

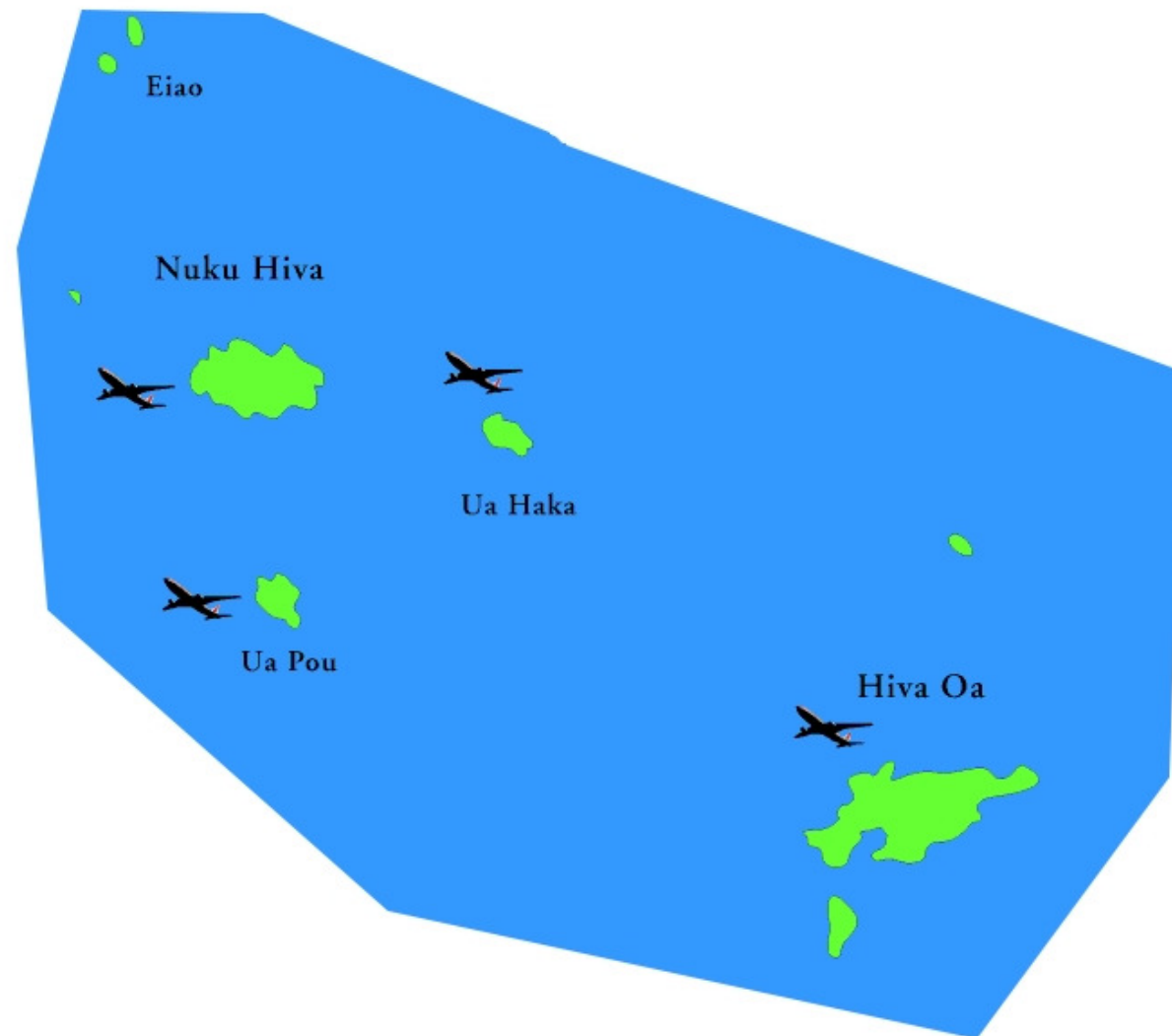
Code siblings: technical and legal implications of copying code Between applications

