Carnegie Mellon University

Introduction to Software Curation and Preservation

Eric Kaltman

CLIR Fellow for Data Curation in the Sciences

Overview of Workshop

1. Why Software Preservation?
2. What is Software Curation?
    a. Scope and Description
    b. Migration
    c. Reproduction and Access
    d. Outreach and Culture
3. Trends
4. Efforts at CMU

Carnegie Mellon University

Overview of Workshop

1. Why Software Preservation?
2. What is Software Curation?
   a. Scope and Description
   b. Migration
   c. Reproduction and Access
   d. Outreach and Culture
3. Trends
4. Efforts at CMU

Carnegie Mellon University

What is Software?

Source Code

- Instructions to a computational device to do something

Executable binaries

- Compiled source code that executes on in a target environment

Documentation

- Descriptions of implementation and use

What is Software?

Data encoded to be executed by a specific computational context (‘execution context’)

Execution Context - the collection of hardware and software dependencies required to allow for execution

Hardware Dependencies - CPUs, system architecture, peripherals, displays, etc.

Software Dependencies - Compiled libraries, support programs, APIs, etc.

Why Software Preservation?

History

- Modern society is built on and from interactions with software

Progress

- Modern research, regardless of the field, is tied to software infrastructures

Pedagogy

- Access to legacy software and systems for teaching and critical engagement.

Practicality

- Software at the root of commercial progress, ties to competition and legal issues

History and Reproducibility

Research reproducibility

- Maintaining citable access to previous results, analyzes and data sets
- Shifting research practices toward an archival and curatorial mindset
- Models for sustainable software development

History, legacy and maintenance

- Preserving the historical record, the intellectual history of humanity
- Aligning with a maintenance narrative instead of an innovation one
- Access to the past will be dependent on software for the rest of time

Access to software is also access to the files produced and interpreted by software

Digital Dark Ages

"People think that bits are somehow immortal because somehow they're this ethereal thing in cyberspace...It could be that the format of those bits, the way in which they are interpreted requires a piece of software to figure out what the bits mean. How they should be presented as an image or a video or how you should interact with it in a spreadsheet, but the software doesn't exist. What if the operating system that the old software used to run on doesn't exist anymore? What if the latest software doesn't know how to read the formats of those complex digital objects? Guess what? That information is gone.

There isn't a systematic way to ensure the information that we create today will still be usable 100 years from now. That's why I'm worried about a ***digital dark ages***.

Vint Cerf, co-inventor of TCP/IP

Carnegie Mellon University

Ensuring the Longevity of Digital Information

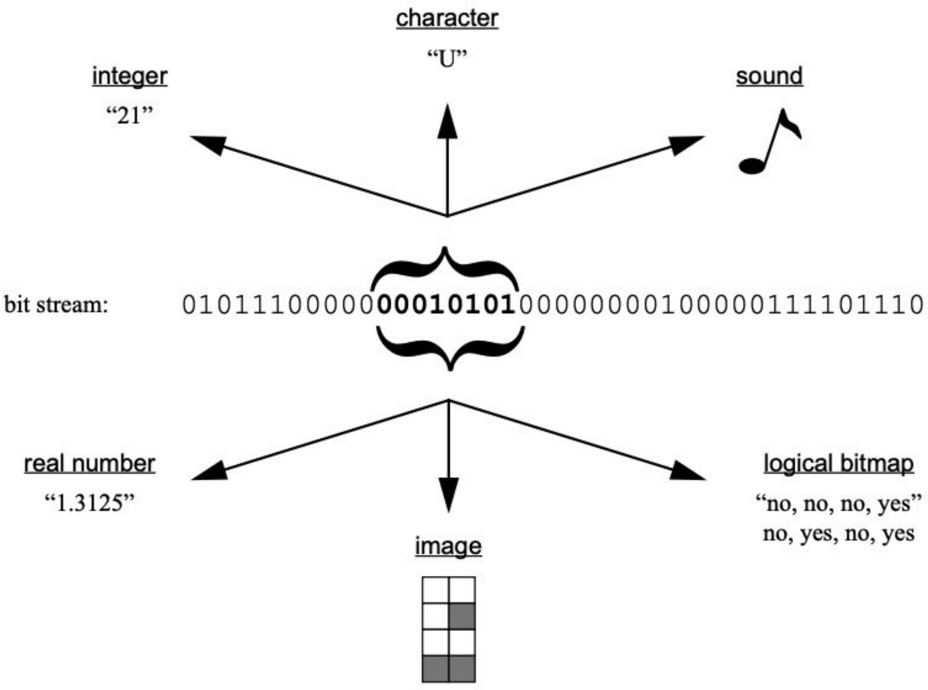


Figure 4: A bit stream can represent anything at all

Jeff Rothenberg
Scientific American
1995

Carnegie Mellon University

Overview of Workshop

1. Why Software Preservation?
2. What is Software Curation?
    a. Scope and Description
    b. Migration
    c. Reproduction and Access
    d. Outreach and Culture
3. Trends
4. Efforts at CMU

Carnegie Mellon University

Curation

Scope

- What is historically important?
- What will be important for future research efforts?

Description and Standards

- How do we describe software for future access, study and use?

Migration

- How do we transition and preserve data across time?

Storage and Access

- Where do we put these things?
- How do we find them again?

