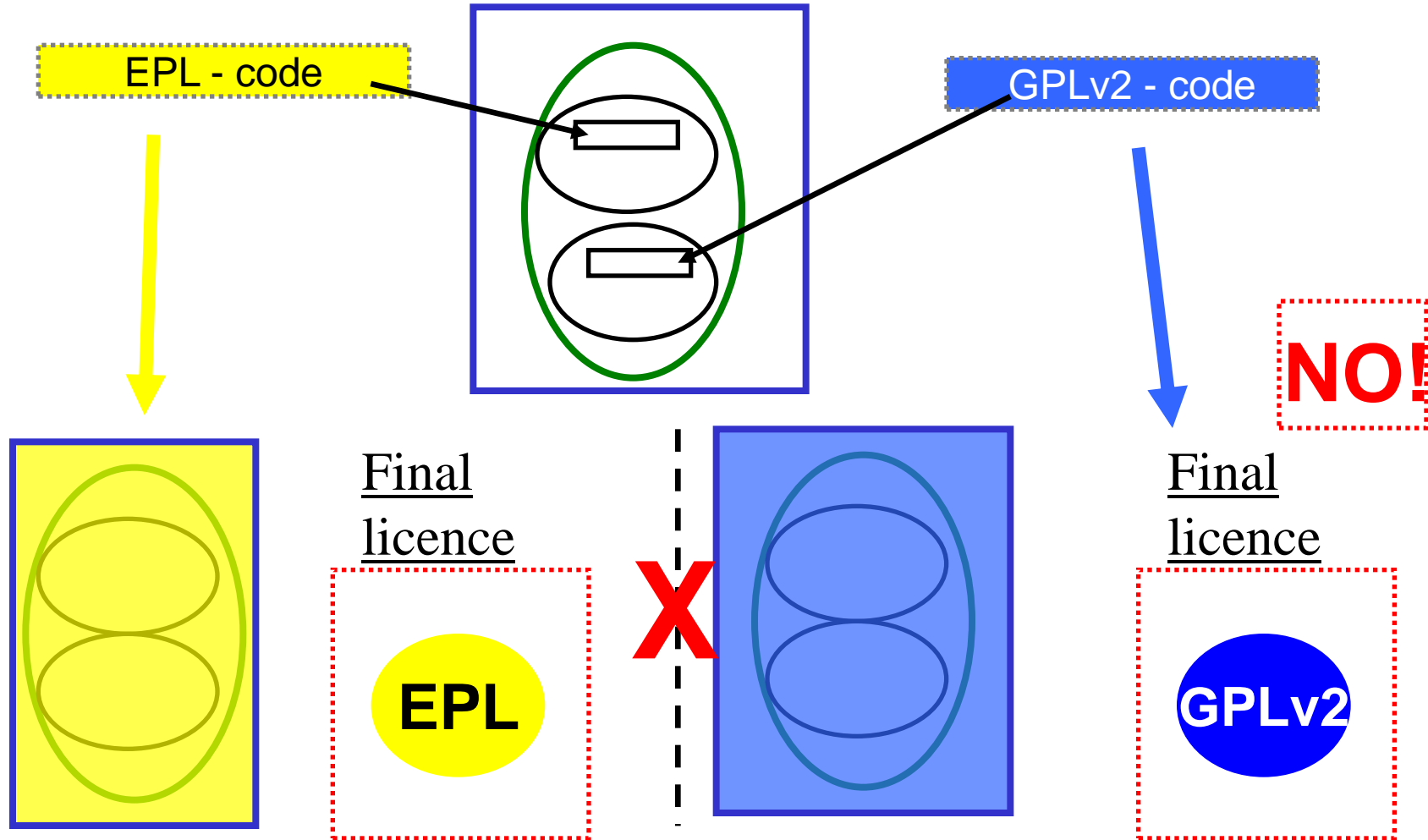
- This copyleft effect is reached by introducing a **copyleft clause** in the FLOSS licence, which, in general, reads more or less as follows*:
“ You are free to modify or merge the software with another one, but if you redistribute the modified or merged version of the software, this redistribution must be done under the same licence”



- There are many types of copyleft effects, with different extents
- Merging some code with copyleft licensed code usually means that the copyleft licence is predominant => **copyleft excludes other licences**

