

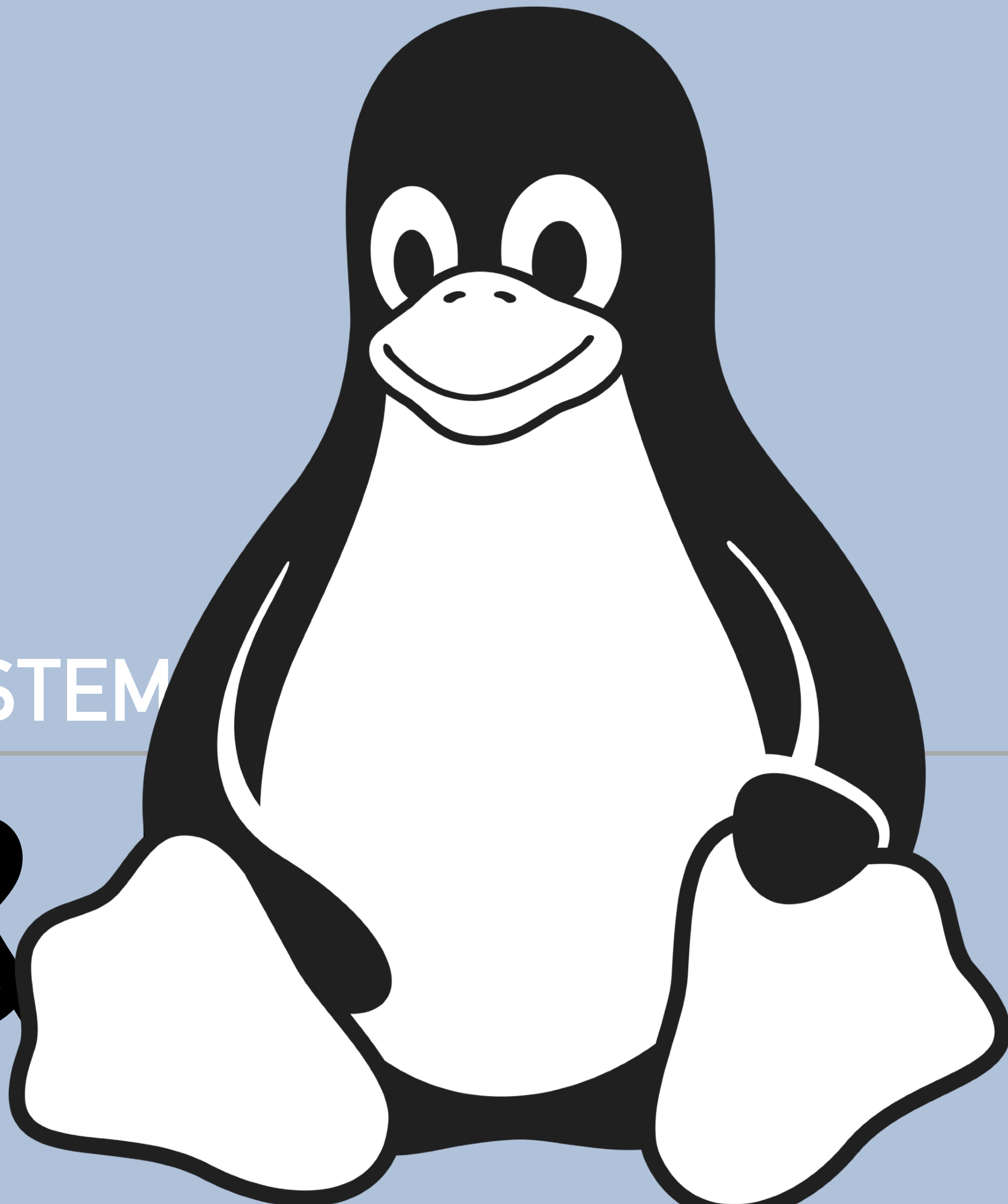
OPERATING SYSTEMS

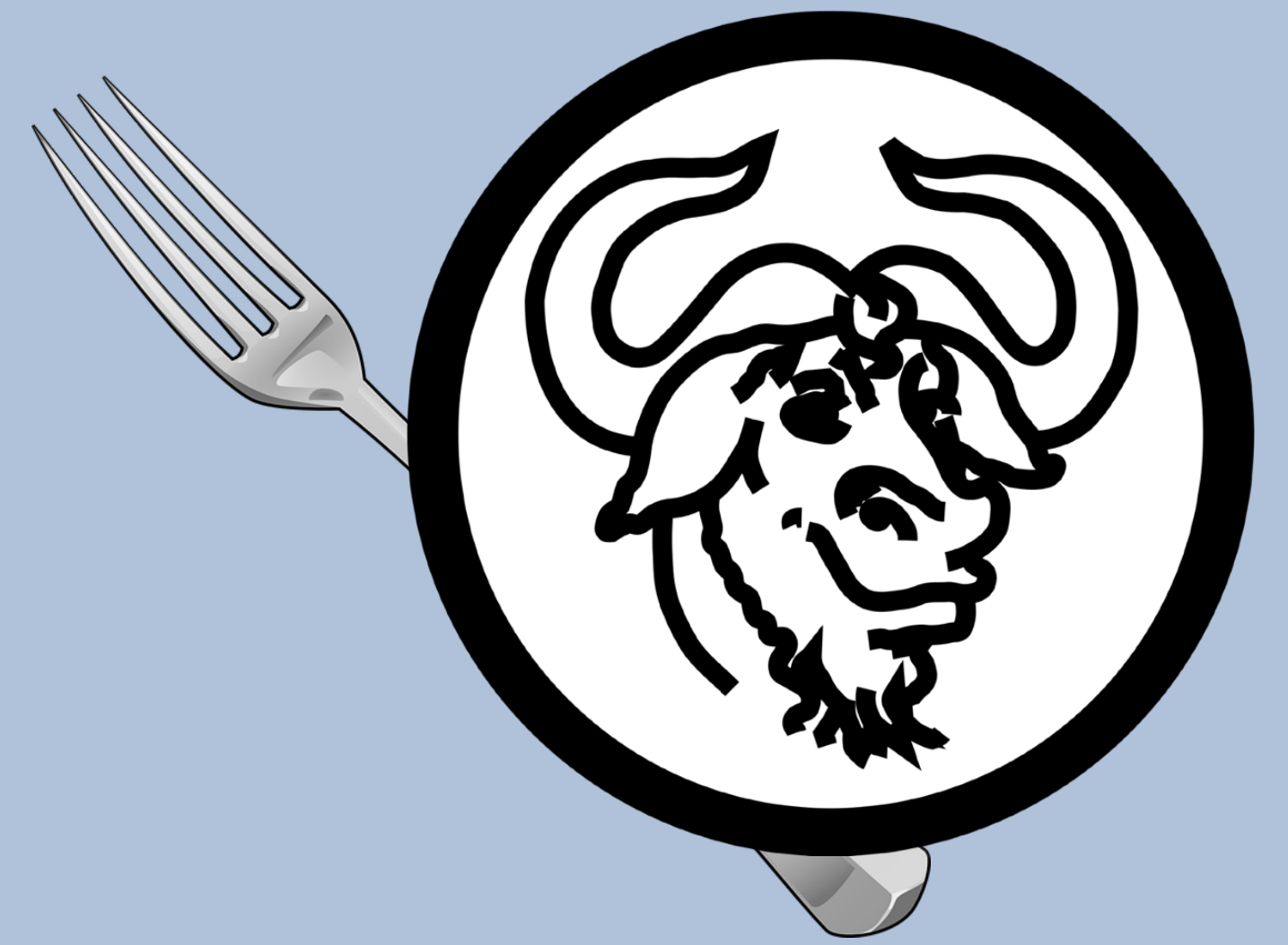
GNU & LINUX

OPERATING SYSTEM



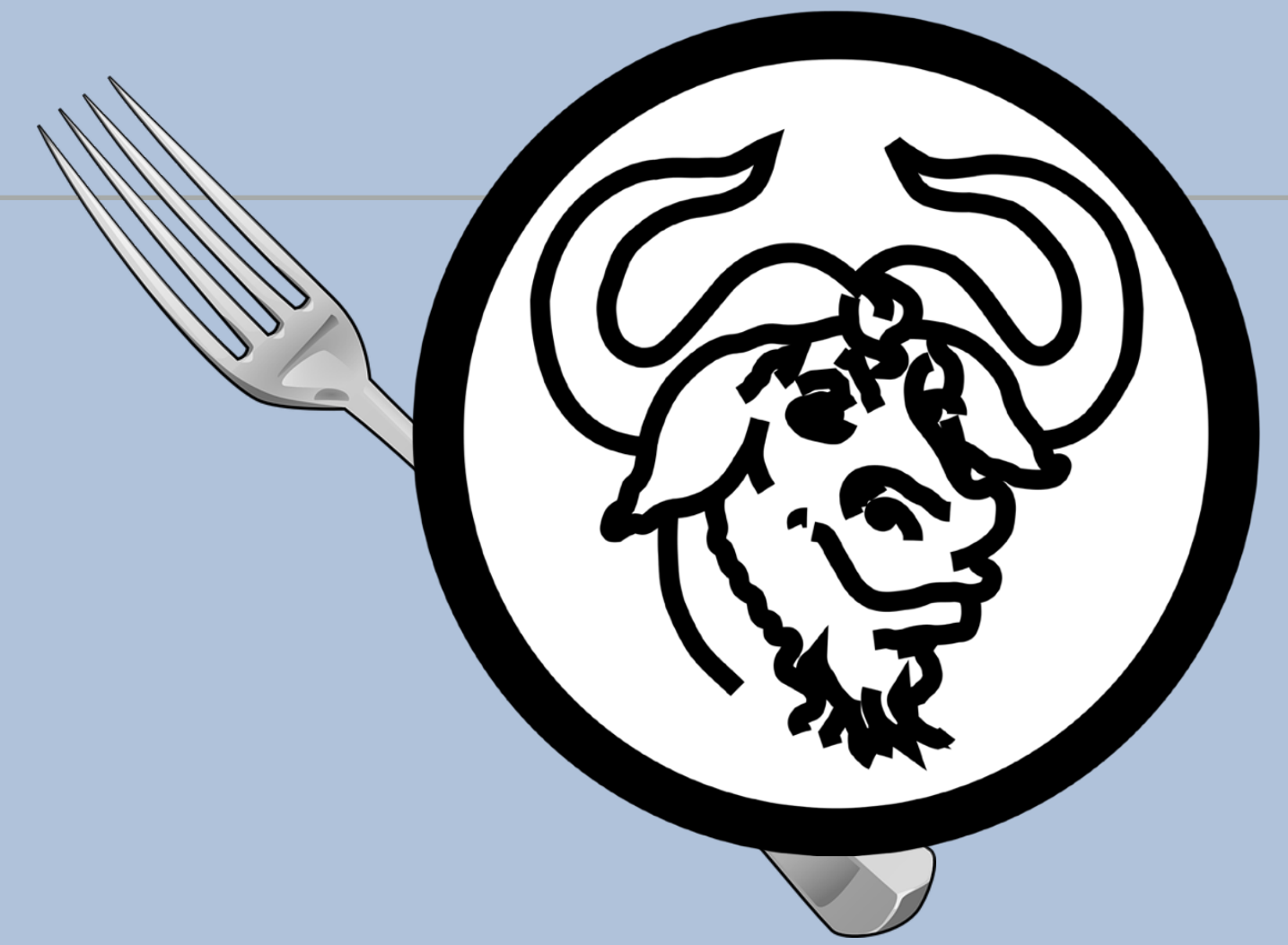
&





GNU

What is GNU?



- ▶ GNU: "GNU is not UNIX"
- ▶ Free and open source **UNIX-like** system
- ▶ Contains **no UNIX code** (and no BSD license)
- ▶ **Program Library**: Includes suite of programs that can be used alone or ***in combination*** (super powerful, more creative computing)

What is GNU?

HISTORY



- ▶ Launched in September of **1983** (early collaborative model)
- ▶ Collaborative free software with hundreds of thousands of contributors
- ▶ Initiated by “hacker” **Richard Stallman** from MIT, founder of the Free Software Foundation

What is GNU?

GNU MANIFESTO - FREE SOFTWARE FOUNDATION

1985: Stallman quits his job at MIT and writes "GNU Manifesto" outlining social and technical goals of the project.



What is GNU?



GNU MANIFESTO - FREE SOFTWARE FOUNDATION

1985: Stallman quits his job at MIT and writes "GNU Manifesto" outlining social and technical goals of the project.

"Extracting money from users of a program by restricting their use of it is destructive."



What is GNU?



GNU MANIFESTO - FREE SOFTWARE FOUNDATION

1985: Stallman quits his job at MIT and writes "GNU Manifesto" outlining social and technical goals of the project.

"Extracting money from users of a program by restricting their use of it is destructive."

"Once GNU is written, everyone will be able to obtain good system software free, just like air."



What is GNU?



GNU MANIFESTO - FREE SOFTWARE FOUNDATION

1985: Stallman quits his job at MIT and writes "GNU Manifesto" outlining social and technical goals of the project.

"Extracting money from users of a program by restricting their use of it is destructive."

"Once GNU is written, everyone will be able to obtain good system software free, just like air."