***Daniel German, Massimiliano Di Penta, Yann-Gaël
Guéhéneuc, and Giuliano (Giulio) Antoniol***

The Challenge

- Code, as any other artistic production, is regulated by copyright law
- Companies own the property of source code
- Free and open source software (FOSS) model is different
- Copying 27 LOC out of 525 KLOC resulted in a copyright infringement
- Users and companies must be aware of copyright law and ownership

Code Has Preferential Migration Flows



License Types

- Permissive – the MIT/X11 and BSD licenses
 - Minor constraints on the licensee
 - Inclusion of fragments in a system under a different license
 - BSD licensed fragments can be included in proprietary systems.
 - CAVEAT! Multiple BSD licenses: original BSD (4-clauses BSD), the new BSD (3-clauses BSD), and the 2-clauses BSD
 - **Code licensed under the original 4-clauses BSD cannot be included inside systems licensed under the GPL**
- Reciprocal – GNU variants
 - Any system that includes the fragments must be licensed under the same license
 - GPL-licensed fragments can only be included in systems licensed under the same version of the GPL

The Scale of the Problem

- Widely adopted systems are in the range of MLOC and thousands of files
- If 27LOC in 525KLOC lead to copyright infringement
 - Companies implication in reusing code
 - End user implications
- We are like detectives
 - Help monitoring and detecting license inconsistencies
 - Help monitoring and identifying inconsistent licenses in code fragments

Empirical Study

- Code siblings: code fragments that migrated from one system to another and then evolved following their own paths
- Three *nix kernels
 - Linux ~7MLOC and 20,000 files
 - FreeBSD ~8MLOC and 21,000 files
 - OpenBSD ~2MLOC and 5,500 files
- Overall Size as of Jan. 2009, 17MLOC

Research Questions

- RQ1: What kinds of open source licenses are used in the three kernels?
- RQ2: How many potential siblings exist between the BSD kernels and the Linux kernel?
- RQ3: What licenses are used by siblings and, if different, why?

Technologies and Setup

- Clone detection tool
 - *CCFinderX* tool
 - Min 100 tokens
 - Parse only .c files
 - Concentrate on pair of files sharing a high percentage of common code fragment, least ~30%, i.e., ~20LOC
 - Prune files mapped into more than five siblings
- License detection and identification
 - First comment(s)
 - FoSSology version 1.0.0
 - 78 different license variants
 - Added 5 more licenses

Sibling(s) Origin

- Identify current siblings
- Trace back into past siblings – their code fragments in the same files
- When they disappear, then we have their origins
- Take the oldest of the two as the true origin

