How to Understand FOSS
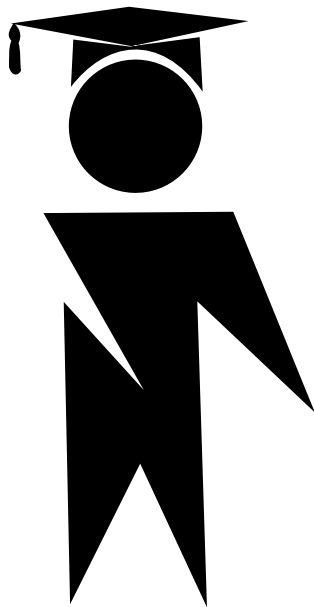
a visual guide to

Free &
Open
Source
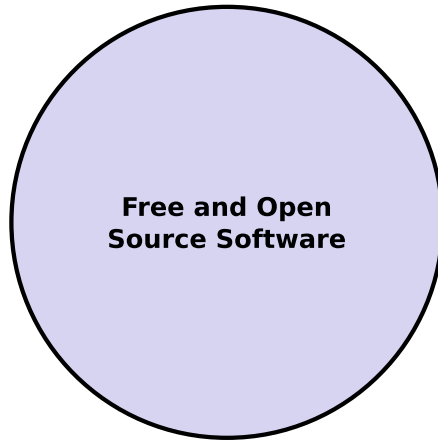# Software

by Martin Owens

FOSS
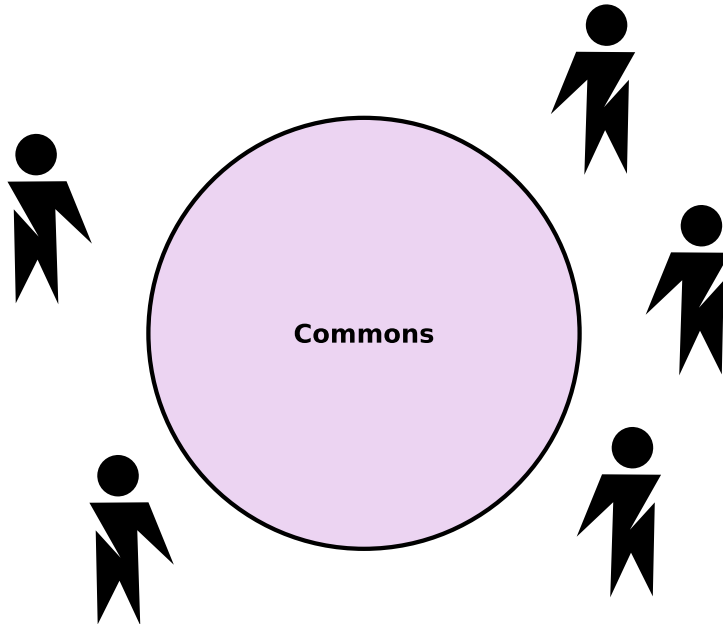
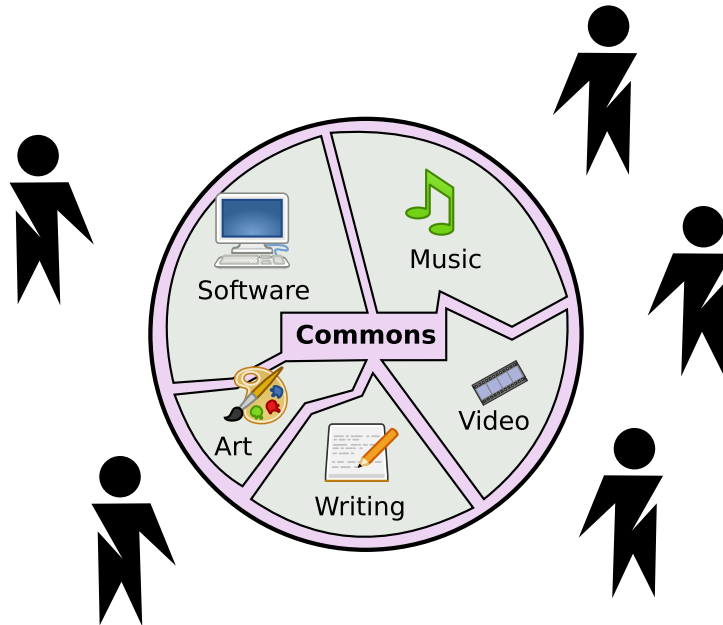**Is a Creative Commons...**

**Free and Open
Source Software**

The Commons

Resources that are accessible to everyone,
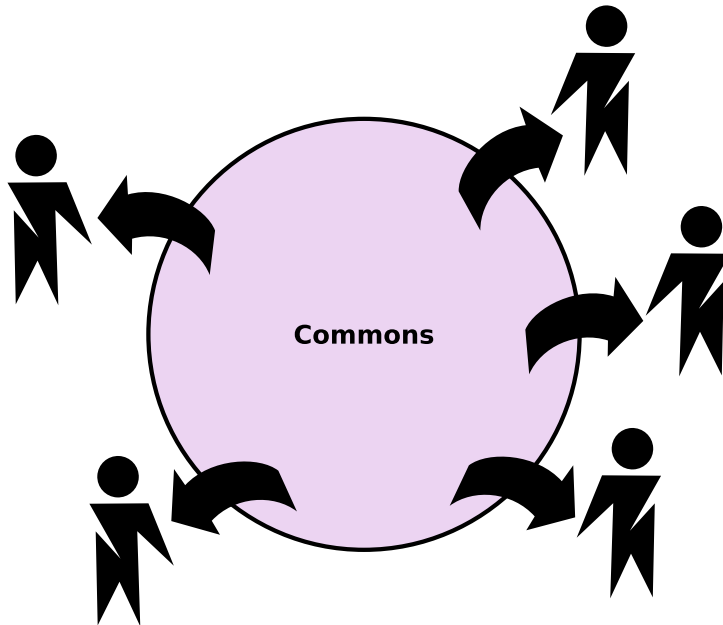Regardless of country or ability to pay.