"People said I should accept the world. Bullshit. I don't accept the world."



Isn't UNIX Already Free?



Isn't UNIX Already Free?



Isn't UNIX Already Free?



GNU VS. BSD UNIX



- ▶ Both are UNIX derivatives
- ▶ Both are portable operating systems free of charge
- ▶ Different code (BSD uses UNIX code, GNU does not)

GNU VS. BSD UNIX



- ▶ Both are UNIX derivatives
- ▶ Both are portable operating systems free of charge
- ▶ Different code (BSD uses UNIX code, GNU does not)

LICENSING



BSD uses the **BSD (Berkeley Software Distribution) license**



GNU uses **GPL license** - "General Public License"

GNU VS. BSD UNIX



- ▶ Both are UNIX derivatives
- ▶ Both are portable operating systems free of charge
- ▶ Different code (BSD uses UNIX code, GNU does not)

LICENSING



BSD uses the **BSD (Berkeley Software Distribution) license**



GNU uses **GPL license** - "General Public License"

♥♥♥ **SHARE ALIKE** ♥♥♥



GNU S. BSD UNIX

are UNIX derivatives

▶ Both are available operating systems of charge

ent (BSD uses a different license)

LICENSING



BSD uses the **BSD (Berkeley Software Distribution) license**



GNU uses **GPL license** - "General Public License"

♥♥♥ **SHARE ALIKE** ♥♥♥

GNU VS. BSD UNIX

GPL LICENSE MODEL



GNU VS. BSD UNIX

GPL LICENSE MODEL



- ▶ GPL: GNU General Public License (Free Software Foundation)
 - ▶ v.1 published in 1989
 - ▶ v.3 published in 2007
- ▶ First "copyleft" license for public use

GNU VS. BSD UNIX

GPL LICENSE MODEL



- ▶ GPL: GNU General Public License (Free Software Foundation)
 - ▶ v.1 published in 1989
 - ▶ v.3 published in 2007
- ▶ First "copyleft" license for public use
- ▶ **Derivative works MUST be distributed under GPL license**

GNU VS. BSD UNIX

GPL LICENSE MODEL



- ▶ GNU Licensing adds a "share-alike" requirement
- ▶ Anyone who adds to the software and writes new code is required by law to share it openly and for free

GOOD FOR PRESERVATION

"Free as in speech, not free as in beer."