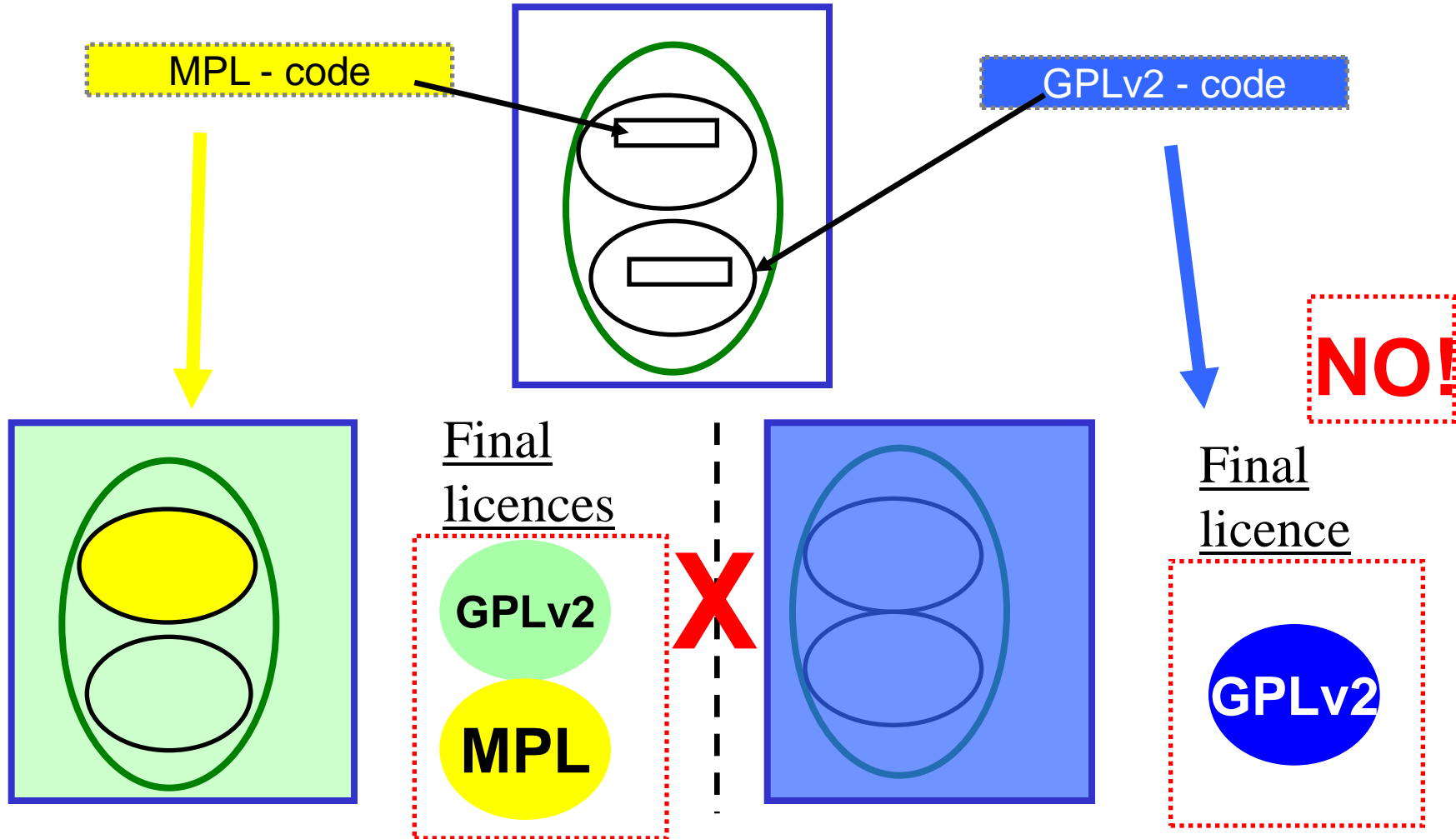


Ex.: the EPL is not GPL compatible