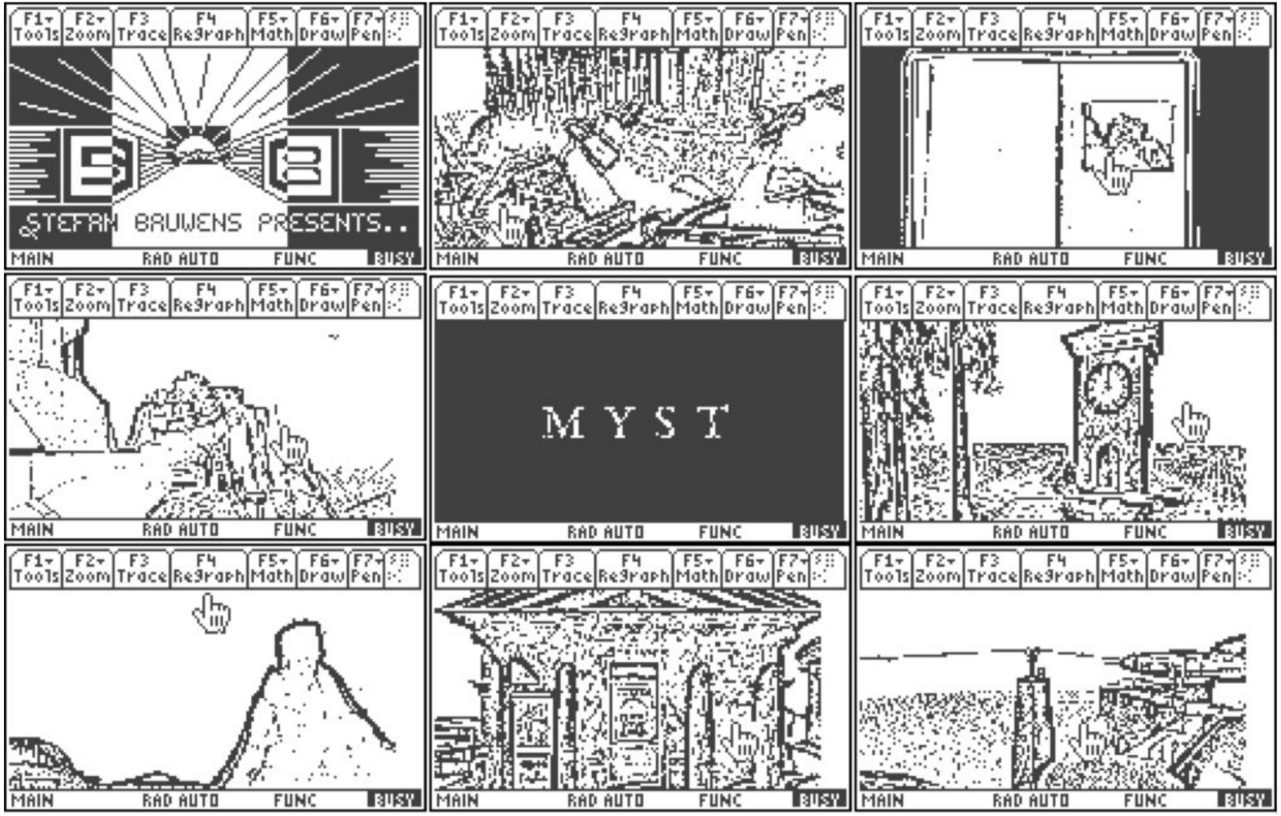
Carnegie Mellon University

What is Important?

A medical supply company in Miami had received a delivery of botulin, which was to be processed into Botox and distributed. However, it was misprocessed, and a dangerous concentrate was distributed. The FDA had all of the information needed to identify the recipients, but the information was in a file created with a 2003 version of a popular business software application. The 2004 version available to the FDA could not open the data file. The manufacturer of the software was also unable to supply the relevant version.

It so happened that one of the agents involved in the case was familiar with the NSRL, and had in fact provided software to us earlier in the year. He called, explained the situation, and asked if we had the 2003 version of the software. We did! The agent then arranged for an FDA contact to come to NIST, get the software, and put it on a jet to Miami. The people working the case in Miami were able to install the old version, open the data file, and trace the paths of the botulin.