(a matter of liberty, not price)

GNU VS. BSD UNIX

BSD LICENSE



GNU VS. BSD UNIX

BSD LICENSE



- ▶ BSD: Berkeley Software Distribution (originally for BSD UNIX)
 - ▶ First published in **1988**
- ▶ Considered a "**permissive**" license
- ▶ **No "share-alike" necessary"**
- ▶ Retain the option of commercializing at any time

GNU VS. BSD UNIX

BSD LICENSE



- ▶ BSD: Berkeley Software Distribution (originally for BSD UNIX)
- ▶ First published in 1988

▶ ALSO PRETTY GOOD FOR PRESERVATION

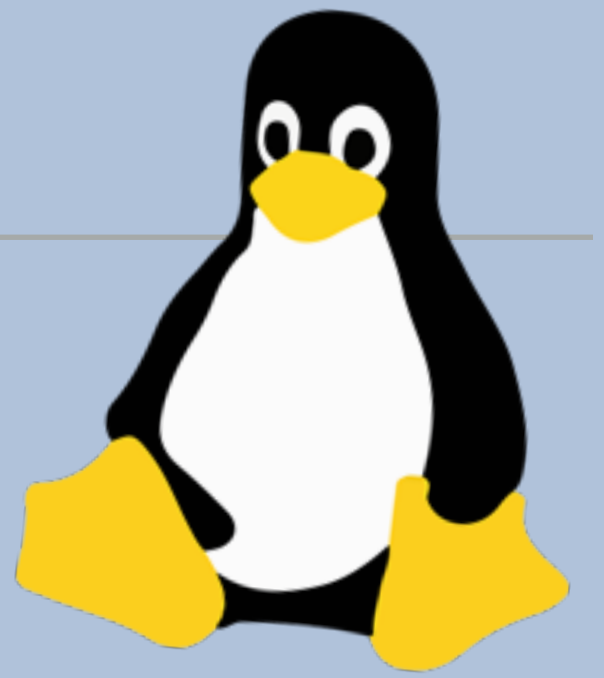
- ▶ No "**share-alike**" necessary"
- ▶ Retain the option of commercializing at any time

LINUX



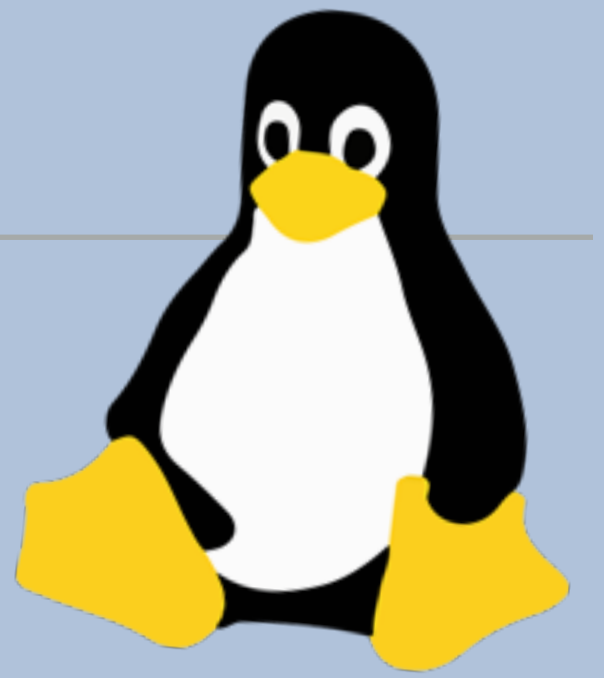
LINUX

What is Linux?



- ▶ Free and open source UNIX-like operating system released in **1991**
- ▶ Written by **Linus Torvalds**, a student at the University of Helsinki

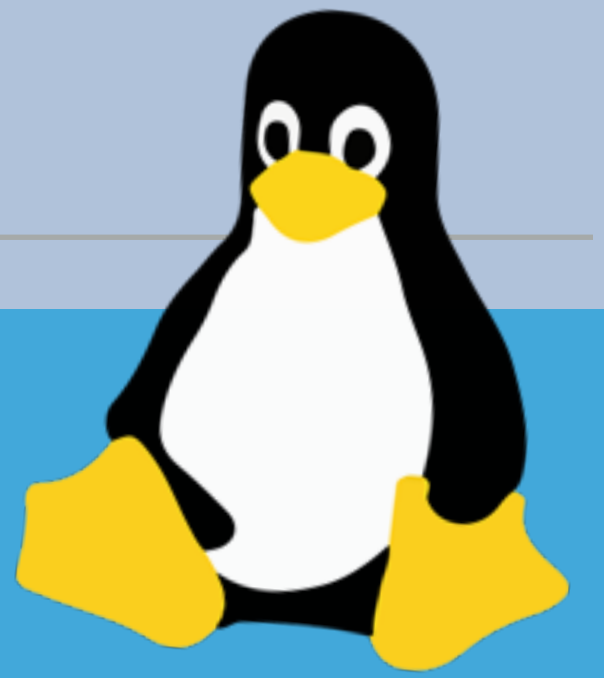
What is Linux?



- ▶ Free and open source UNIX-like operating system released in **1991**
- ▶ Written by **Linus Torvalds**, a student at the University of Helsinki



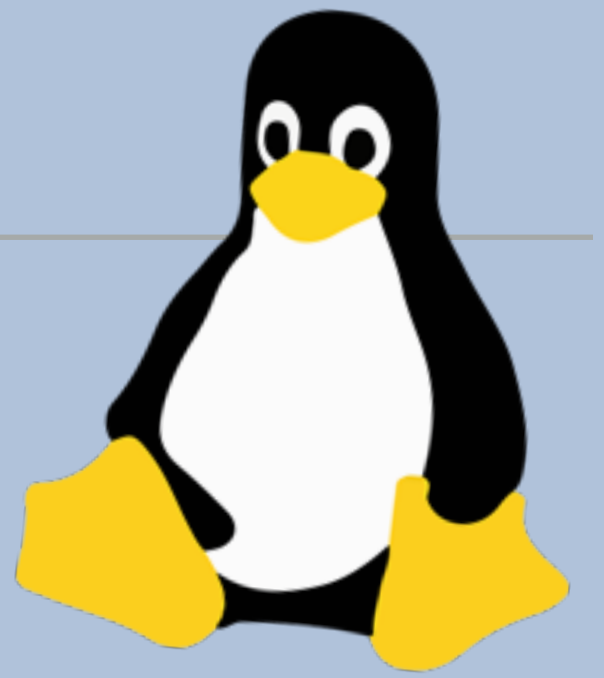
What is Linux?



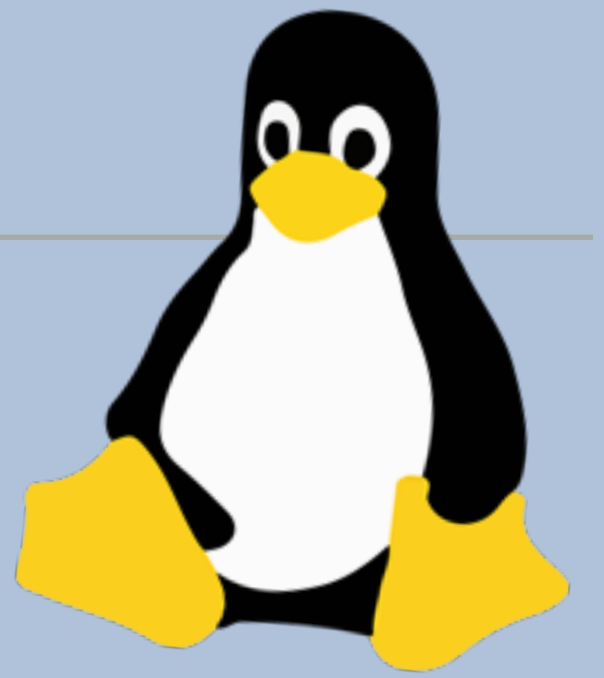
**GOAL: CREATE AN OPEN SOURCE, FREELY AVAILABLE UNIX-
LIKE OPERATING SYSTEM FOR PERSONAL COMPUTERS**



Didn't that already exist?