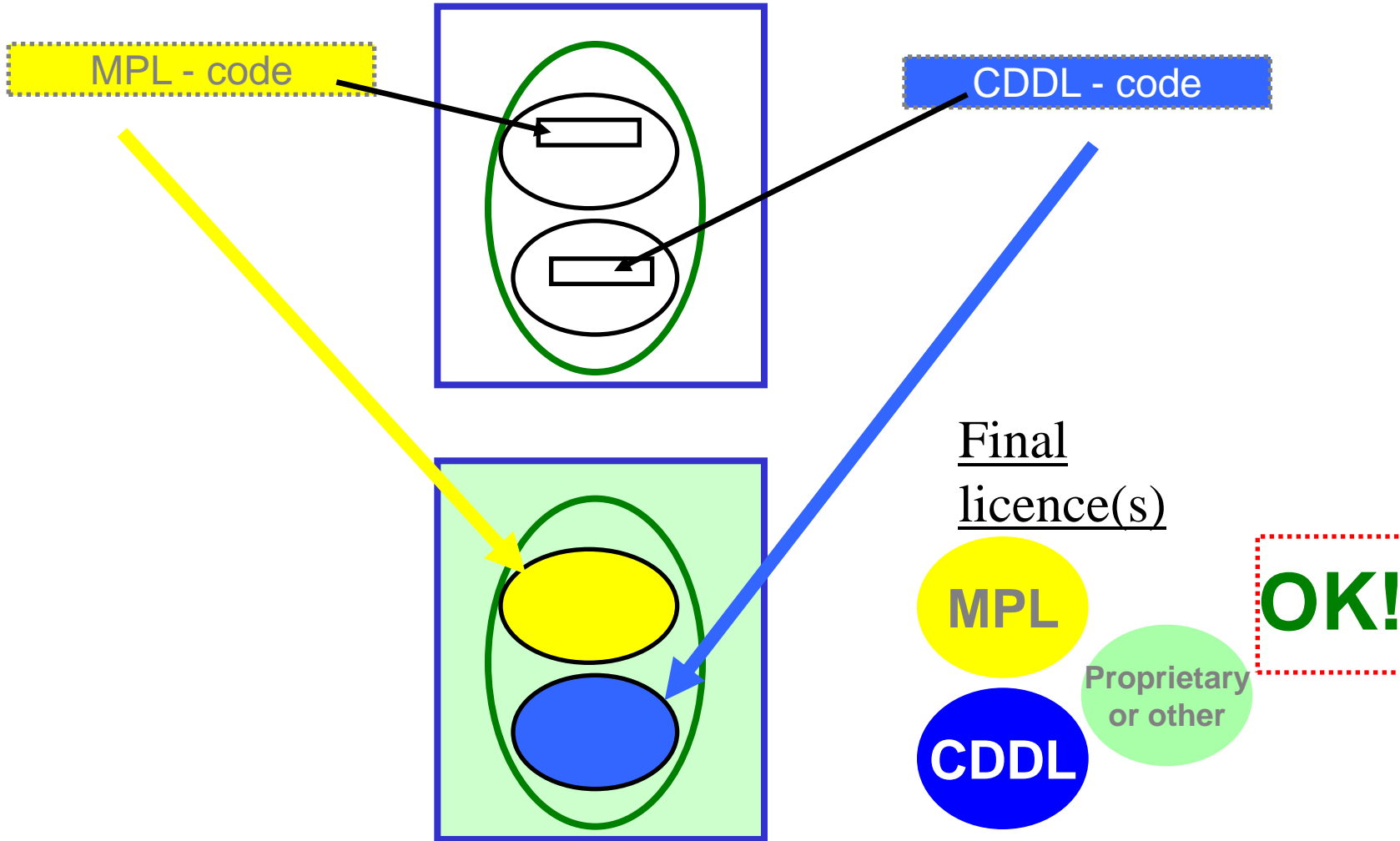


EX. : the MPL is not GPL Compatible