Preserving.Exe: Toward a National Strategy for Software Preservation - 2013

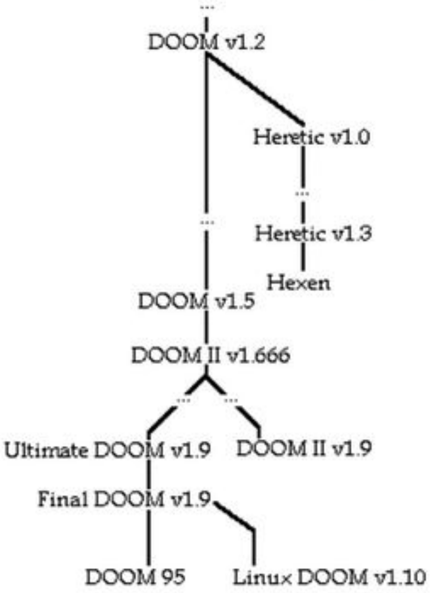


Carnegie Mellon University

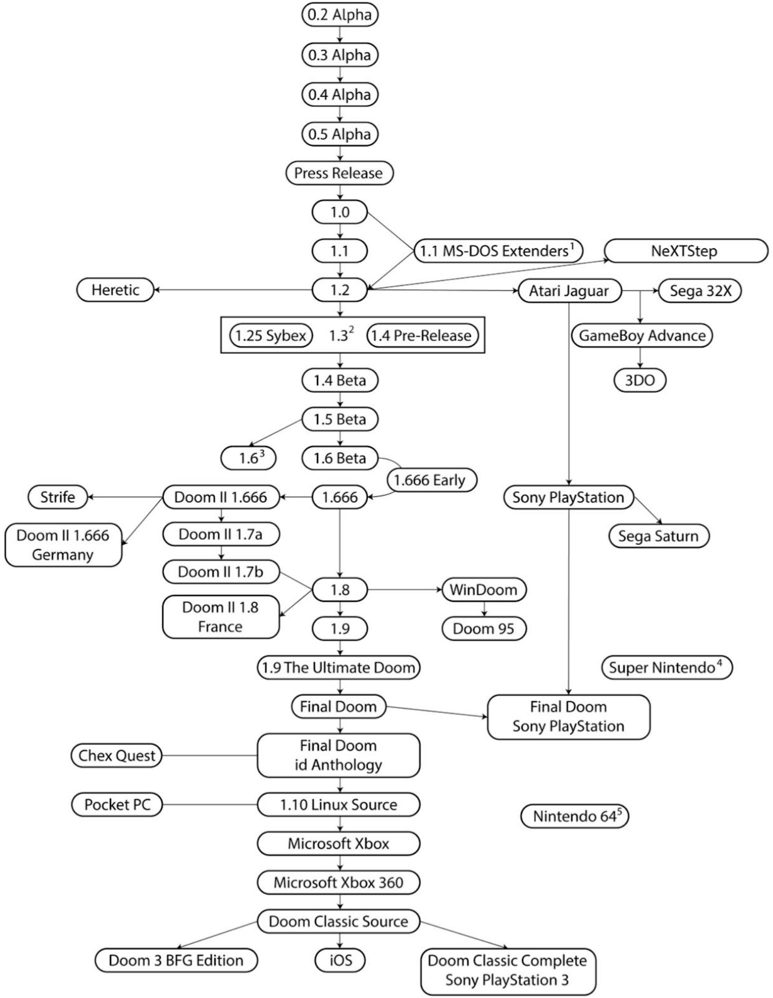

MARIO x 6
COINS:32
0027800

DOOM 89
lives: 005
health: 100
bullets: 100
shells: 020
LEVEL
002

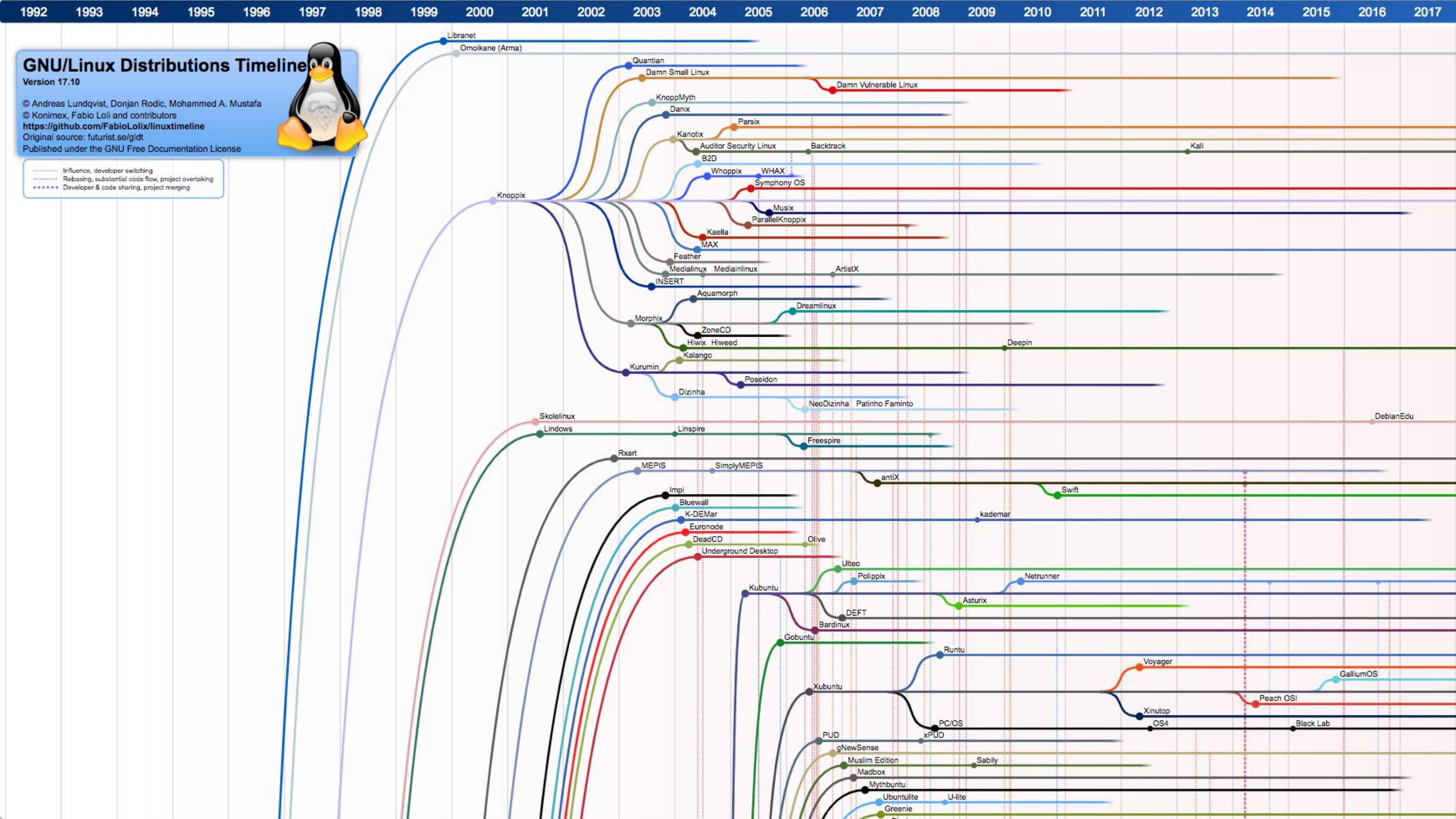
Carnegie
Mellon
University



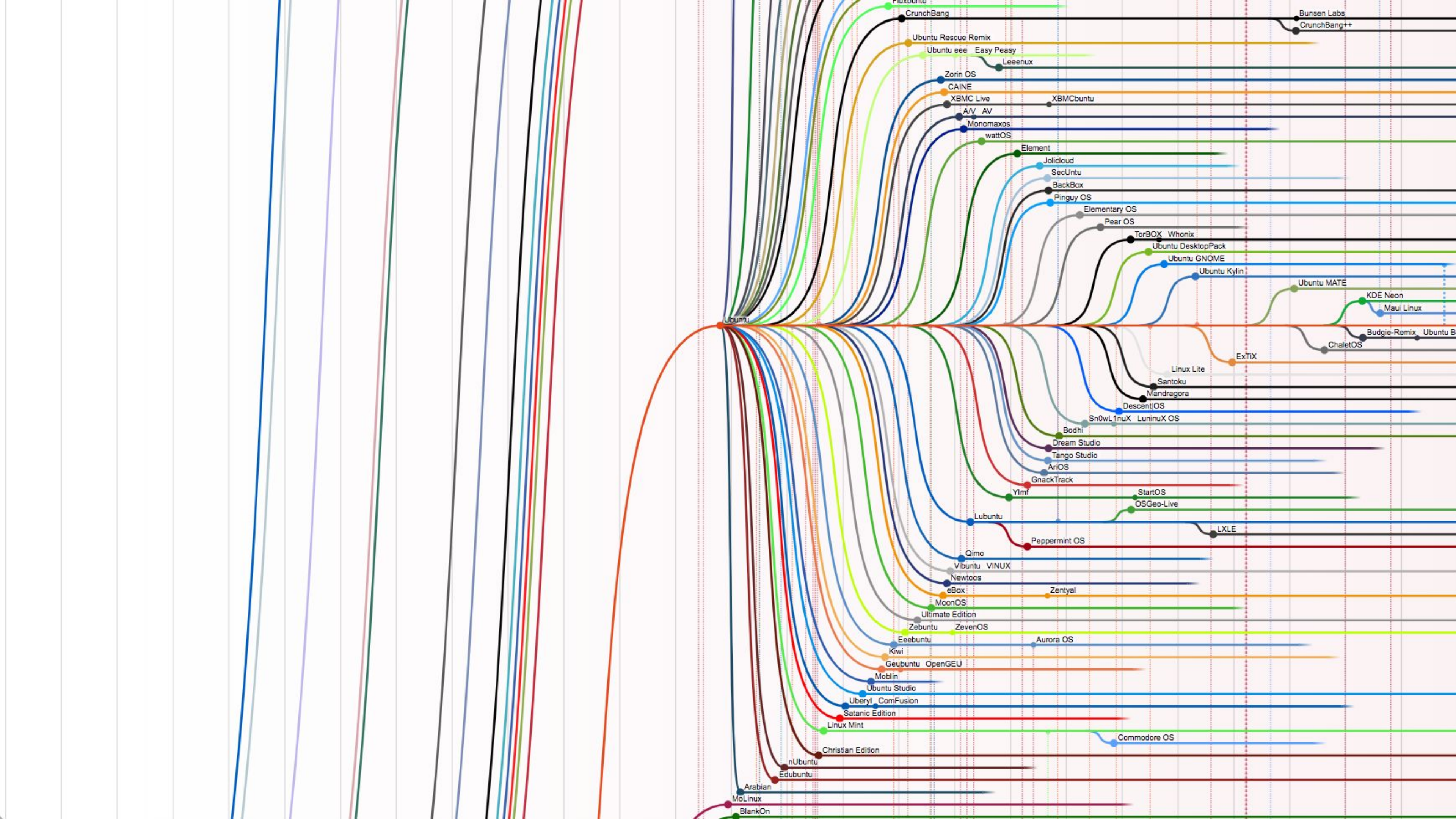
Filename	File Size	Release Date	notes
doom0_2.zip	254k	Feb. 4, 1993	alpha version - doom0_2.zip downloaded by HEL from www.doomworld.com/pageofdoom, 18 July 2008
doom0_4.zip	950k	Apr. 2, 1993	alpha version
doom0_5.zip	1264k	May 22, 1993	alpha version
doom_pre_beta.zip	2555k	Oct. 4, 1993	press-release beta version
doom1_0.zip	2113k	Dec. 10, 1993	v0.99 shareware version
doom1_1.zip	2160k	Dec 16, 1993	v1.1 shareware version
doom1_2.zip	2203k	Feb 17, 1994	shareware version
doom14bt.zip	2246k	Jun. 28, 1994	v1.4beta shareware version
doom15bt.zip	2262k	Jul. 8, 1994	v1.5beta shareware version
doom16bt.zip	2234k	Aug. 3, 1994	v1.6beta shareware version
dm1666sw.zip	2293k	Sep. 1, 1994	v1.666 shareware
doom_v18.zip	2423k	Jan 23, 1995	v1.8 shareware
doom19s.zip	2393k	N/A	v1.9 shareware version