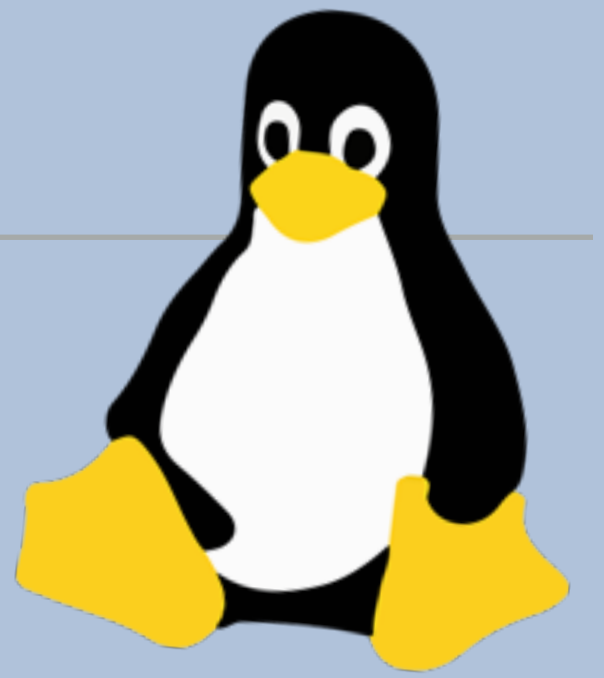


Didn't that already exist?



... almost but not quite.

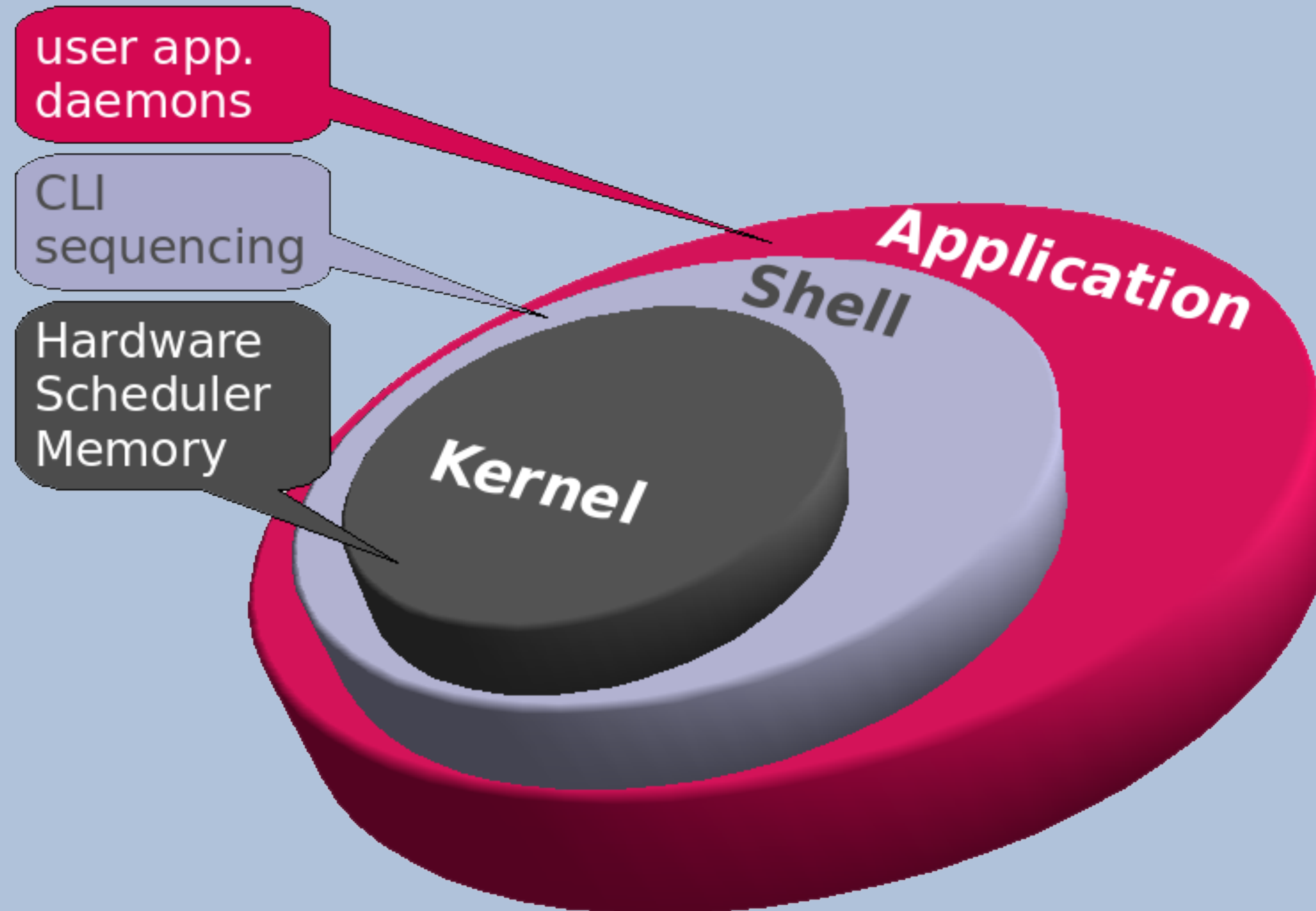
Didn't that already exist?



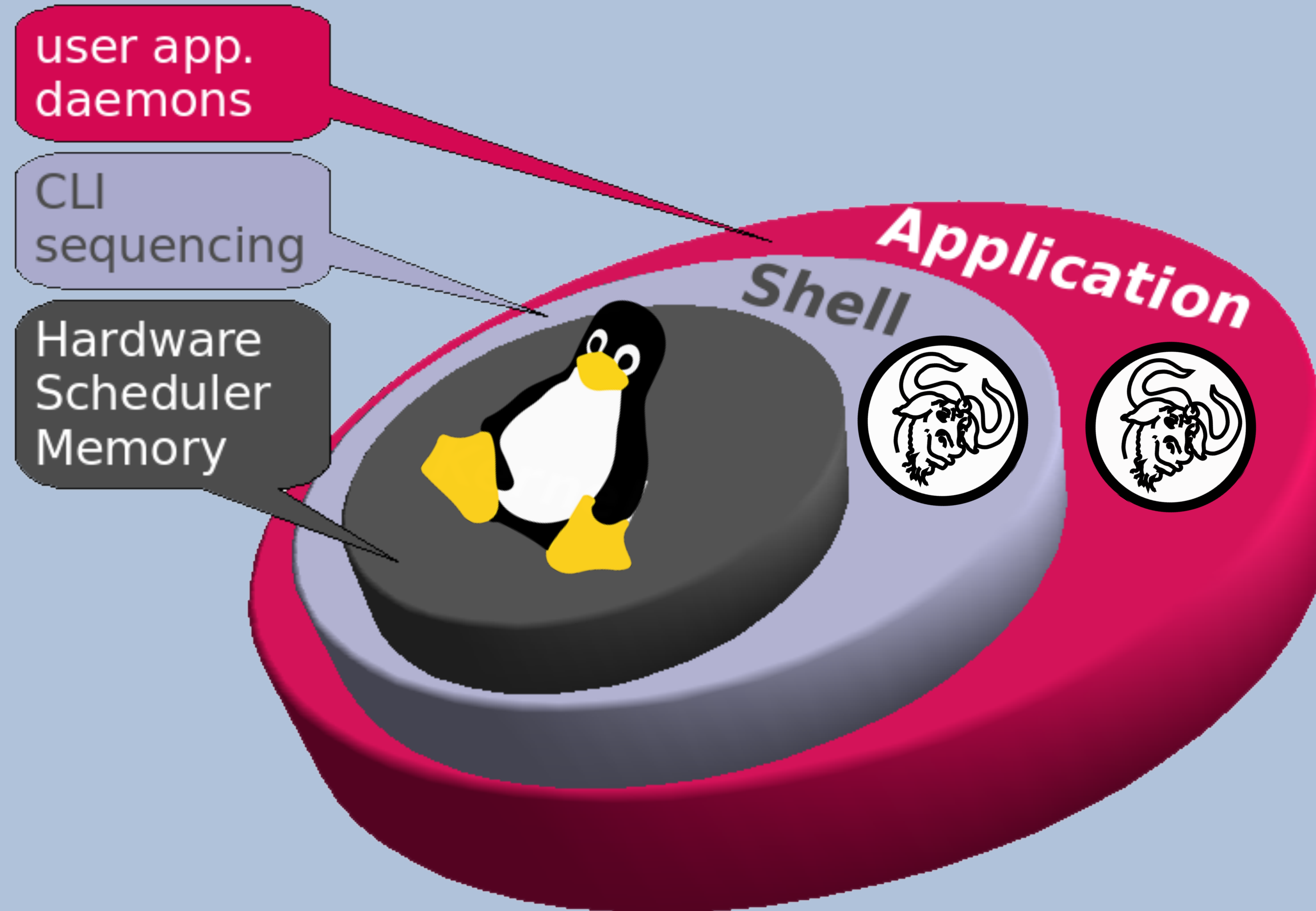
... almost but not quite.