Sys 1 – File i



Cloned fragments

**Migration
direction**

siblings

Sys 2 – File j

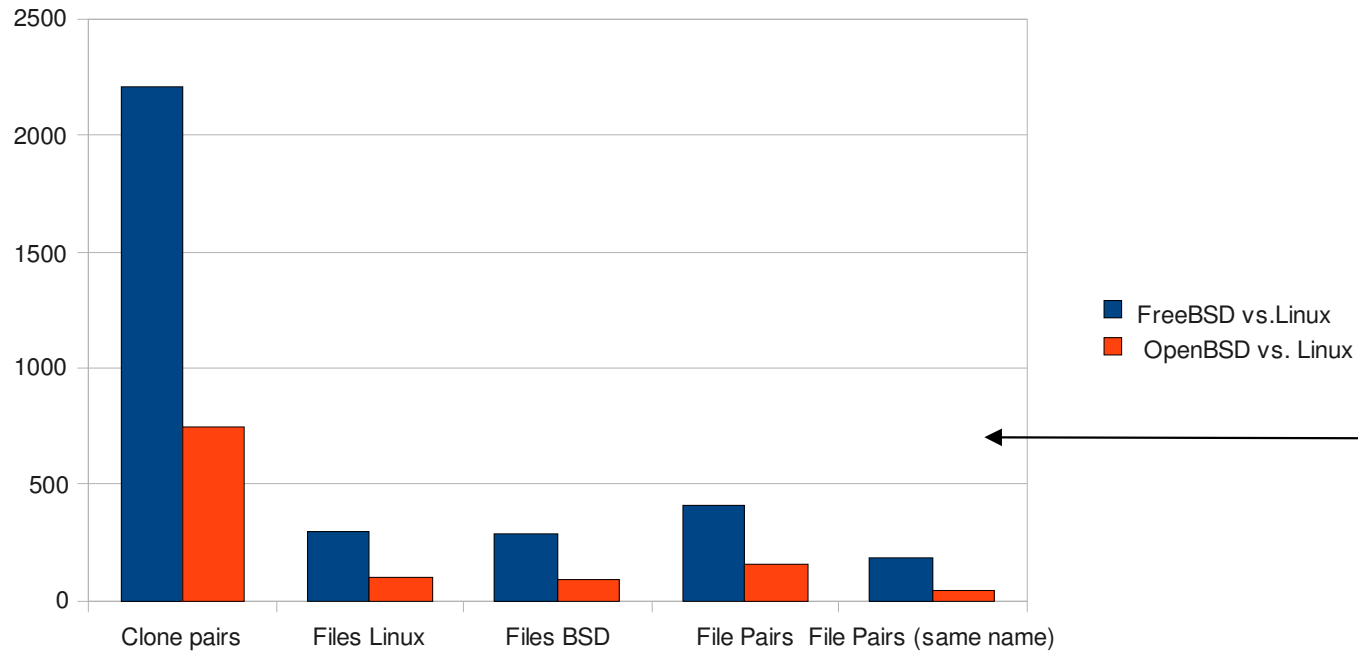


Cloned fragments

RQ1: Kinds of open source licenses

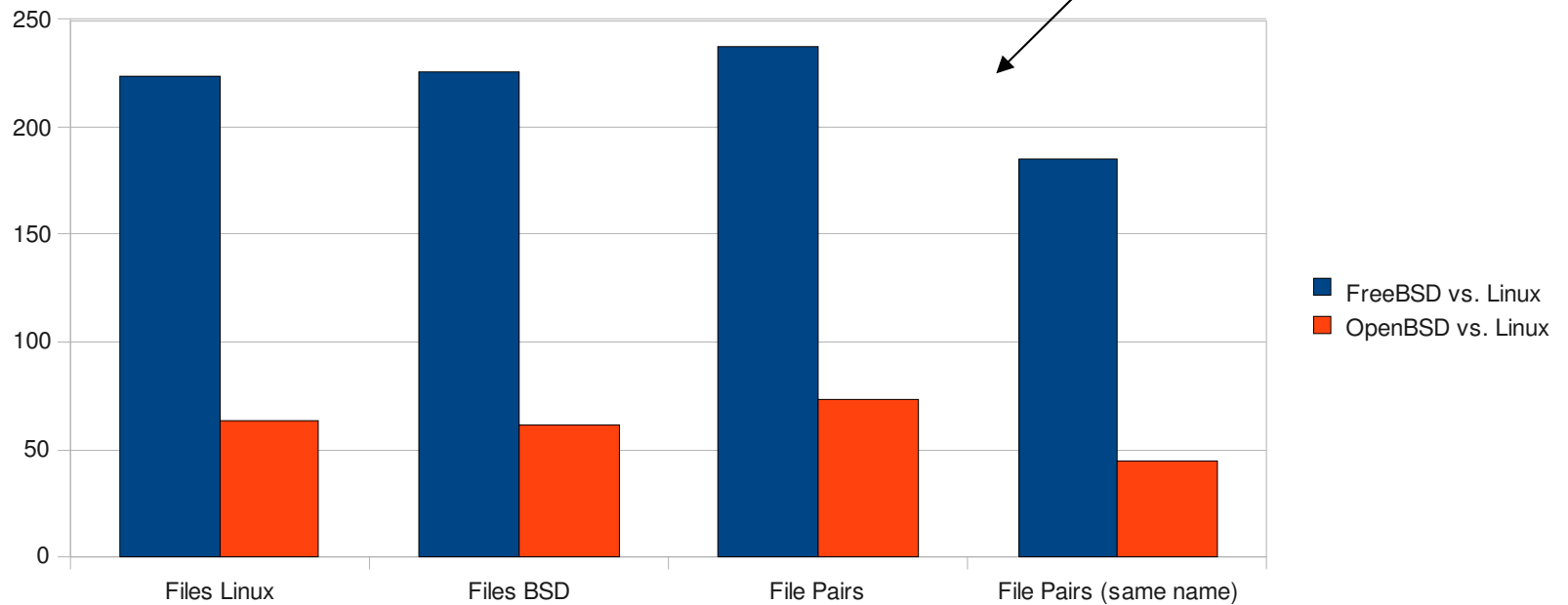
- Linux... is Linux... 65% of GPL files plus 25% of files “promoted” to GPL by L. Torvald
 - A few files (35) have two licenses
- FreeBSD 75% of the files with BSD license
 - 189 files (5%) with no license
 - 179 files with a corporate license (Intel licenses)
 - 167 files with MIT license
 - A few multiple licenses – 19 BSD and GPL, 15 BSD and Educational, 14 MIT and GPL
- OpenBSD 76 % BSD licenses
 - 295 files (9%) with a MIT license, 179 with an educational license
 - 138 (84%) without license
 - 59 files with BSD and Educational, 25 with MIT and BSD, and 14 with BSD and GPL