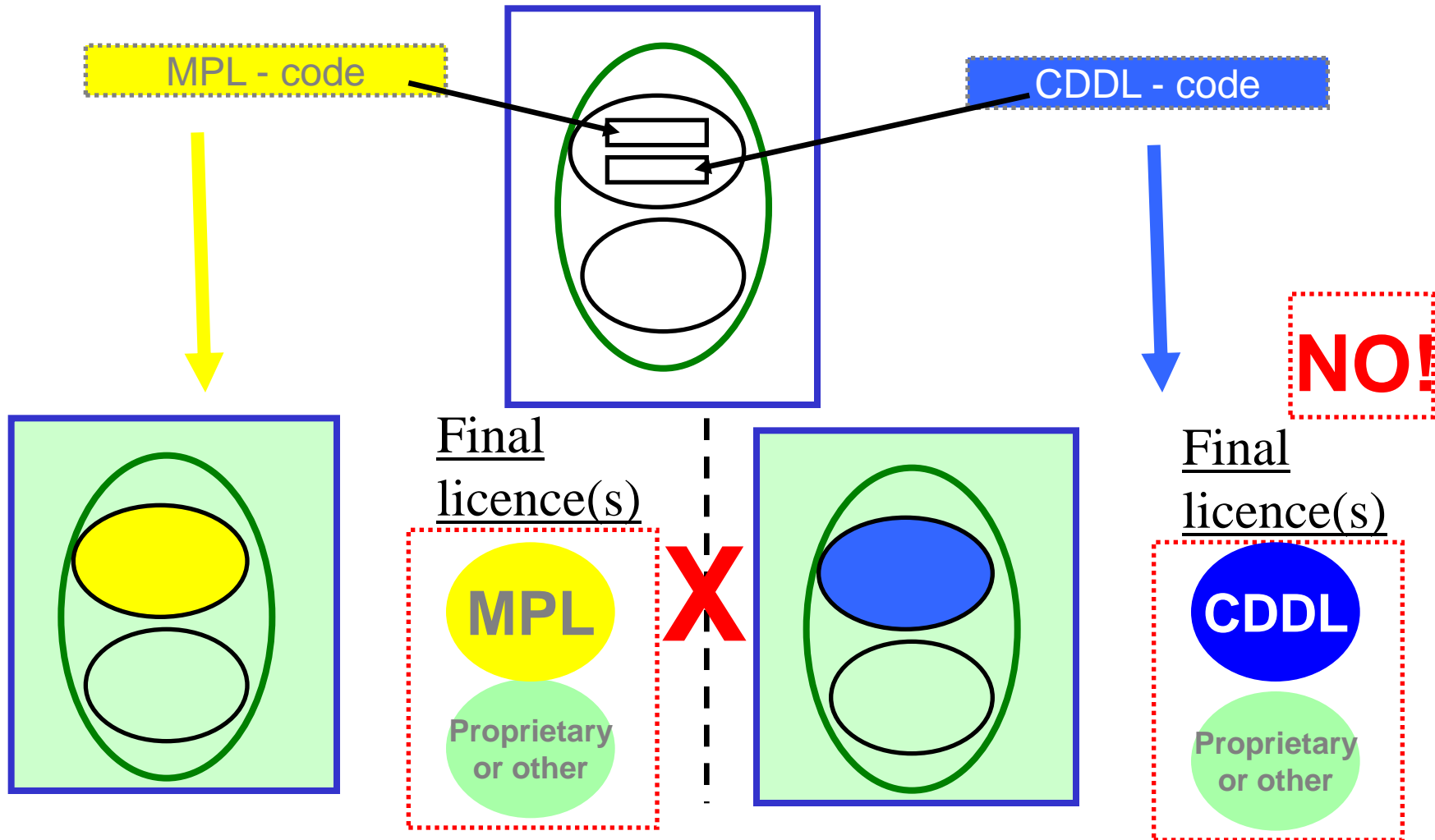




EX. : Mozilla and CDDL ????



EX. : Mozilla and CDDL ????