- ▶ GNU's programs were ready by 1991
- ▶ GNU lacked a "**kernel**", or the core component around which the operating system is based
- ▶ **Kernel considered to be the last missing piece of GNU**

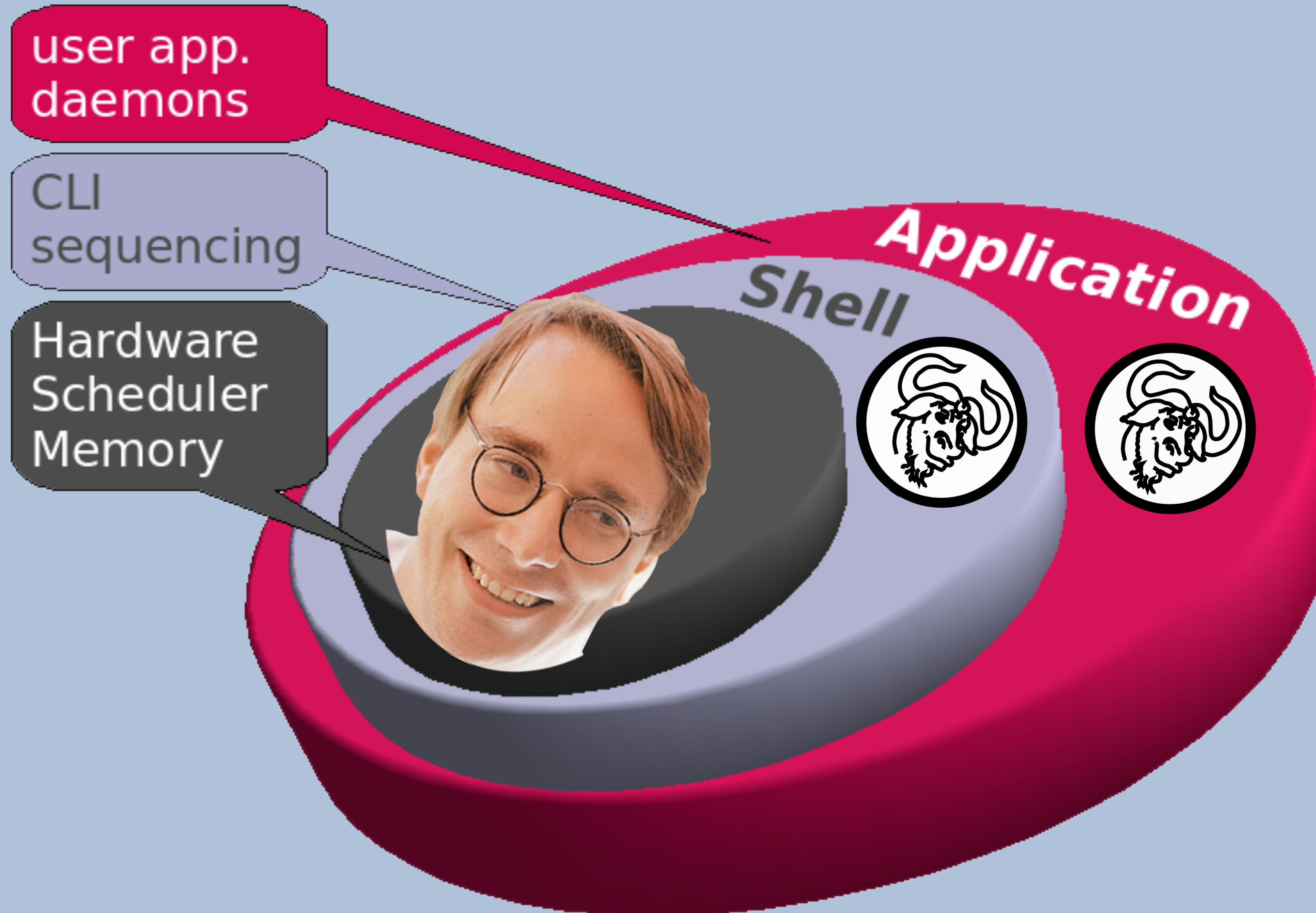
Linux Kernel - Core of Operating System



Linux Kernel - Core of Operating System



Linux Kernel - Core of Operating System



GNU/LINUX

- ▶ 1991: Torvalds creates a kernel for the GNU operating system
- ▶ Names the new kernel "**LINUX**"
- ▶ Introduces a completed GNU/LINUX operating system