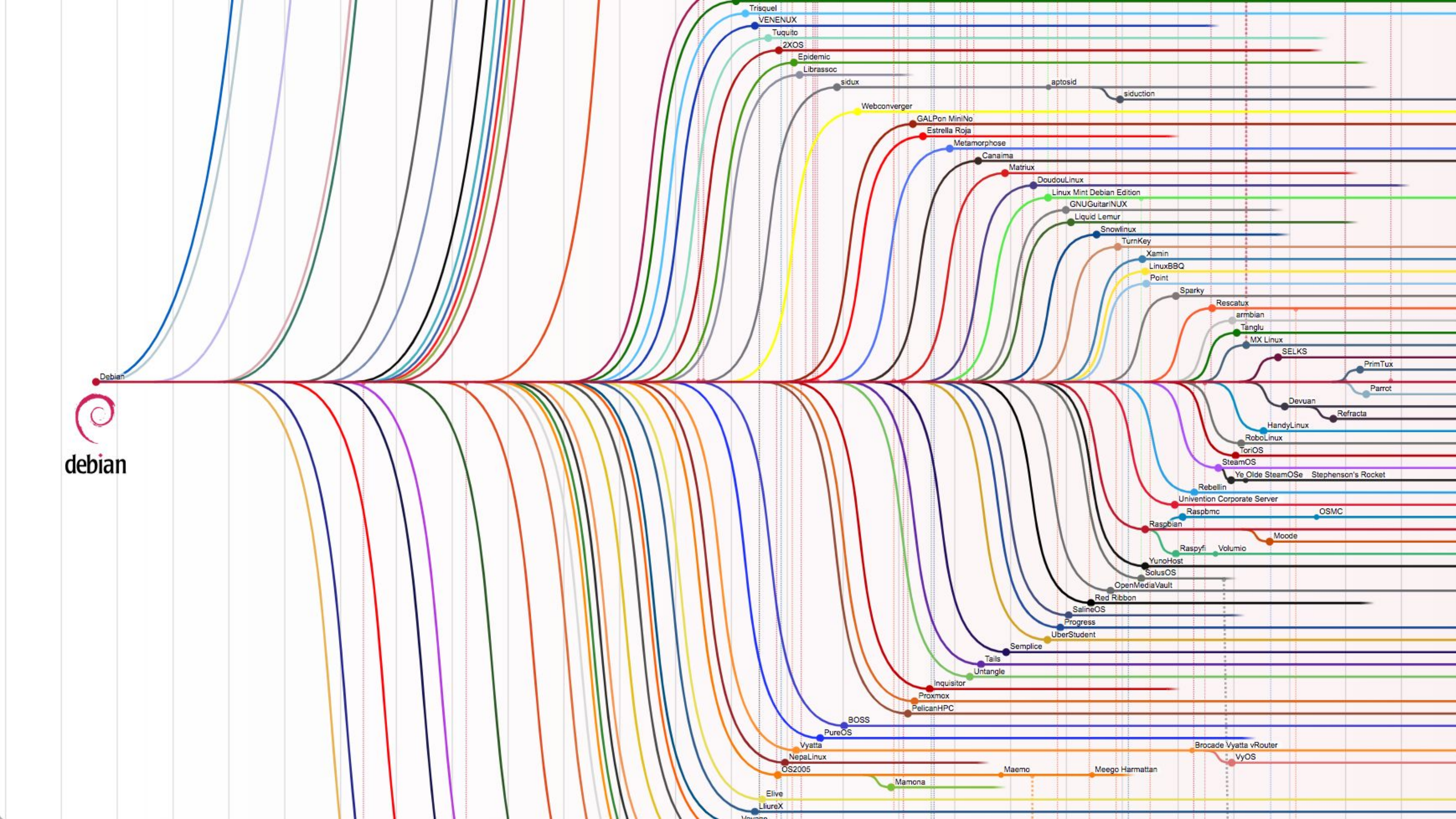
Carnegie Mellon University



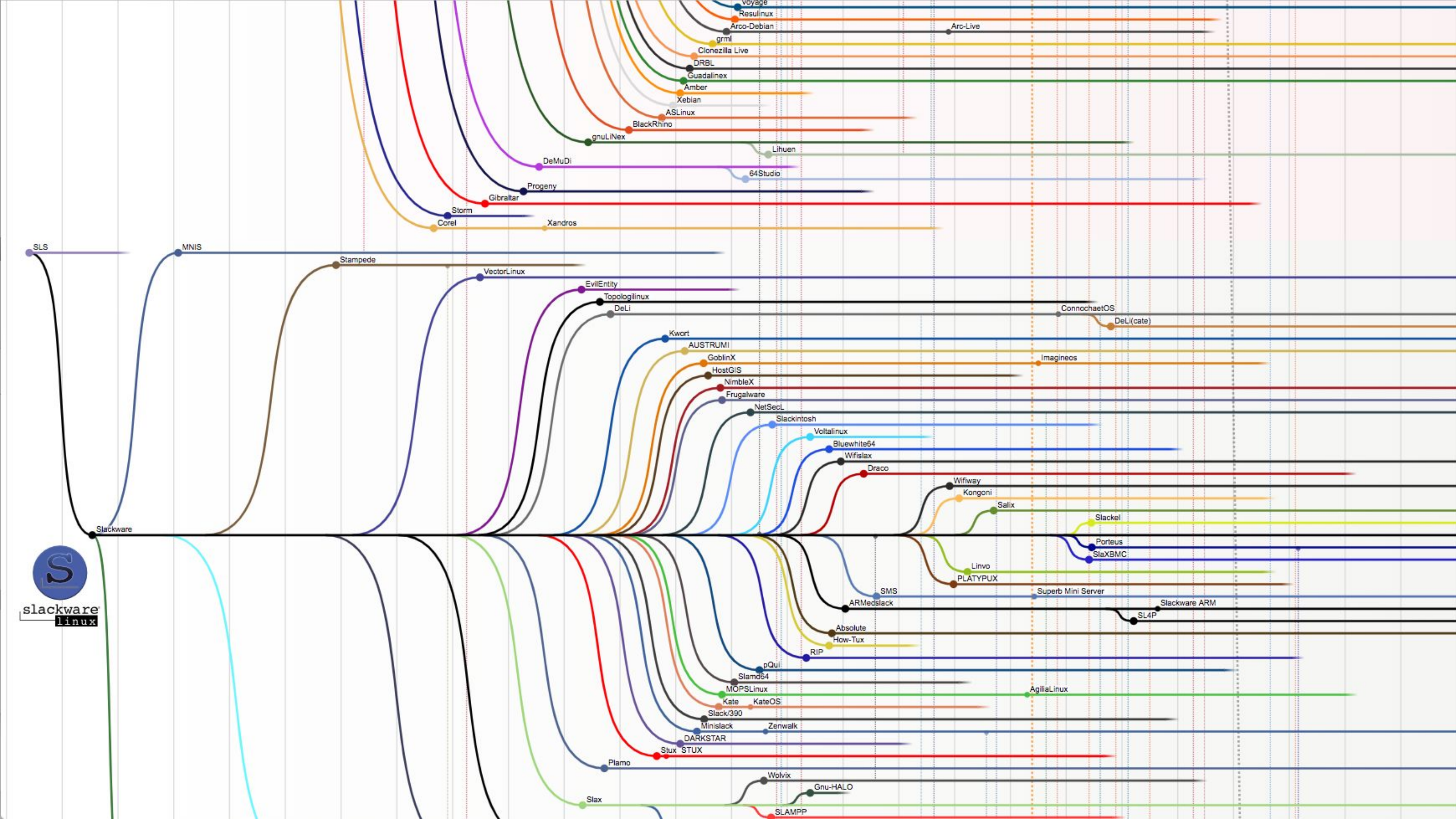
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
GNU/Linux Distributions Timeline
Version 17.10
© Andreas Lundqvist, Donjan Rodic, Mohammed A. Mustafa
© Konimex, Fabio Loli and contributors
https://github.com/FabioLolix/linuxtimeline
Original source: futurist.se/gldt
Published under the GNU Free Documentation License
Influence, developer switching
Rebasing, substantial code flow, project overtaking
Developer & code sharing, project merging
Libranet
Omoikane (Arma)
Quantian
Damn Small Linux
Damn Vulnerable Linux
KnoppMyth
Danix
Parsix
Kanotix
Auditor Security Linux
Backtrack
Kali
B2D
Whoppix
WHAX
Symphony OS
Knoppix
Musix
ParallelKnoppix
Kaella
MAX
Feather
Medialinux
Medialinux
ArtistX
INSERT
Aquamorph
Dreamlinux
Morphix
ZoneCD
Hiwix
Hiweed
Deepin
Kalango
Kurumin
Poseidon
Dizinha
NeoDizinha
Patinho Faminto
Skolelinux
DebianEdu
Lindows
Linspire
Freespire
Rxart
MEPIS
SimplyMEPIS
antiX
Swift
Impi
Bluewall
K-DEMar
kademar
Euronode
DeadCD
Olive
Underground Desktop
Ulteo
Polippix
Netrunner
Kubuntu
Asturix
DEFT
Bardinux
Gobuntu
Runtu
Voyager
GalliumOS
Xubuntu
Peach OSI
Xinutop
PC/OS
OS4
Black Lab
xPUD
PUD
gNewSense
Muslim Edition
Sabily
Madbox
Mythbuntu
Ubuntulite
U-lite
Greenie