DRAFTING LICENCES TO SOLVE COMPATIBILITY PROBLEMS

**Ex.: - GPLv3
- EUPL**

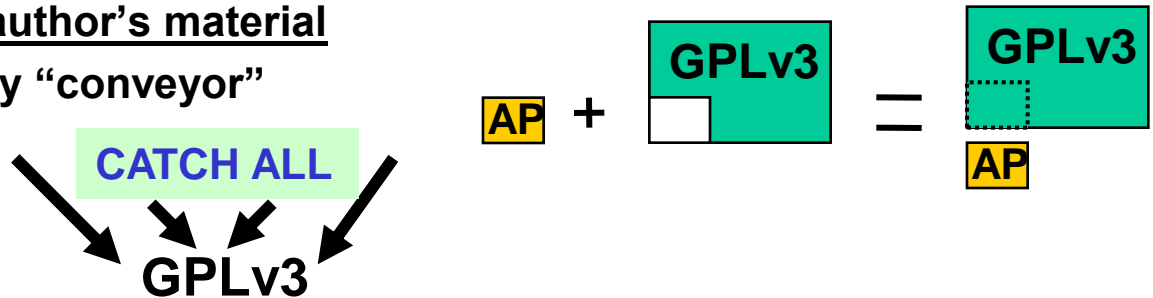


GPL3's Compatibility Related Clauses



Additional permissions (AP): OK

- = additional **exceptions** to one or more of the GPLv3 **conditions**
- may only be added to GPLv3 by the author of additional original code
- will only apply to this author's material
- may be removed by any "conveyor"



Additional non-permissive terms (ANPT): Limited list

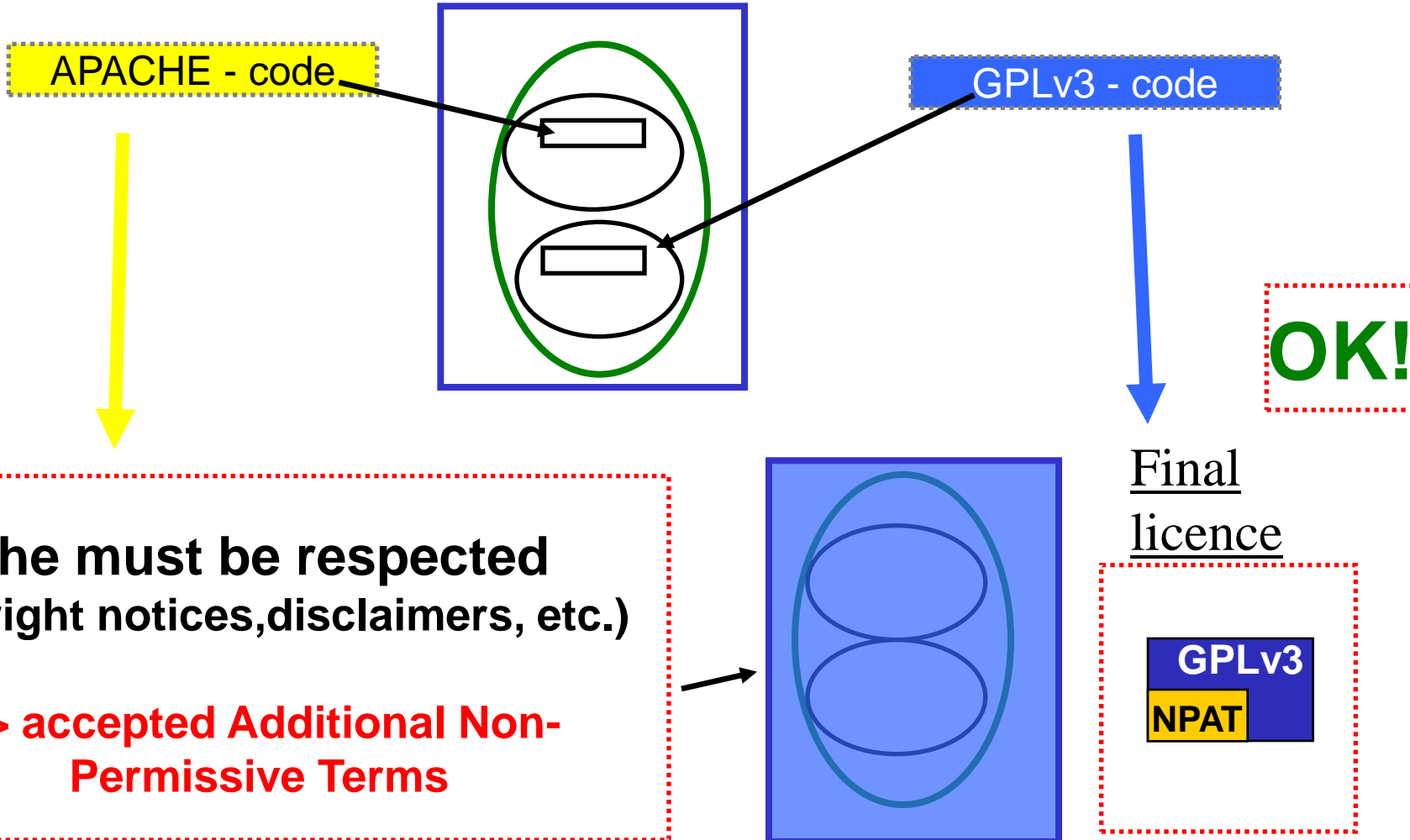
- 6 possible ANPT :
 - 1) different disclaimers (warranty & liability)
 - 2) Legal notices
 - 3) prohibition of misrepresentation of origin
 - 4) Limitation of use of licensors'/authors' names in publicity
 - 5) No trade mark rights grant
 - 6) indemnification clause
- apply only to material added by licensee to a covered work
- not removable