Commons

The Commons

These resources can be anything.
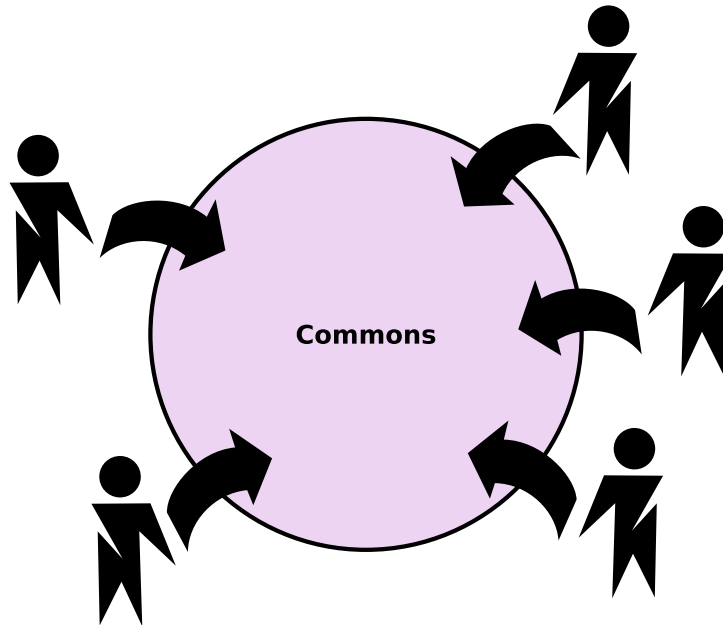**The Creative Commons is for all types of creative works.**

The Commons

Everyone can make copies of anything
in the Creative Commons.
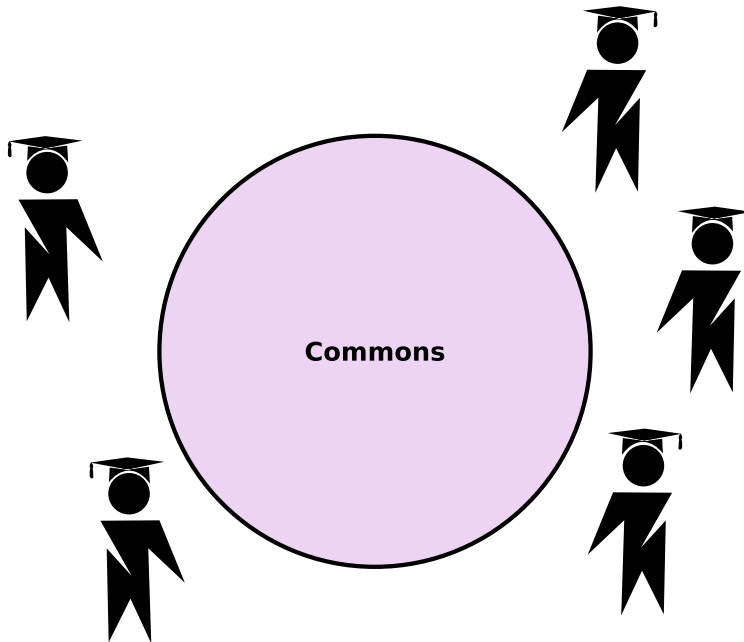
Commons