RQ2: Siblings between kernels



Siblings

Filtered siblings



RQ3: Code Migration and Licenses

FreeBSD	Linux	Files
BSD	GPL	8
BSD	MIT	2
BSD	None	2
Corporate	BSD+GPL	89
GPL	None	1
Phrase	BSD+GPL	1
X.Net+BSD	MIT	1

Before Jan 1, 2002
Almost nothing after

OpenBSD	Linux	Files
BSD	BSD+GPL	1
BSD	MIT	2
BSD	Unknown	1
BSD+GPL	GPL	1
BSD+Phrase	Phrase+GPL	1
MIT	GPL	23

Linux	FreeBSD	Files
BSD+GPL	Corporate	8
GPL	BSD	17
GPL	BSD+GPL	1
GPL	CPL+BSD+GPL	1
MIT	BSD	1
MIT+GPL	None	2
None	BSD	1
Phrase+GPL	MIT	2

After Jan 1, 2002
Nothing before

AIC7xxx Maintaining Siblings

- 1994: Linux AIC7xxx series SCSI adapters
- 1995: Linux code is incorporated into an OpenBSD driver
- 1996: NetBSD driver is ported to FreeBSD
 - #ifdef to maintain the variants
- 1997: A mailing list is created in FreeBSD to unify the efforts of people in the different kernels
 - The major development of the driver seems to happen in FreeBSD
- 2000: Development propagates to Linux, NetBSD, and OpenBSD
- Today: Development mostly Linux and FreeBSD

GPC code in FreeBSD

- 2002: Silicon Graphics xfs file system integrated into Linux
- Dec 12, 2005 xfs appears in FreeBSD
 - The license of xfs is GPL
 - FreeBSD is licensed under the 2-clause BSD
 - Including xfs in a BSD kernel requires the kernel to be under the GPL too a **contradiction!**
- Compiling GPL-licensed code into the kernel makes it “RESTRICTED”
 - It can no longer be distributed in binary form, its source code be made available for mirroring

License Defects

- FreeBSD rdma_cma.c / Linux cdma.c are siblings
- In Linux, it appeared on Jun 17, 2006, with 64 changes plus including 8 changes after it appeared in FreeBSD
- The Linux sibling is licensed under GPL v2 and the 2-clause BSD licenses
- The FreeBSD sibling is licensed under the terms of the new BSD license, the GPL v2, and Commons Public License
- Original license still present in FreeBSD
- **Linux license was changed:**

```
commit a9474917099e007c0f51d5474394b5890111614f
```

```
Author: Sean Hefty <sean.hefty@intel.com>
```

```
Date: Mon Jul 14 23:48:43 2008 -0700
```

```
RDMA: Fix license text
```

```
The license text for several files references a third software license that was inadvertently copied in. Update the license to what was intended. This update was based on a request from HP. [..]
```

Conclusion

- Code move and code siblings do exist
- Siblings have a preferential flow
 - Initially from BSD(s) to Linux – frequent
 - Today from Linux to FreeBSD – less frequent
- Companies directly contribute to code in different kernels – see Intel drivers with dual licenses
- Managing siblings is a difficult problem

If you don't monitor code may sneak in ...

Questions ?