Further restrictions (FR): NO



Ex. : the Apache licence is GPLv3 compatible

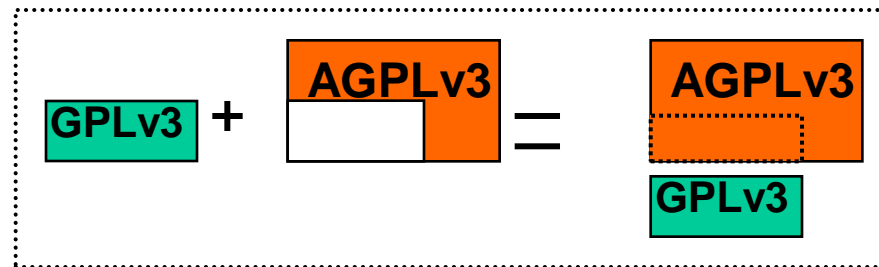




Use of the GPLv3 with the GNU Affero General Public License.

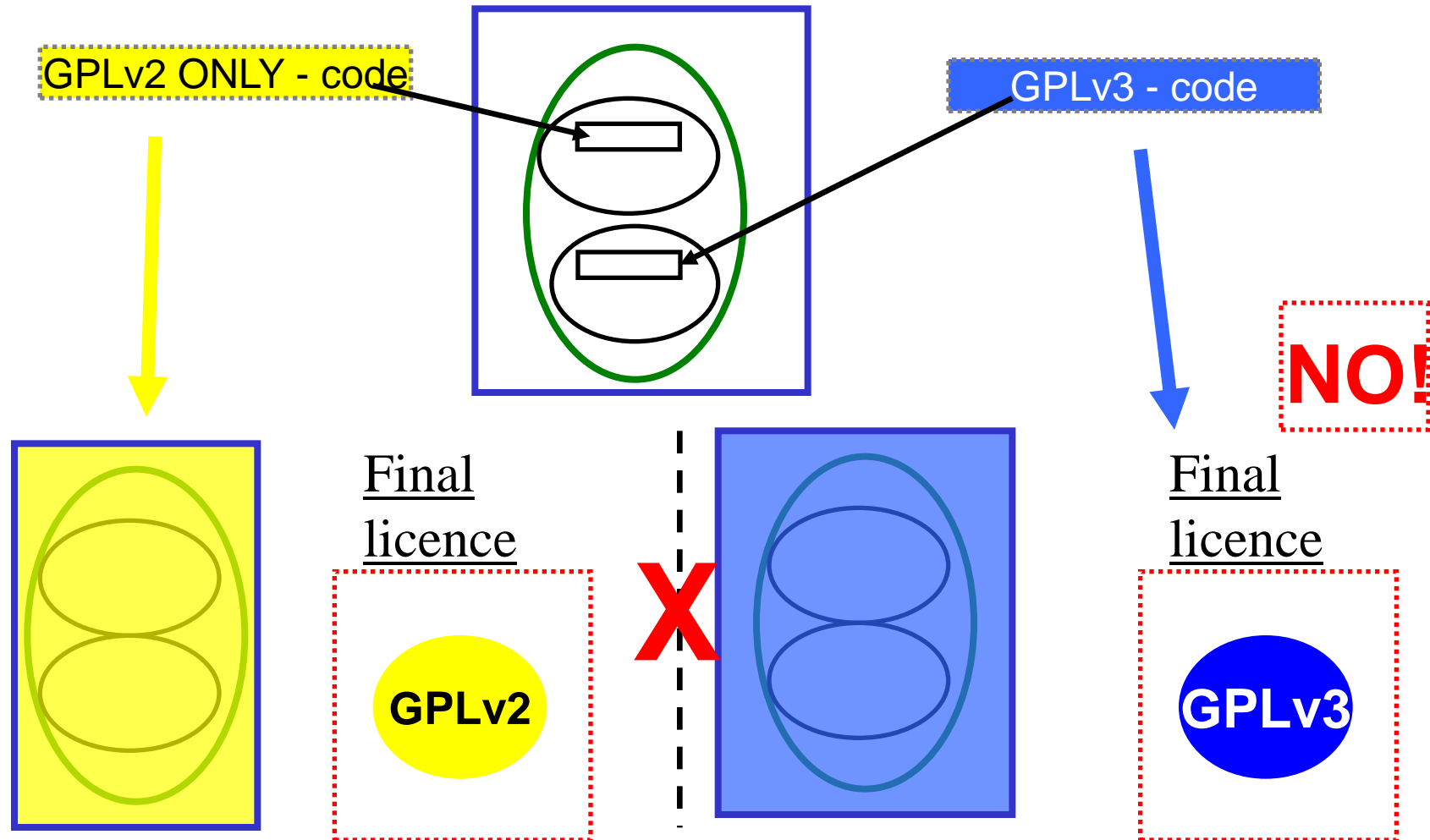


- Notwithstanding any other provision of this License, you have permission **to link or combine** any covered work with a work licensed under version 3 of the GNU Affero General Public License **into a single combined work, and to convey the resulting work.**
- The terms of this License will continue to apply to the part which is the covered work, but **the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.**