The Commons

**Everyone can add anything
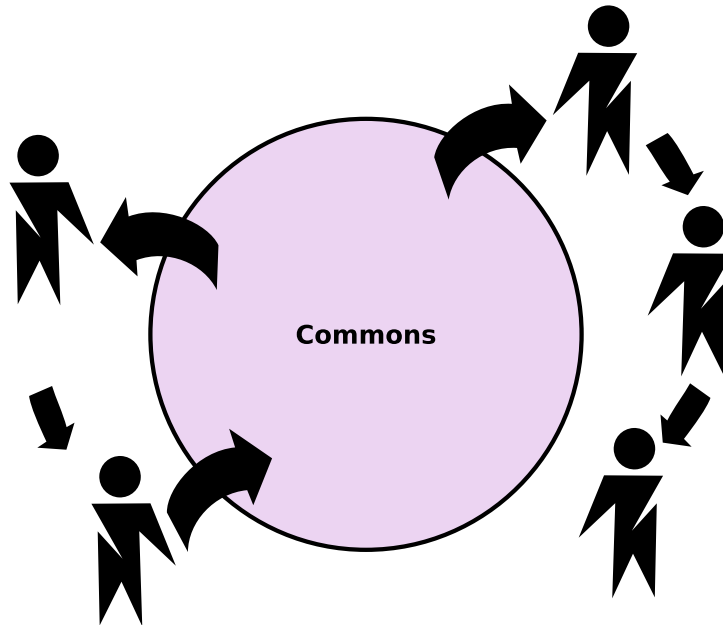to the Creative Commons.**

Commons

The Commons

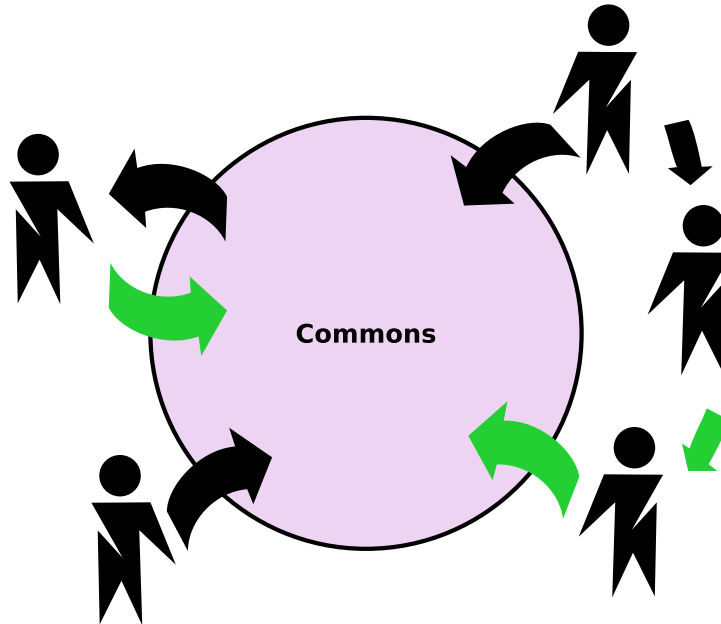Everyone can learn from
the Creative Commons.

Commons

The Commons

Anyone can share the Creative Content with anybody else.

Commons

The Commons

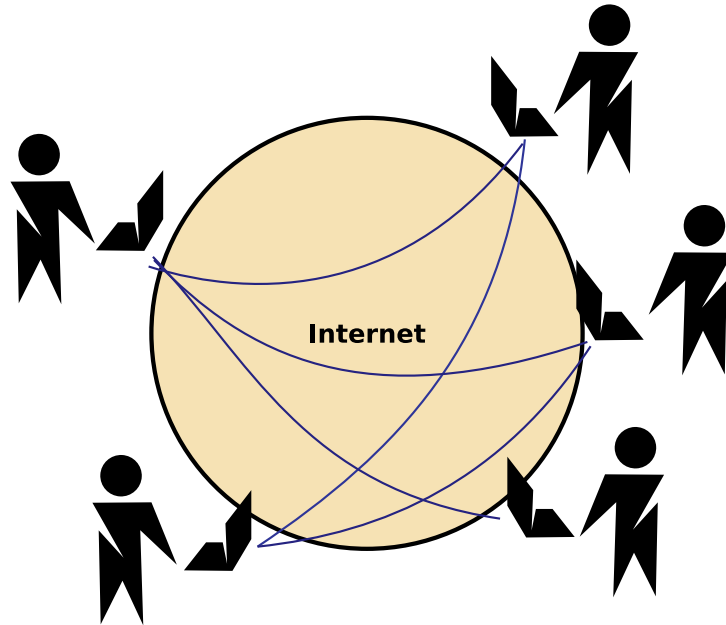Because the source is available, anyone can build
on other works to create new "derivative" works.

The Internet

The Internet allows this free exchange
of ideas and creative works across the whole world.

Internet

Closed Source

Closed source is the same as Non-Free
it is the creation of content as a private property freely
accessible only to the owner (company or person).
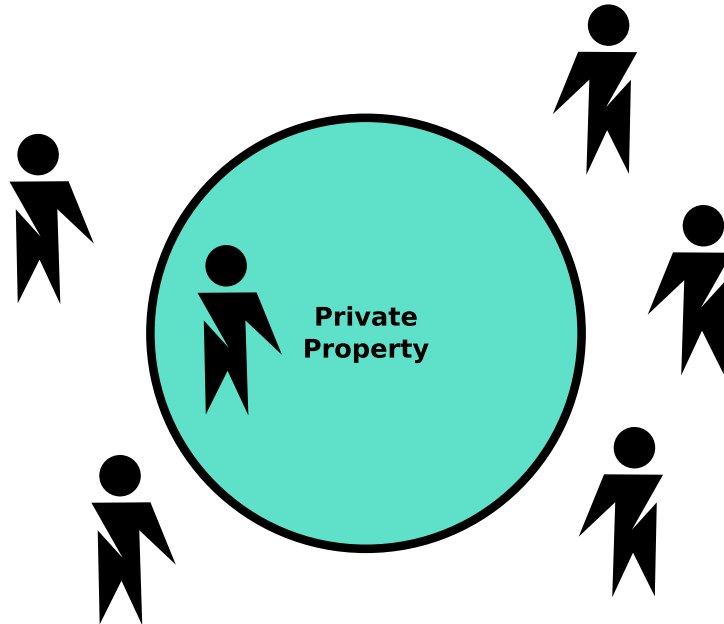
**Private
Property**

Closed Source

The Private Property model does not allow any sharing, building upon or learning to happen.