GNU/LINUX

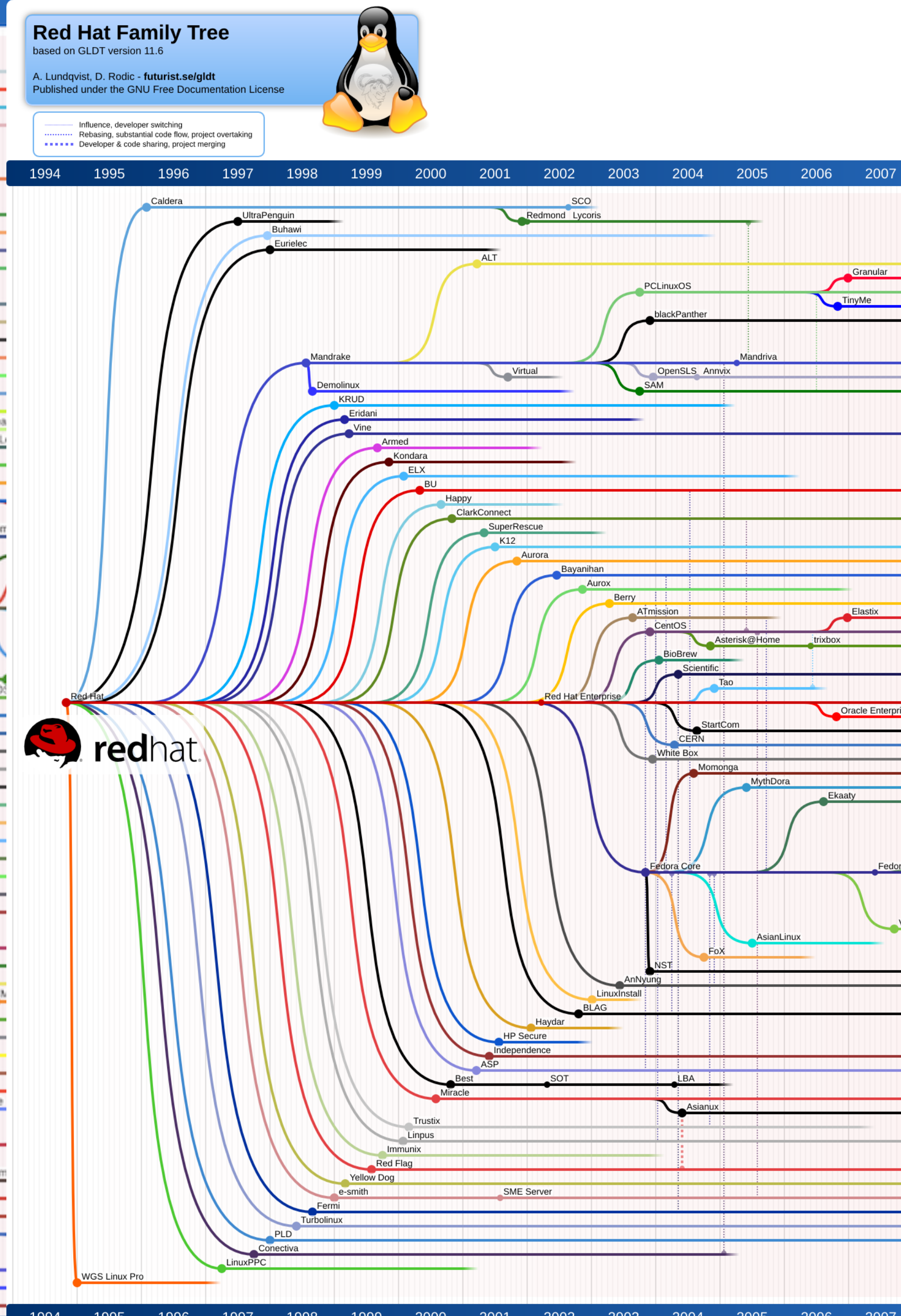
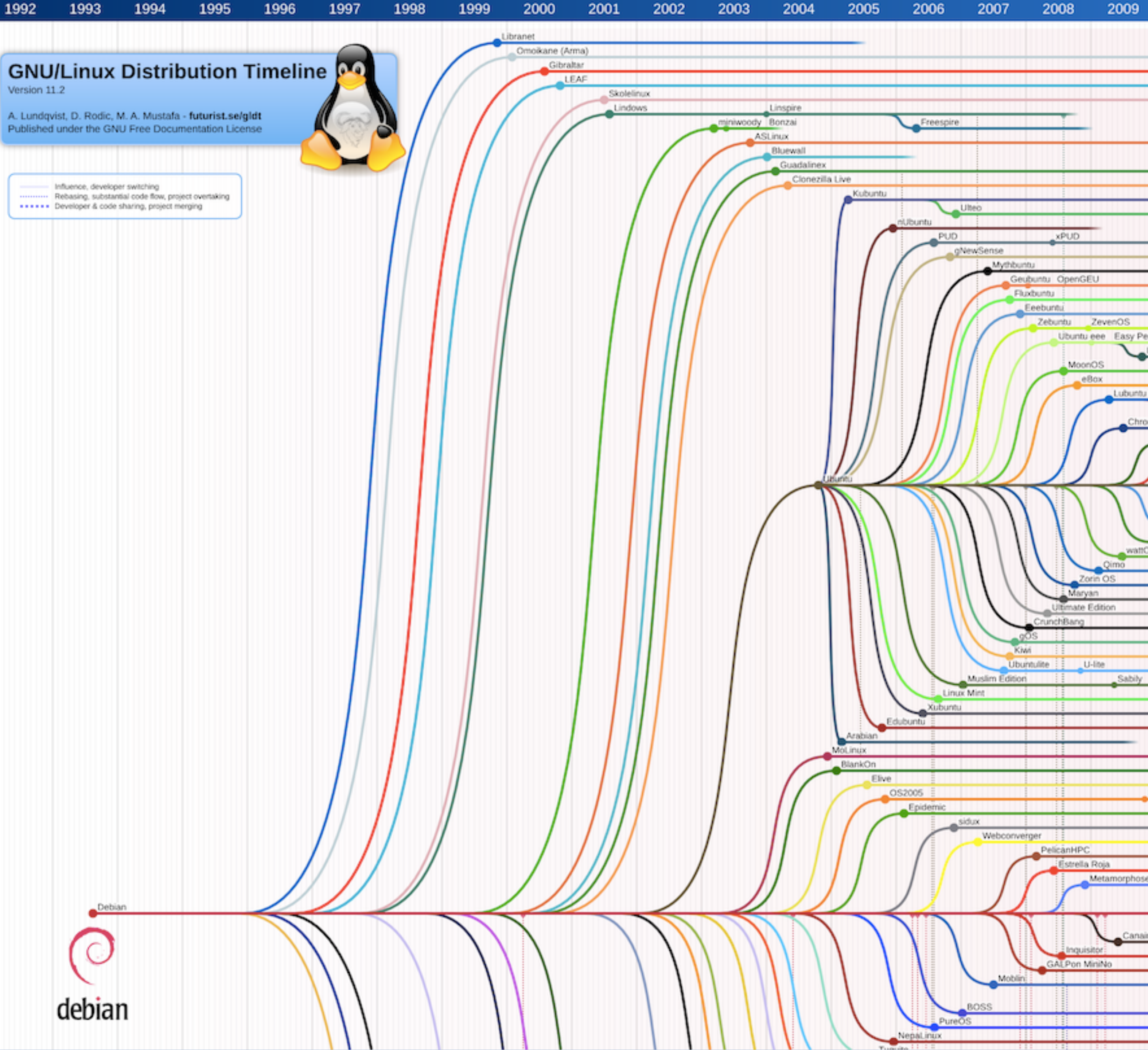
- ▶ 1991: Torvalds creates a kernel for the GNU operating system
- ▶ Names the new kernel "**LINUX**"
- ▶ Introduces a completed GNU/LINUX operating system
- ▶ **First freely available, UNIX-like OS (with no UNIX code)**



GNU/LINUX

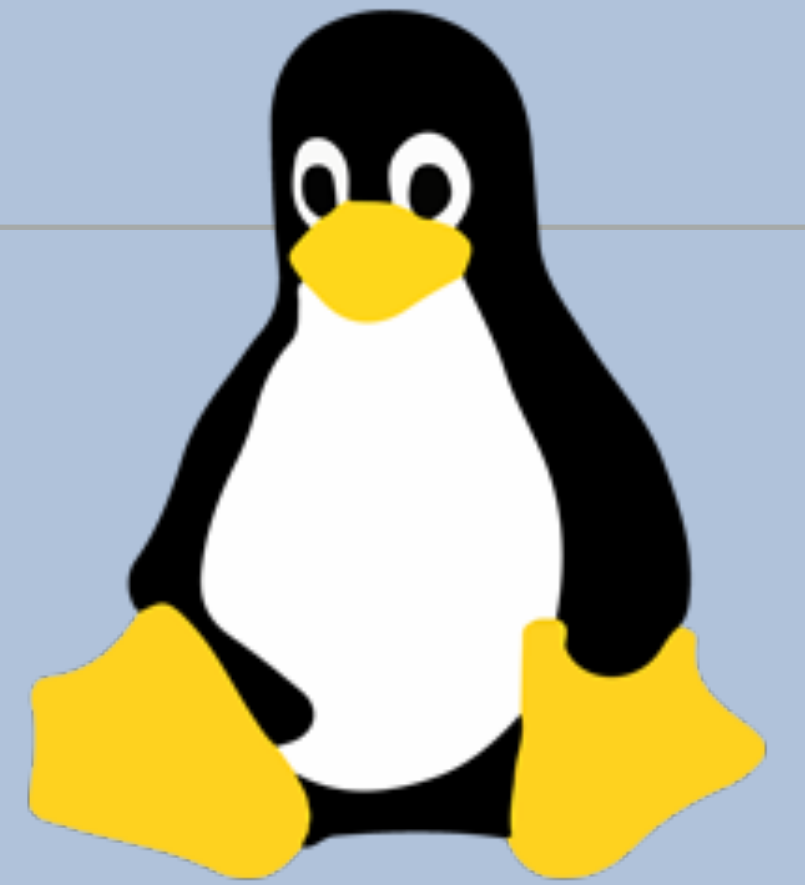


- ▶ 1991: Torvalds creates a kernel for the GNU operating system
- ▶ Names the new kernel "**LINUX**"
- ▶ Introduces a completed GNU/LINUX operating system
- ▶ **First freely available, UNIX-like OS (with no UNIX code)**
- ▶ General Public License (Free Software Foundation)
 - ▶ Linux proliferates like crazy
 - ▶ Geeks proceed to argue about what to call it (Linux or GNU/Linux)