NOTE : “GPLv2 ONLY” is **not** compatible with GPLv3





Compatibility Clause of EURL V1.0



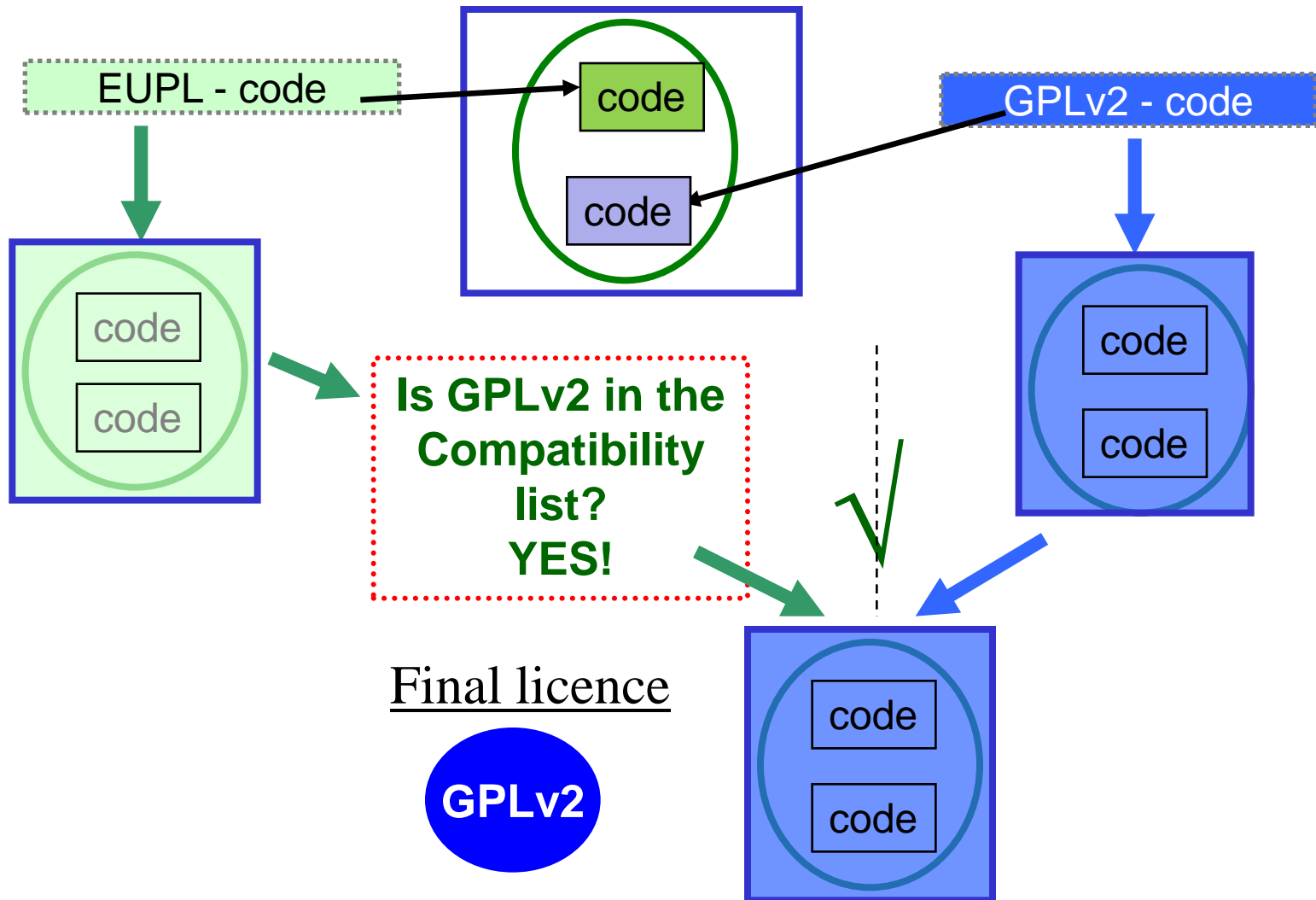
▪ Art. 5, §4

*“Compatibility clause: If the Licensee Distributes and/or Communicates Derivative Works or copies thereof based upon both the Original Work and another work licensed under a **Compatible Licence**, this Distribution and/or Communication can be done under the terms of this **Compatible Licence**. For the sake of this clause, “Compatible Licence” refers to the licences listed in the appendix attached to this Licence. Should the Licensee’s obligations under the **Compatible Licence** conflict with his/her obligations under this Licence, the obligations of the **Compatible Licence** shall prevail.”*

- **MIND** the “unnatural” definition given to Compatible Licence
 - **Compatible Licences** = “ The licences which the EURL is compatible to”
- **LIST** of 5 “Compatible Licences”:
 - *General Public License (GPL) v. 2*
 - *Open Software License (OSL) v. 2.1, v. 3.0*
 - *Common Public License v. 1.0*
 - *Eclipse Public License v. 1.0*
 - *Cecill v. 2.0*



Ex.: EUPL has been rendered GPLv2-compatible thanks to the compatibility clause





CONCLUSION:



Even if licensed under 2 ≠ FLOSS licences,
2 pieces of code cannot sometimes be merged...

- ⇒ **A good licence choice is crucial to the development project**
 - influences the « reusability » of the code
 - influences the business model
 - Influences the acceptability by developers/community
 - ...
- ⇒ **Compatibility issues must be dealt with before and during the development (not after!)**
 - Establish good development practices
 - Inform developers
 - Control the supply chains
 - Foster communication between developers and lawyers
 - Document the developments
 - Use tools (headers checkers, code matchers, etc...)
- ⇒ **“CROSS BORDER” => additional complexity : IPL issues**
 - Which law applies? Which court has jurisdiction?
 - influence on the effects / validity of the licence / part of it?
 - Licences with applicable law clause
 - Simpler (?)
 - Negative influence on compatibility (?)



Thanks for your attention !



Philippe LAURENT

Senior Researcher at CRID
F.U.N.D.P.



Email :

philippe.laurent@fundp.ac.be

<http://www.crid.be>

Attorney at Law
Lawyer at the Brussels Bar

MVVP

Email :

philippe.laurent@mvvp.be

<http://www.mvvp.be>



Wallonia

*Le Fonds Européen de Développement Régional et
la Wallonie investissent dans votre avenir*