Customer

Customer

Private Property

Customer

Employee
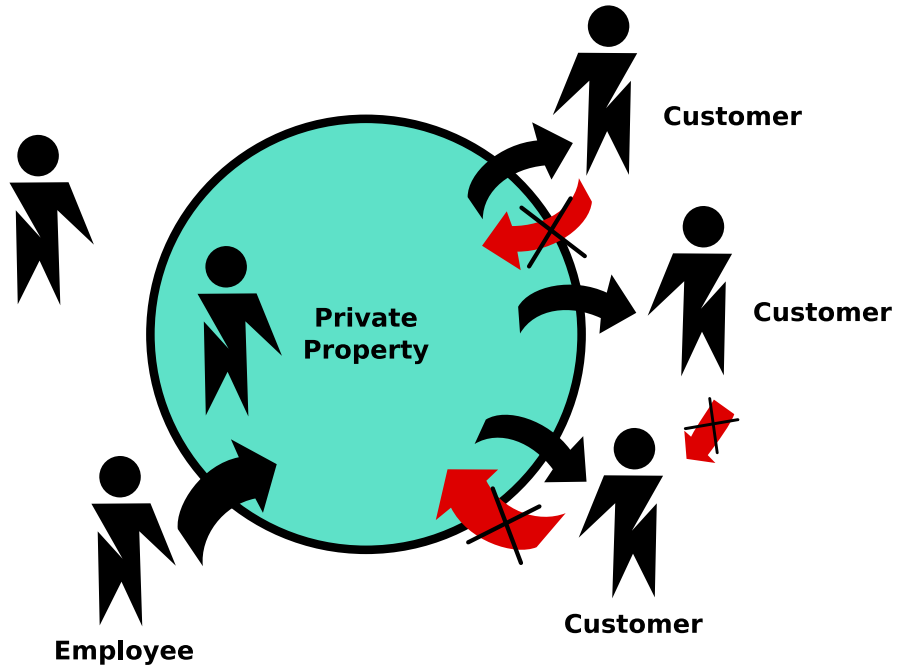
Closed Source

The user has only one right, the right to use.
And the only choice is to buy again or find an alternative.

Customer

Customer

Private Property

Customer
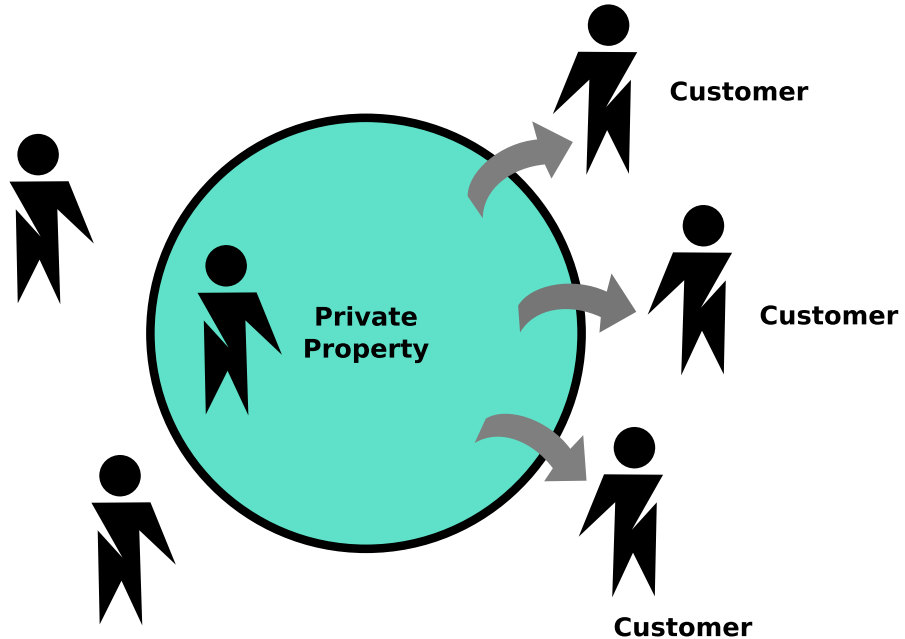
FOSS

That is why FOSS is a Commons system, not a Private Property system. This has some implications....

Free and Open Source Software Commons
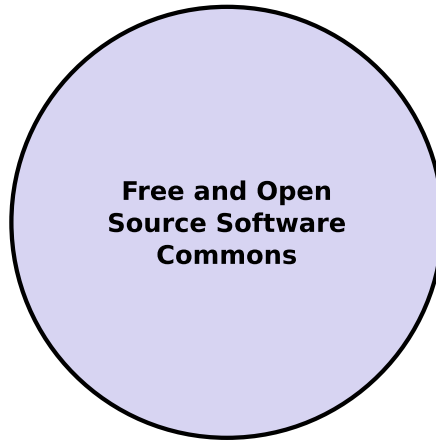
Once Created

Once software has been created and placed into the Commons
It can be used by anyone, for any reason, at any time.
The number of copies is never limited.