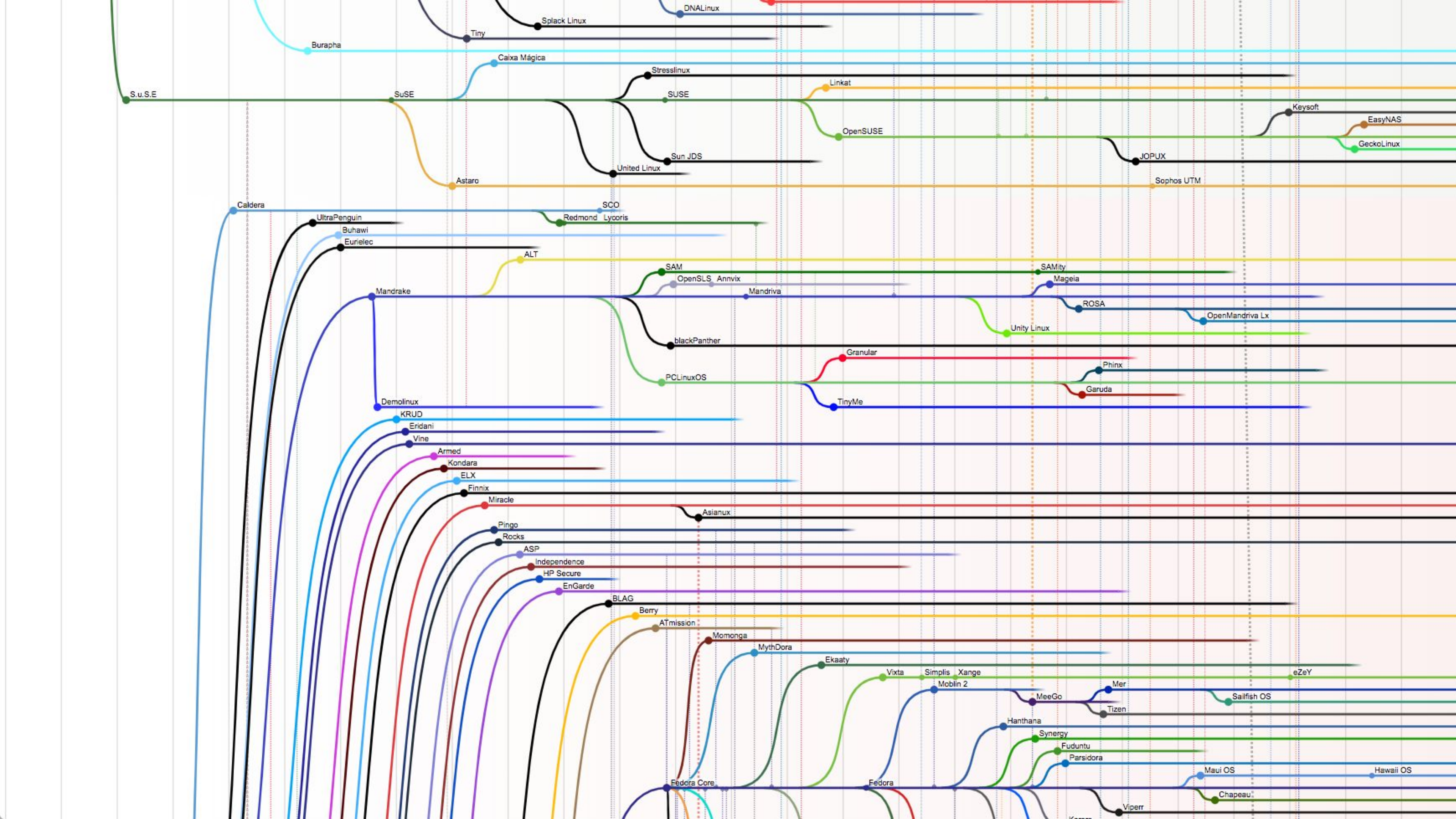
CrunchBang
Bunsen Labs
CrunchBang++
Ubuntu Rescue Remix
Ubuntu eee
Easy Peasy
Leeenux
Zorin OS
CAINE
XBMC Live
XBMCbuntu
A/V
AV
Monomaxos
wattOS
Element
Jolicloud
SecUntu
BackBox
Pinguy OS
Elementary OS
Pear OS
TorBOX
Whonix
Ubuntu DesktopPack
Ubuntu GNOME
Ubuntu Kylin
Ubuntu MATE
KDE Neon
Maui Linux
Ubuntu
Budgie-Remix
ChaletOS
ExTiX
Linux Lite
Santoku
Mandragora
DescentOS
Sn0wL1nuX
LuninuX OS
Bodhi
Dream Studio
Tango Studio
AriOS
GnackTrack
Ylmf
StartOS
OSGeo-Live
Lubuntu
LXLE
Peppermint OS
Qimo
Vibuntu
VINUX
Newtoos
eBox
Zentyal
MoonOS
Ultimate Edition
Zebuntu
ZevenOS
Eeebuntu
Aurora OS
Kiwi
Geubuntu
OpenGEU
Moblin
Ubuntu Studio
Uberyl
ComFusion
Satanic Edition
Linux Mint
Commodore OS
Christian Edition
nUbuntu
Edubuntu
Arabian
MoLinux
BlankOn