GNU/LINUX

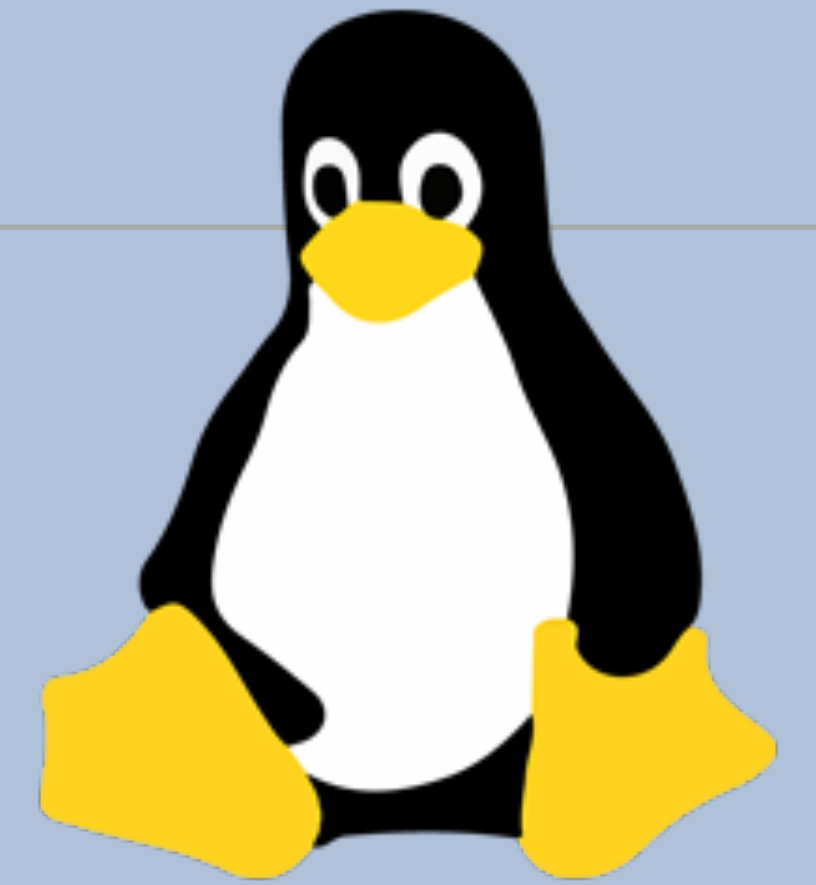
DERIVATIVE OPERATING SYSTEMS



GNU/LINUX

DERIVATIVE OPERATING SYSTEMS

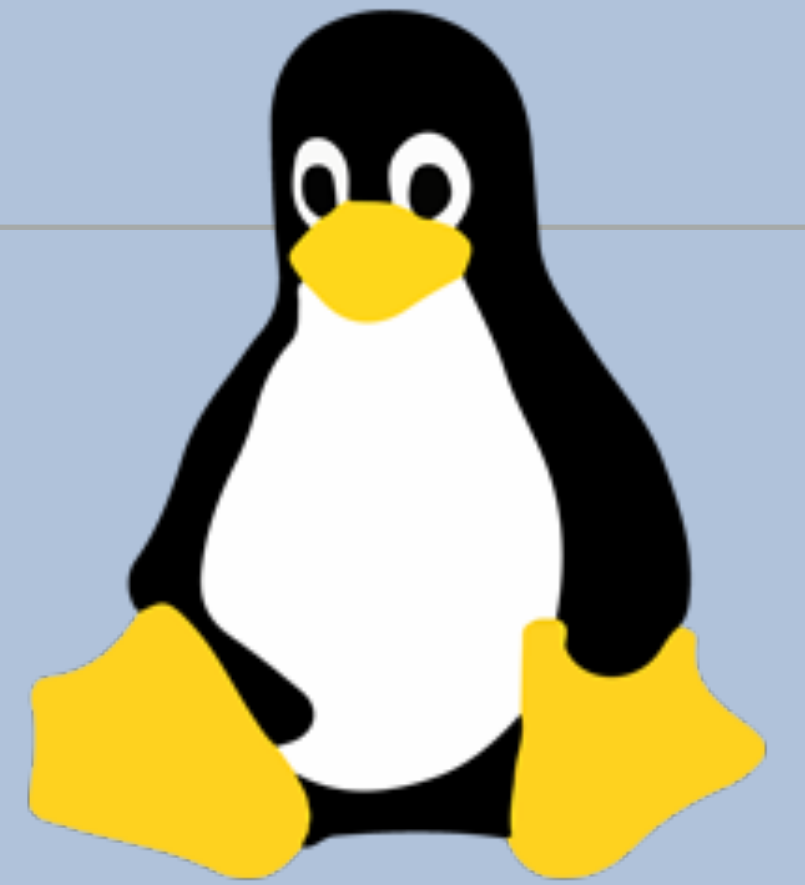
- ▶ Android



GNU/LINUX

DERIVATIVE OPERATING SYSTEMS

- ▶ Android
- ▶ Google Chrome OS



GNU/LINUX

DERIVATIVE OPERATING SYSTEMS

- ▶ Android
- ▶ Google Chrome OS
- ▶ Debian > **Ubuntu**



GNU/LINUX

DERIVATIVE OPERATING SYSTEMS

- ▶ Android
- ▶ Google Chrome OS
- ▶ Debian > **Ubuntu**, Mint



GNU/LINUX

DERIVATIVE OPERATING SYSTEMS

- ▶ Android
- ▶ Google Chrome OS
- ▶ Debian > **Ubuntu**, **Mint**
- ▶ **Red Hat**



GNU/LINUX

DERIVATIVE OPERATING SYSTEMS

- ▶ Android
- ▶ Google Chrome OS
- ▶ Debian > **Ubuntu**, **Mint**
- ▶ **Red Hat**
- ▶ SteamOS



GNU/LINUX

DERIVATIVE OPERATING SYSTEMS **FOR DIGITAL PRESERVATION**

GNU/LINUX

DERIVATIVE OPERATING SYSTEMS **FOR DIGITAL PRESERVATION**

BitCurator



deft

PRESERVATION: PROPRIETARY VS. FREE SOFTWARE

PRESERVATION: PROPRIETARY VS. FREE SOFTWARE

BENEFITS OF PROPRIETARY SOFTWARE

PRESERVATION: PROPRIETARY VS. FREE SOFTWARE

BENEFITS OF PROPRIETARY SOFTWARE

1. **“Turn-Key” Solution** – Software works out of the box
2. **Support** – Vendor provides support
3. **Business Model** – Sustainable capitalist business model

PRESERVATION: PROPRIETARY VS. FREE SOFTWARE

ISSUES WITH PROPRIETARY SOFTWARE

PRESERVATION: PROPRIETARY VS. FREE SOFTWARE

ISSUES WITH PROPRIETARY SOFTWARE

1. **Lock-in** – The vendor might...
 - ... change price of service, upgrades, etc.
 - ... go out of business
 - ... discontinue product
2. **Dependence** – reliance on vendor for technical support
3. **Cost** – can be very expensive
4. **Obsolescence** – product may fall out of favor
5. **Transparency** – code is not accessible
6. **Validation/Auditing** – no access to code for quality assurance

PRESERVATION: PROPRIETARY VS. FREE SOFTWARE

ISSUES WITH FREE SOFTWARE

PRESERVATION: PROPRIETARY VS. FREE SOFTWARE

ISSUES WITH FREE SOFTWARE

1. **Business Model** – No obvious sustainable business model
2. **Modular Design** – Systems do not work together by default
3. **No support** – User must customize/manage software and systems