Creation Costs

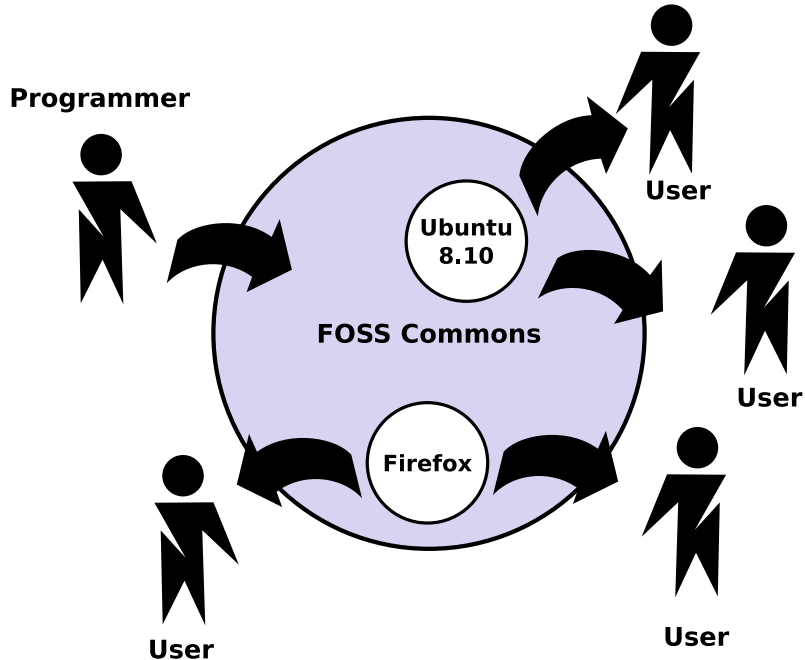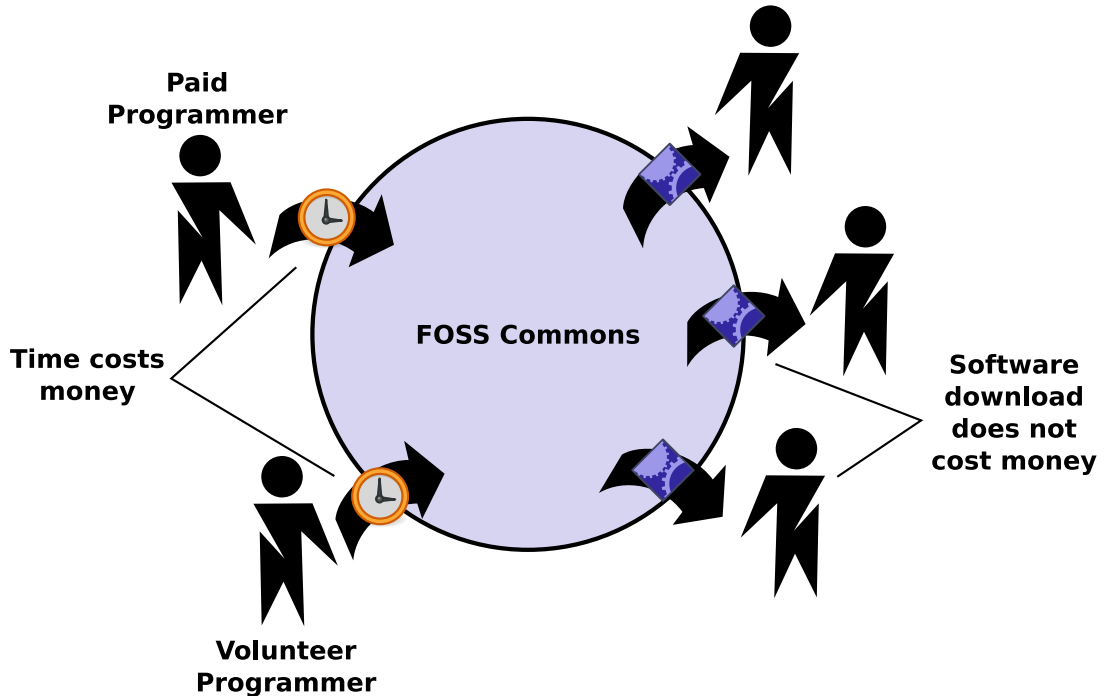**But software creation, new features and bug fixes do cost resources (money, time, etc).**

Paid Programmer

Time costs money

FOSS Commons

Volunteer Programmer

Software download does not cost money

Creation Costs

Compared to the costs of closed source.

Customer

Employee Programmer

Customer

Private Property

Time is paid for by investment of the owner. Users uninvolved.

Every copy costs money

Employee Programmer

Customer
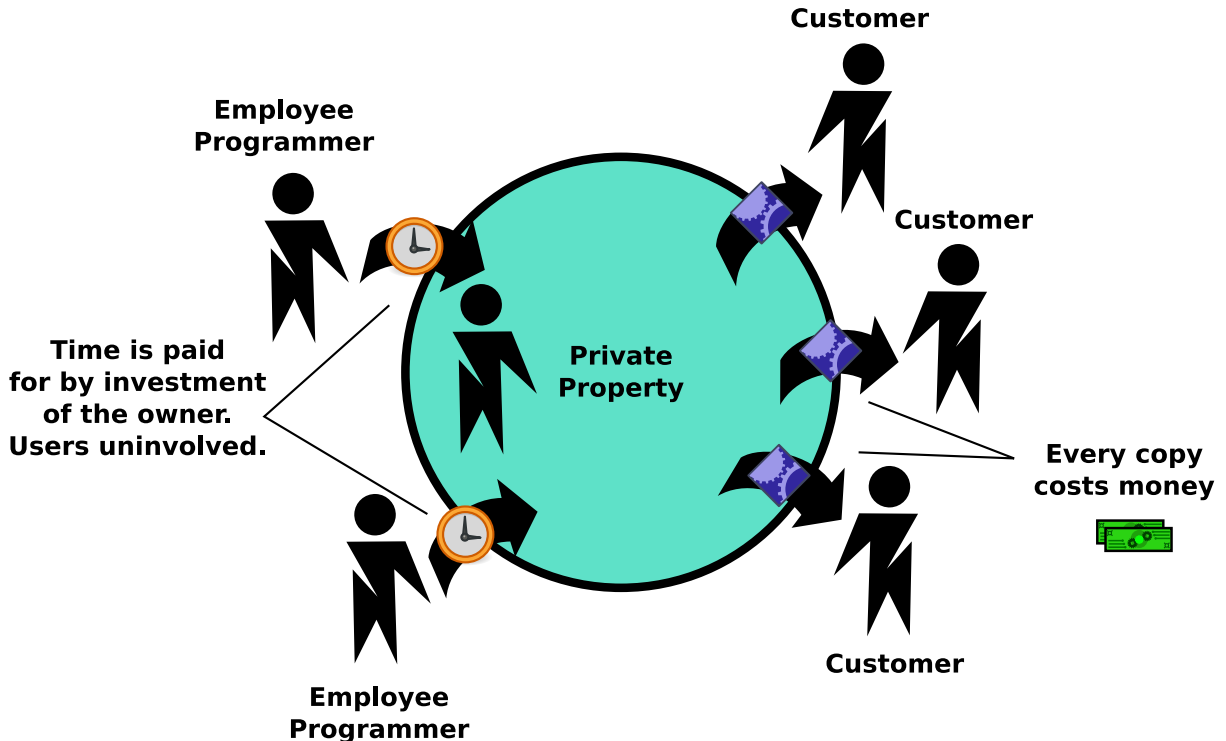
Paying the Costs

Supporting development costs requires resources.
Example:

Programmer
(ready to work)

Programmer
(needs feature)

FOSS Commons

User
(needs feature)

User
(needs nothing)
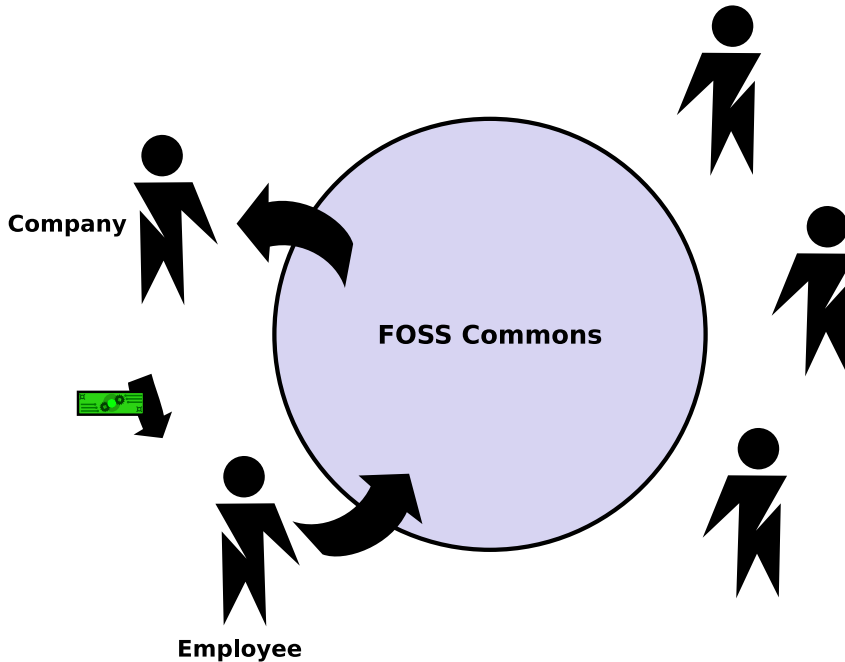
User
(needs feature)
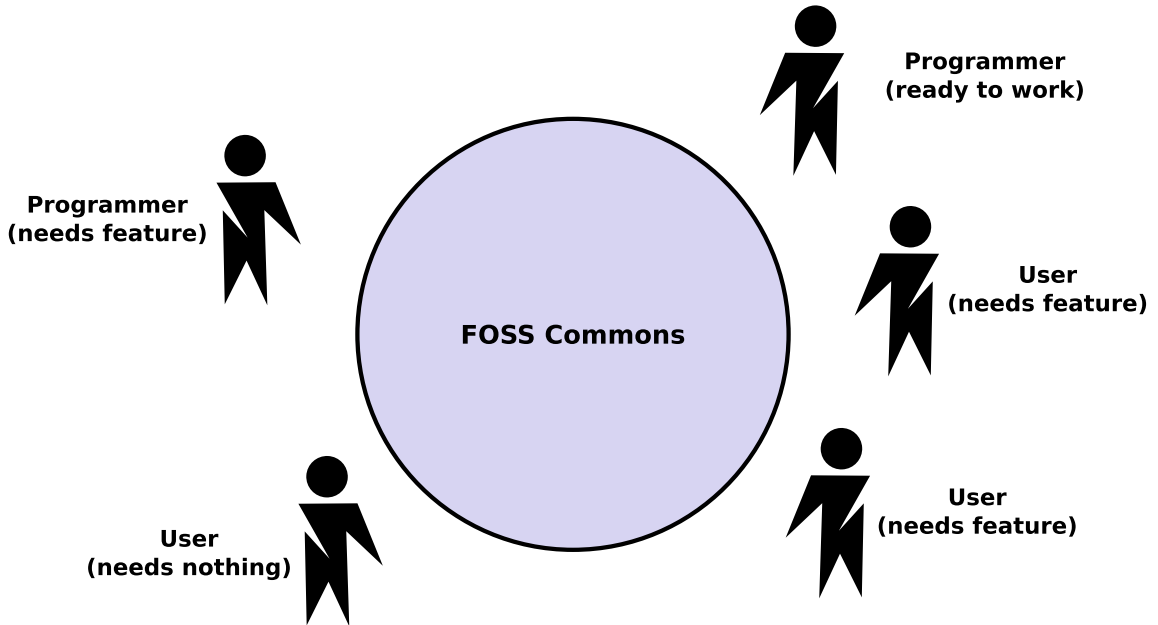
Enlightened Self Interest