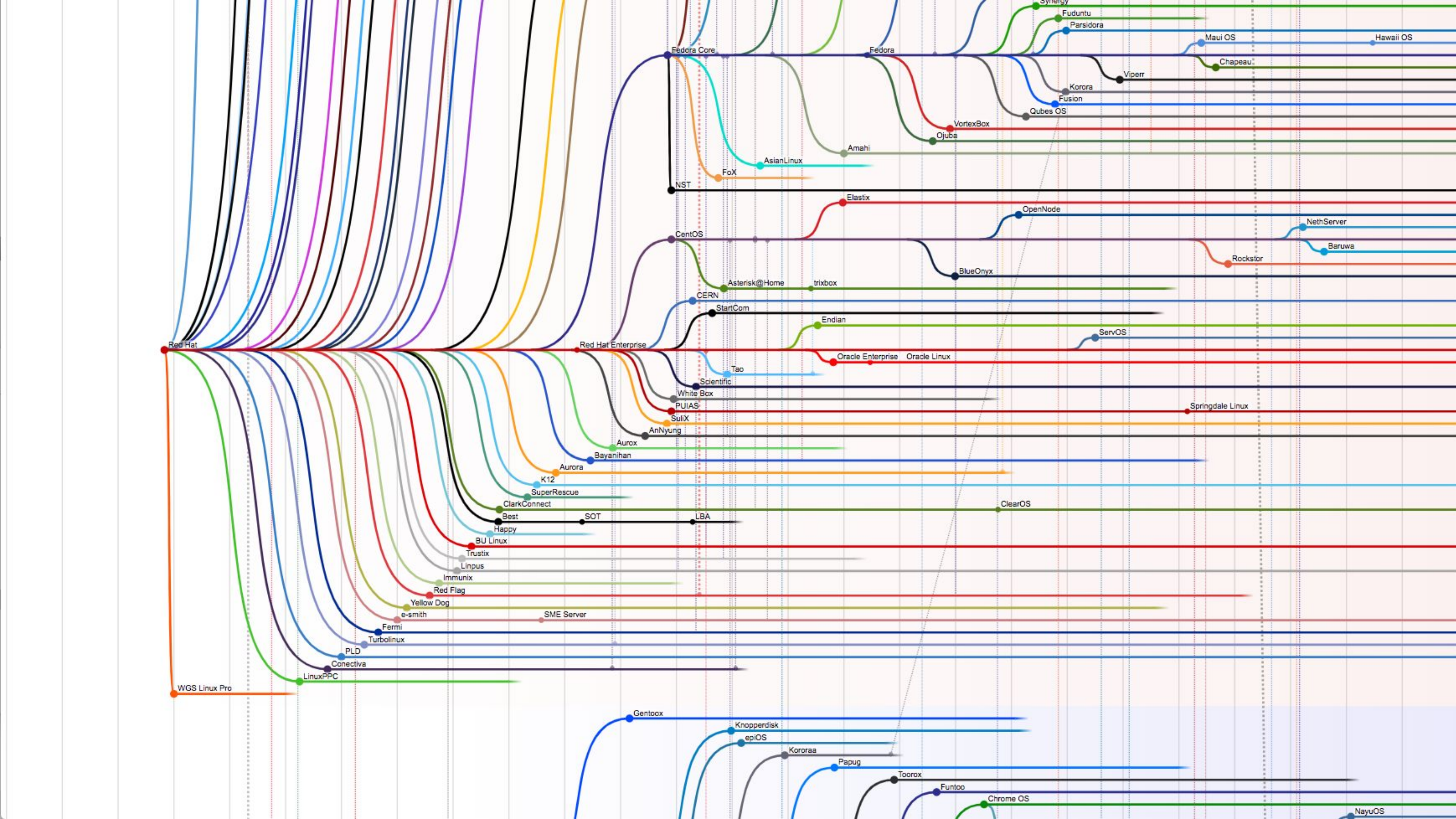
Debian
debian
Trisquel
VENENUX
Tuquito
2XOS
Epidemic
Librassoc
sidux
aptosid
siduction
Webconverger
GALPon MiniNo
Estrella Roja
Metamorphose
Canaima
Matriux
DoudouLinux
Linux Mint Debian Edition
GNUGuitarINUX
Liquid Lemur
Snowlinux
TurnKey
Xamin
LinuxBBQ
Point
Sparky
Rescatux
armbian
Tanglu
MX Linux
SELKS
PrimTux
Parrot
Devuan
Refracta
HandyLinux
RoboLinux
ToriOS
SteamOS
Ye Olde SteamOSe
Stephenson's Rocket
Rebellin
Univention Corporate Server
Raspbmc
OSMC
Raspbian
Moode
Raspyfi
Volumio
YunoHost
SolusOS
OpenMediaVault
Red Ribbon
SalineOS
Progress
UberStudent
Semplice
Tails
Untangle
Inquisitor
Proxmox
PelicanHPC
BOSS
PureOS
Vyatta
Brocade Vyatta vRouter
VyOS
NepaLinux
OS2005
Maemo
Meego Harmattan
Mamona
Elive
LliureX



