Outline

1. Open Source Software
   - licensing software
   - copyright and copyleft
2. Modules and Packages
   - a package is a collection of modules
   - pair programming and code review

MCS 260 Lecture 25
Introduction to Computer Science
Jan Verschelde, 12 July 2023

software licensing
modules and packages

1. Open Source Software
   - licensing software
   - copyright and copyleft

2. Modules and Packages
   - a package is a collection of modules
   - pair programming and code review

History of Open Source

In the 1960s, software came free with hardware.
Users were programmers and software developers.

In the mid 1970s, software became a commercial product.
Personal Computing meant that users did not program.

Unix, developed at Bell Labs and University of Berkeley (BSD) became a commercial product of AT&T in 1984.

GNU (not Unix) project started at MIT in 1984 by Richard Stallman.
The Free Software Foundation published a first version of the General Public License (GPL) in 1988.

In 1991, a first version of what later became Linux is released.
Linux = Linus+Unix, started by Linus Torvalds.

MIT started the OpenCourseWare project in 2000.

Intellectual Property Rights

and software

Intellectual property rights (IPR) grant authors the possibility to control copying, modifying, and distributing products.

Four tools to protect IPR:

trade secret confidential information used to compete

trademark is word, name, symbol used to distinguish

copyright grants authors exclusive rights to copy, sell, license, distribute, modify, translate, etc.

patent gives creators exclusive rights to manufacture and sell invented goods.

A commercial program is typically delivered by giving only its object code, in contrast to open source.
Licenses may explicity forbid reverse engineering of code.

General Public License (GPL)

Motivation: needs and benefits of open source.

1. Mission critical and scientific applications place a very strong emphasis on correctness.
2. *Given enough eyeballs, all bugs are shallow*.

Note: free as in free speech, not as in free beer.

Software is also a service.

software licensing
modules and packages

1. Open Source Software
   - licensing software
   - copyright and copyleft

2. Modules and Packages
   - a package is a collection of modules
   - pair programming and code review

copy*left* versus copy*right*

Copy*left* as the reverse of copy*right*:

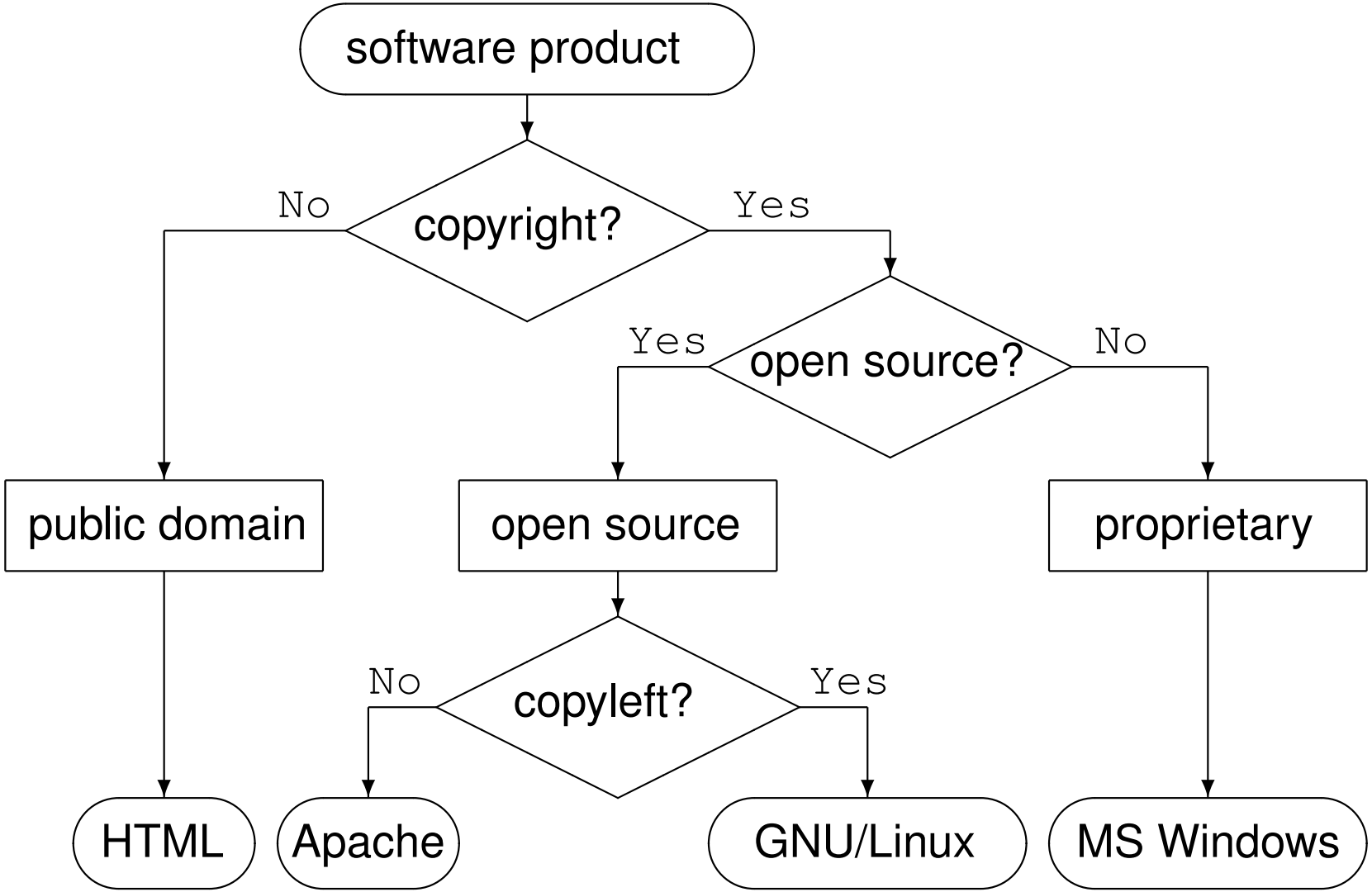
1. first state that the code is copyrighted
2. give everyone all rights on the code
3. *only if the distribution terms are unchanged*