A skilled programmer with access to the source code can serve her own needs, giving back to the community in the process.
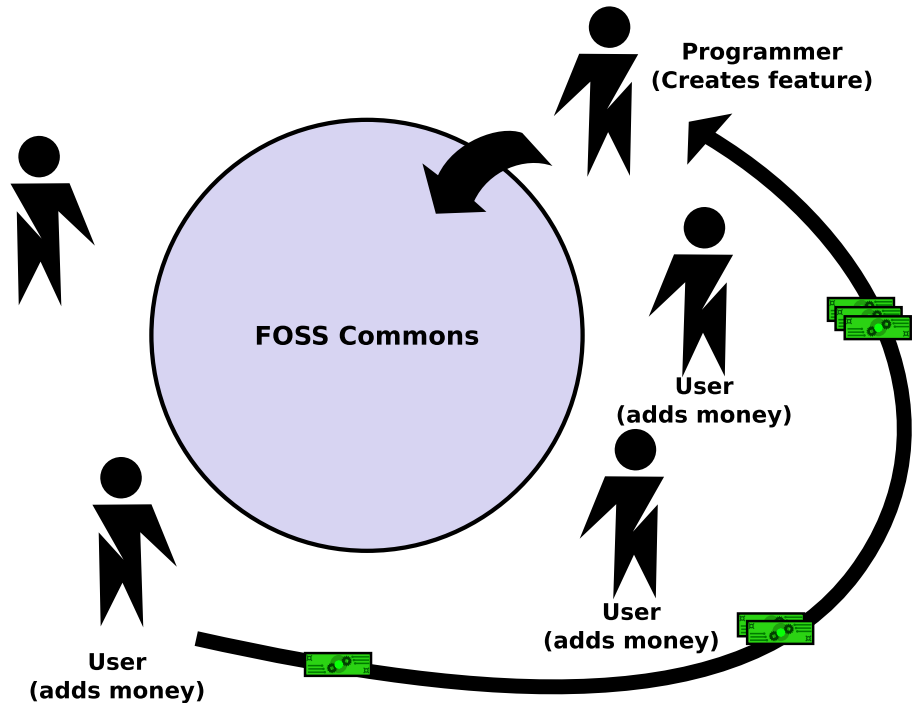
Programmer (adds features)

FOSS Commons
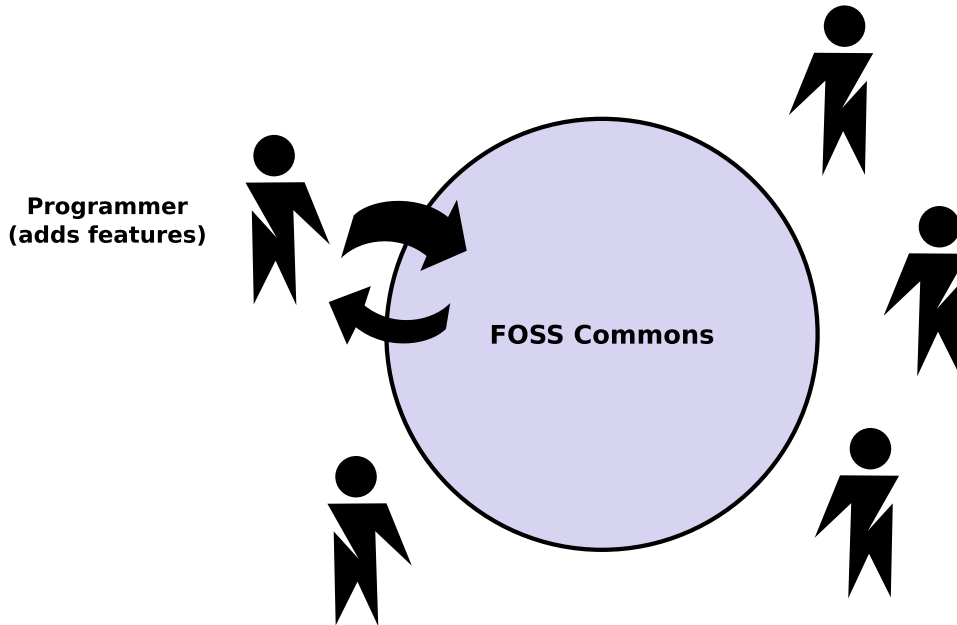
Employed

An organisation can use an employee to serve
their software development needs in the commons.

Aggregated Funding

If the programmer can aggregate money from multiple users, she can get paid for creating software they need.


Programmer
(Creates feature)

FOSS Commons

User
(adds money)

User
(adds money)

User
(adds money)

Choice of Support

Because users get to choose who supports and develops their software, they have the choice of any supplier.