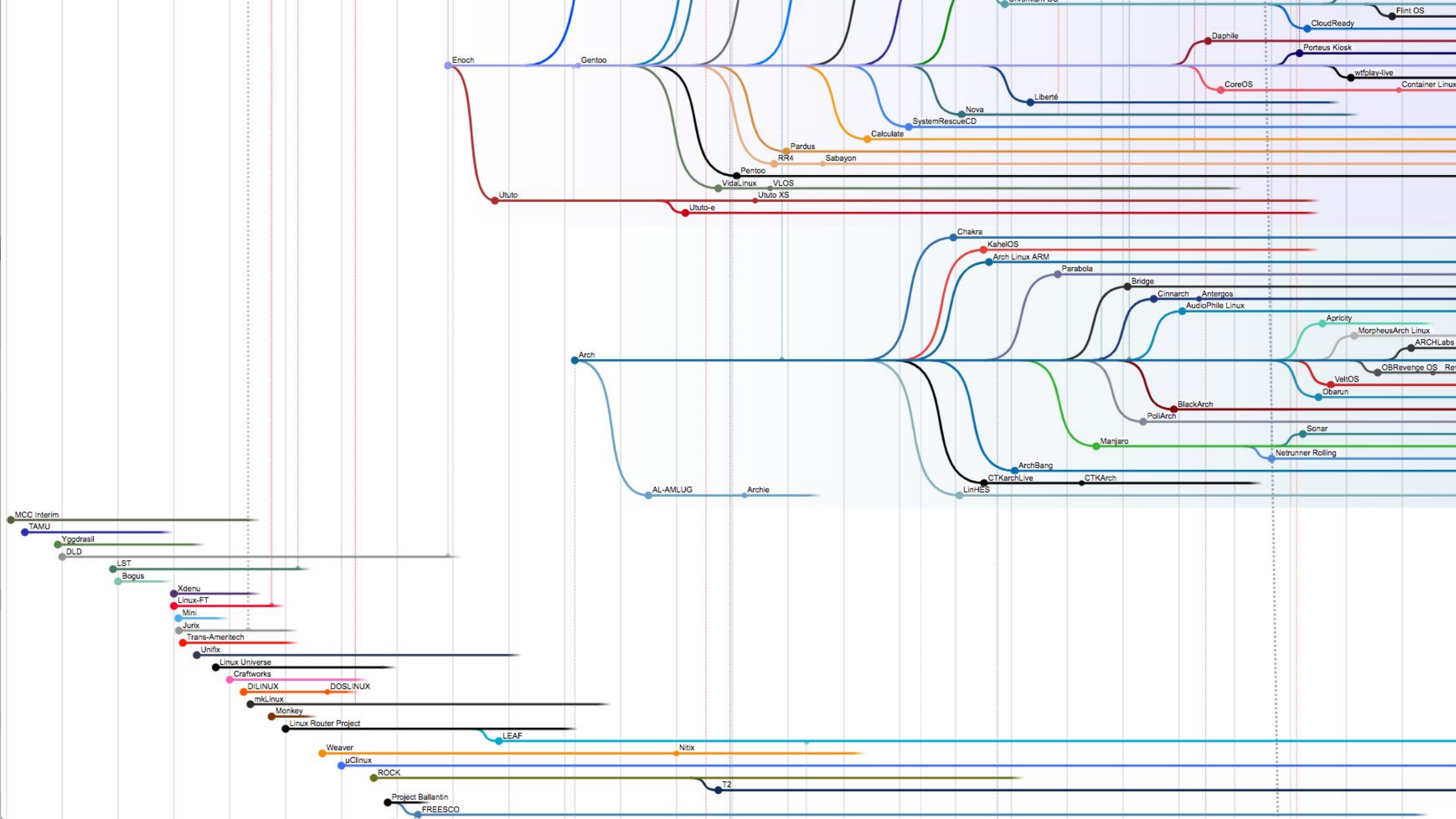
Voyage
Resulinux
Arco-Debian
Arc-Live
grml
Clonezilla Live
DRBL
Guadalinex
Amber
Xebian
ASLinux
BlackRhino
gnuLiNex
Lihuen
DeMuDi
64Studio
Progeny
Gibraltar
Storm
Corel
Xandros
SLS
MNIS
Stampede
VectorLinux
EvilEntity
Topologilinux
DeLi
ConnochaetOS
DeLi(cate)
Kwort
AUSTRUMI
GoblinX
Imagineos
HostGIS
NimbleX
Frugalware
NetSecL
Slackintosh
Voltalinux
Bluewhite64
Wifislax
Draco
Wifiway
Kongoni
Salix
Slackel
Slackware
Porteus
SlaXBMC
Linvo
PLATYPUX
SMS
Superb Mini Server
ARMedslack
Slackware ARM
SL4P
Absolute
How-Tux
RIP
pQui
Slamd64
MOPSLinux
AgiliaLinux
Kate
KateOS
Slack/390
Minislack
Zenwalk
DARKSTAR
Stux
STUX
Plamo
Wolvix
Gnu-HALO
Slax
SLAMPP
slackware linux



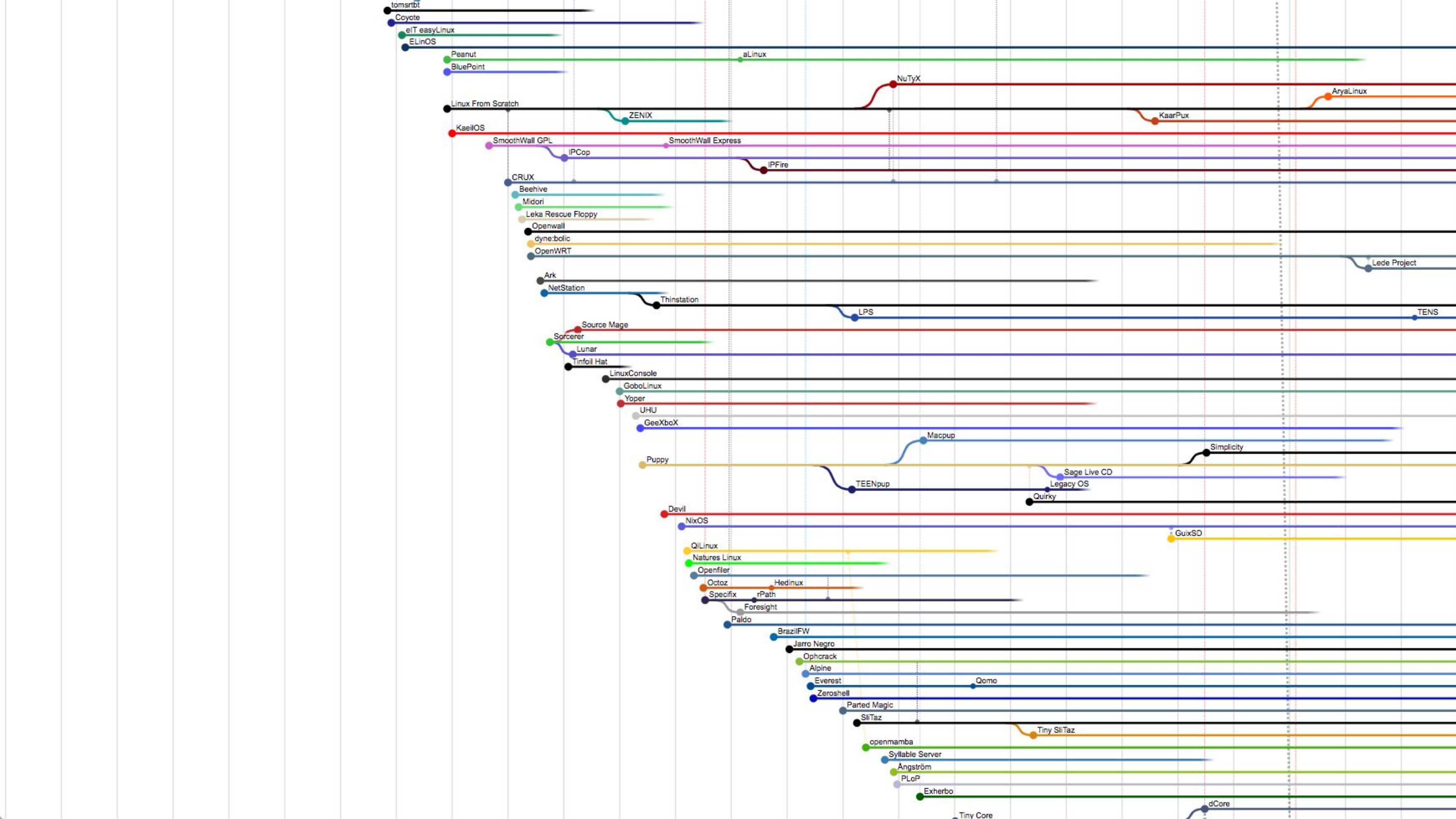
DNALinux
Splack Linux
Tiny
Burapha
Caixa Mágica
Stresslinux
S.u.S.E
SuSE
SUSE
Linkat
Keysoft
EasyNAS
OpenSUSE
GeckoLinux
Sun JDS
JOPUX
United Linux
Astaro
Sophos UTM
Caldera
SCO
UltraPenguin
Redmond
Lycoris
Buhawi
Eurielec
ALT
SAM
SAMity
OpenSLS
Annvix
Mageia
Mandrake
Mandriva
ROSA
OpenMandriva Lx
Unity Linux
blackPanther
Granular
Phinx
PCLinuxOS
Garuda
Demolinux
TinyMe
KRUD
Eridani
Vine
Armed
Kondara
ELX
Finnix
Miracle
Asianux
Pingo
Rocks
ASP
Independence
HP Secure
EnGarde
BLAG
Berry
ATmission
Momonga
MythDora
Ekaaty
Vixta
Simplis
Xange
eZeY
Moblin 2
Mer
MeeGo
Sailfish OS
Tizen
Hanthana
Synergy
Fuduntu
Parsidora
Maui OS
Hawaii OS
Fedora Core
Fedora
Chapeau
Viperr