The code and the freedoms become legally inseparable.

The license imposes a ‘share alike’ restriction.

Many open source licenses exist, some of the most popular ones: GNU GPL, GNU Library or Lesser GPL, Mozilla Public License, and Berkeley Software Distribution.

copyright applied to software



M. Muffatto: *Open Source. A Multidisciplinary Approach*, 2006.

computer ethics

Intellectual property rights are an important chapter of ethical issues of computing.
Computers have an enormous impact on society.

Other various social, legal, often controversional issues are

1. privacy and personal information
2. encryption and interception of communications
3. can we trust the computer?
4. freedom of speech
5. computer crime
6. computers and work
7. who benefits most?
8. professional code of ethics

software licensing
modules and packages

1. Open Source Software
   - licensing software
   - copyright and copyleft

2. Modules and Packages
   - a package is a collection of modules
   - pair programming and code review

attributes, locals(), globals()

The three attributes in the module `math`:

`math.__doc__` a documentation string

`math.__file__` location for the static object

`math.__name__` name of the module (without path)

The functions `locals()` and `globals()` take no argument.
They return local and global names.
On return are dictionaries, the keys are strings.

Application: see if a variable is already defined:

```
>>> 'x' in globals()
False
>>> x = 2
>>> 'x' in globals()
True
```

packages: hierarchical file directory structure

Modules in larger software systems are decomposed again into modules.

Python provides multi-modules packages, with naming as

```
< package name > . < module name >
```

NumPy is a multi-module package:

```
>>> import numpy
>>> import numpy.linalg
```

The `linalg` provides linear algebra operations.

Making our own Package

Recall the automated quiz of last lecture.
The code for program is defined in four files.
Each file is one module.

The steps to define a package:

1. create the package directory (or folder)
2. make the initializer module `__init__.py`
3. test the import of the package
4. add statements to `__init__.py`
5. providing a description of the package
6. define `setup()`

The first two steps are essential to make a package.

software licensing
modules and packages

1. Open Source Software
   - licensing software
   - copyright and copyleft

2. Modules and Packages
   - a package is a collection of modules
   - pair programming and code review

Programming in Pairs

To write programs in pairs, two methodologies can be applied.

1. pair programming

   Two developers share the same computer.

   - The driver codes.
   - The navigator reads and checks the code.

   The roles switch from time to time.

2. code review

   Two developers share their code.

   - The reviewer suggests changes to the code.
   - The coder adjusts the code based on the review.

   The roles switch from time to time.

Exercises

1. Visit `www.opensource.org` and `www.gnu.org`.
   Write two paragraphs (one after each visit), about what you have learned from visiting those two sides.
2. Where can you find the Python license on your computer?
   Compare the Python license with GNU GPL and summarize the similarities and differences.
3. Give the Python commands to remove the first element of the module search path. Show also the output of the commands.
4. Read `http://www.acm.org/about/code-of-ethics`.
   Write one paragraph to summarize the essence of this code.
5. Download and install `NumPy`.
   Solve a linear system and verify its answer.