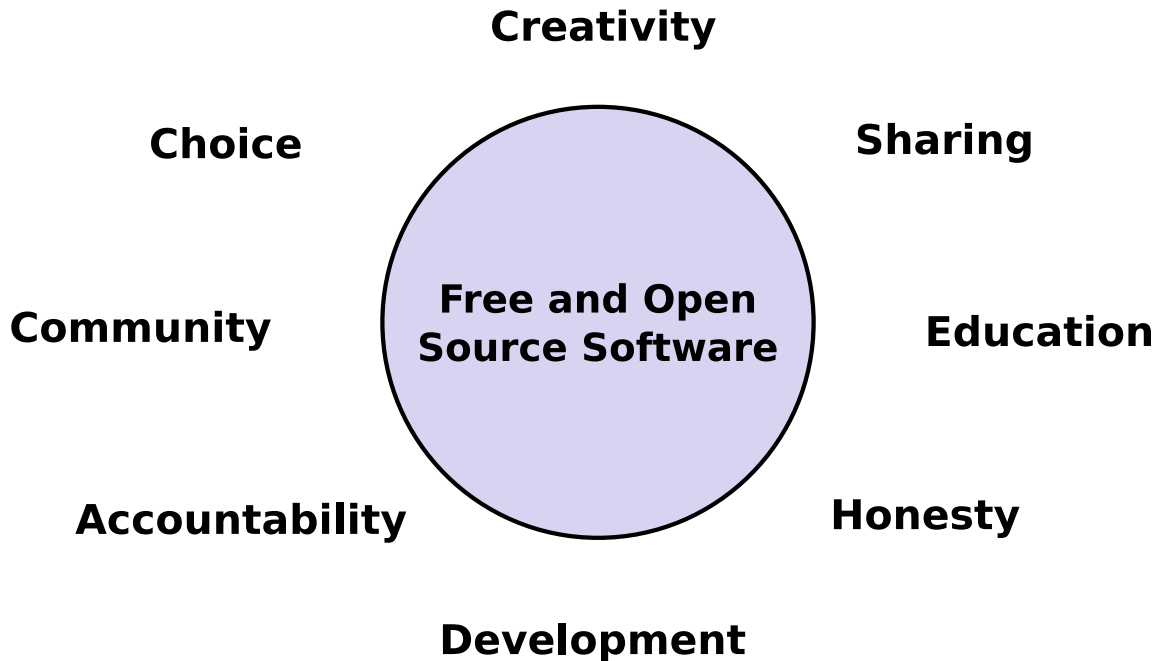Support Company A

User

Support Company B

FOSS Commons
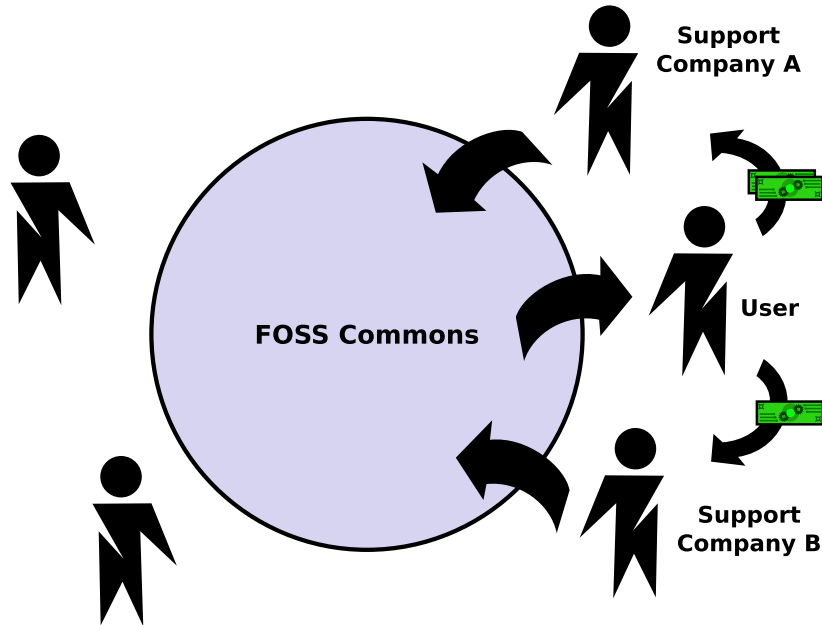
Community

Programmers, organisations and users are all able to work with each other in a community to further the development.
With each contributing member, the pace of development increases.
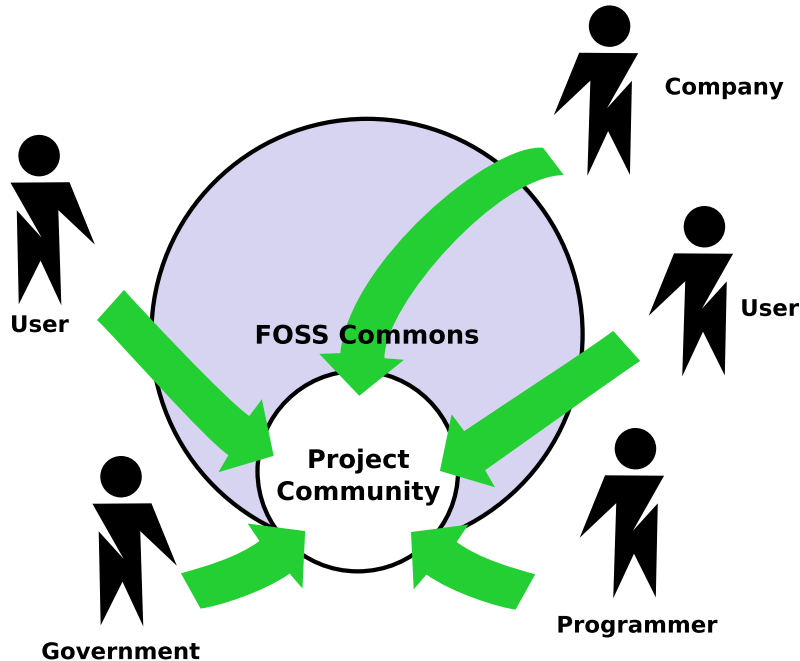
Last Words

FOSS development has many advantages, hopefully this guide has helped you understand how it works.
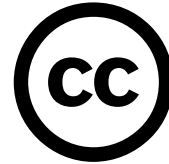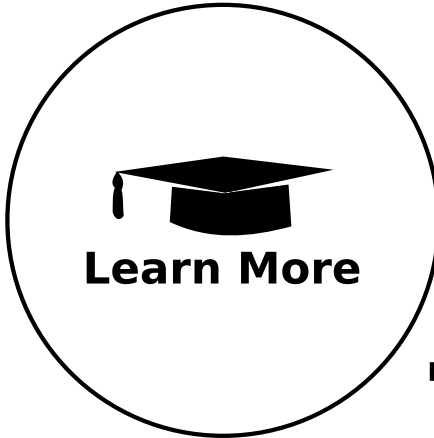
Creativity

Choice

Sharing

Community

Free and Open Source Software

Education

Accountability

Honesty

Development

The End

Thank you for reading.

Free Software Foundation
http://www.fsf.org/

Creative Commons
http://creativecommons.org/

Learn More

Open Source
http://www.opensource.org/

Electronic Frontier Foundation
http://www.eff.org/