PRESERVATION: PROPRIETARY VS. FREE SOFTWARE

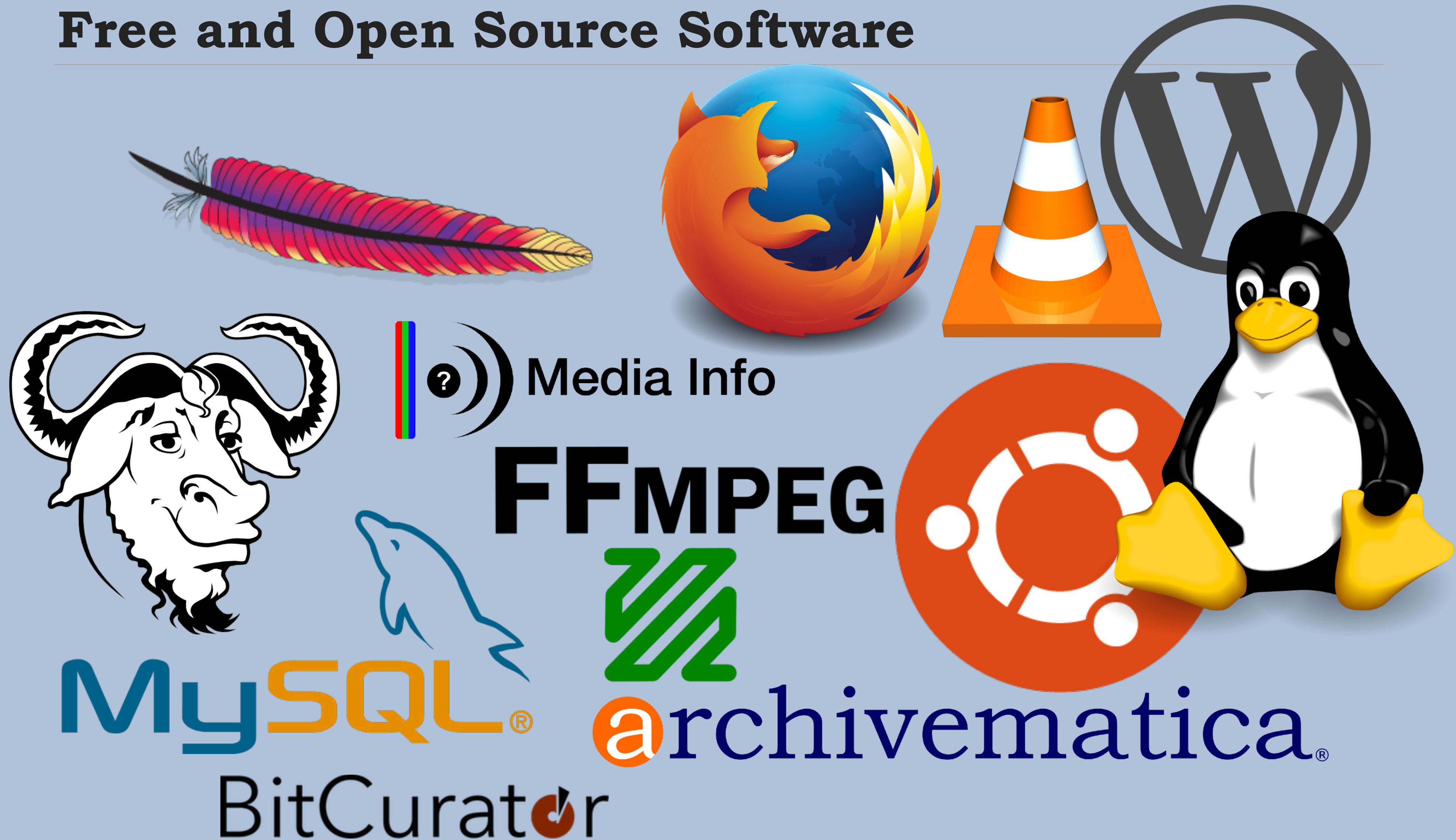
BENEFITS OF FREE SOFTWARE

PRESERVATION: PROPRIETARY VS. FREE SOFTWARE

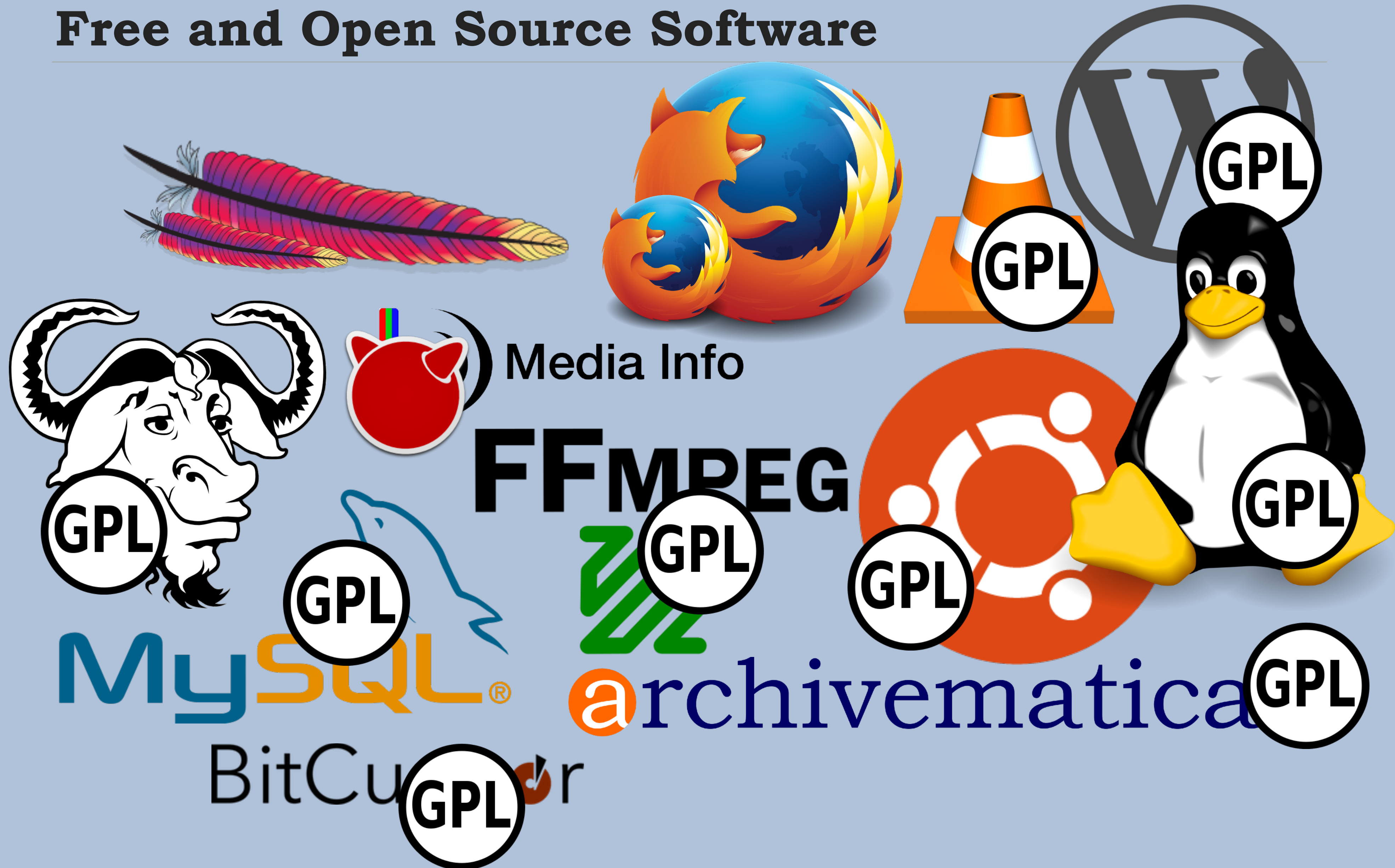
BENEFITS OF FREE SOFTWARE

1. **Community** – Collaboration brings partners together, share work
2. **Access** – Code is available and can be customized
3. **Modular Design** – Systems can be swapped out/reconfigured
4. **Auditing** – Easier to evaluate systems when code is open
5. **Sustainable** – Code will always be available, even after obsolescence
6. **Stock Value** – Wide user adoption results in **high stock value**

Free and Open Source Software



Free and Open Source Software



THE

END.