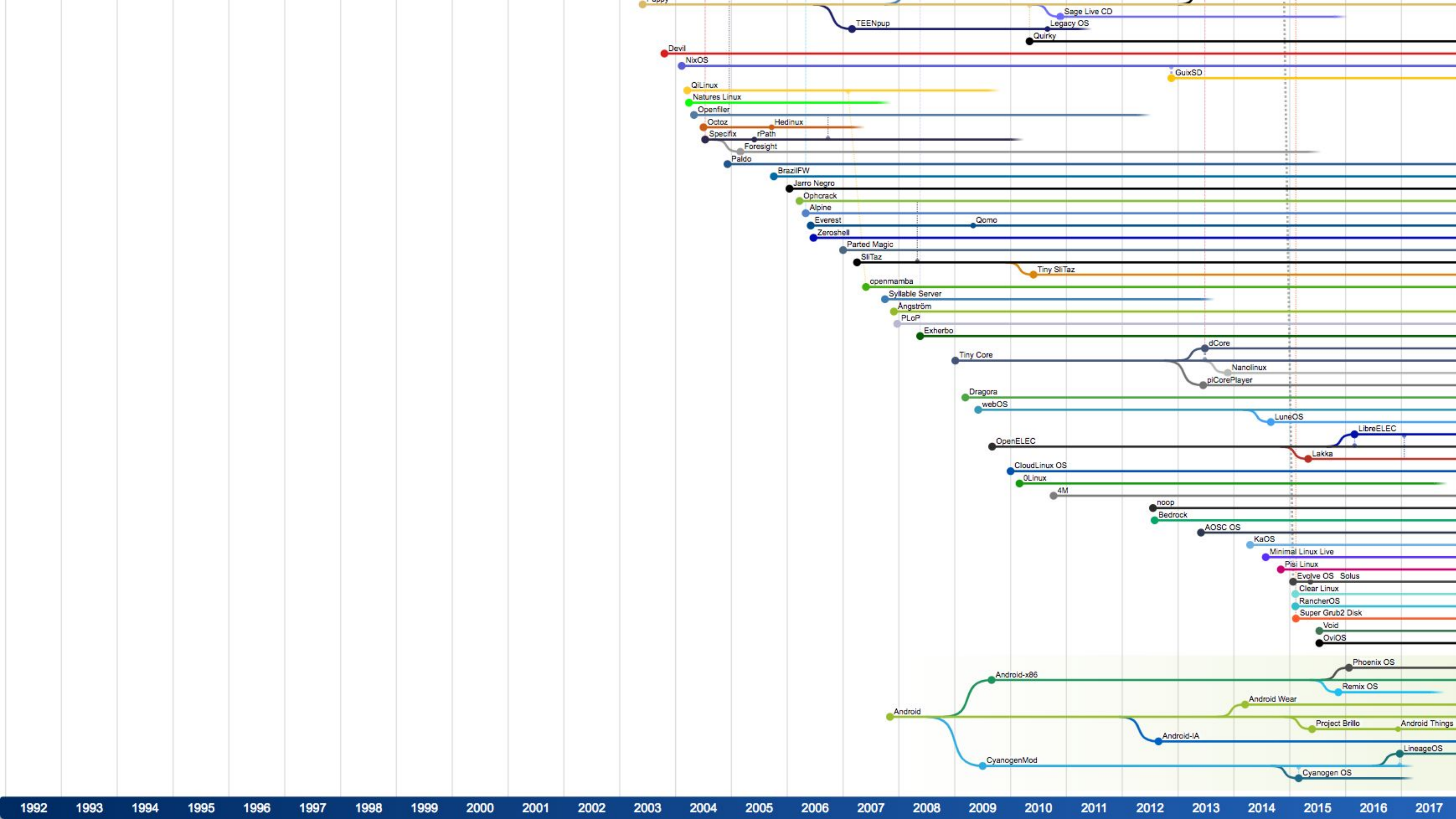
Synergy
Fuduntu
Parsidora
Maui OS
Hawaii OS
Fedora Core
Fedora
Chapeau
Viperr
Korora
Fusion
Qubes OS
VortexBox
Ojuba
Amahi
AsianLinux
FoX
NST
Elastix
OpenNode
NethServer
CentOS
Baruwa
Rockstor
BlueOnyx
Asterisk@Home
trixbox
CERN
StartCom
Endian
ServOS
Red Hat
Red Hat Enterprise
Oracle Enterprise
Oracle Linux
Tao
Scientific
White Box
PUIAS
Springdale Linux
SuliX
AnNyung
Aurox
Bayanihan
Aurora
K12
SuperRescue
ClarkConnect
ClearOS
Best
SOT
LBA
Happy
BU Linux
Trustix
Linpus
Immunix
Red Flag
Yellow Dog
e-smith
SME Server
Fermi
Turbolinux
PLD
Conectiva
LinuxPPC
WGS Linux Pro
Gentoox
Knopperdisk
epiOS
Kororaa
Papug
Toorox
Funtoo
Chrome OS
NayuOS



Flint OS
CloudReady
Daphile
Porteus Kiosk
Enoch
Gentoo
wtfplay-live
CoreOS
Container Linux
Liberté
Nova
SystemRescueCD
Calculate
Pardus
RR4
Sabayon
Pentoo
VidaLinux
VLOS
Ututo
Ututo XS
Ututo-e
Chakra
KahelOS
Arch Linux ARM
Parabola
Bridge
Cinnarch
Antergos
AudioPhile Linux
Apricity
MorpheusArch Linux
ARCHLabs
Arch
OBRevenge OS
VeltOS
Obarun
BlackArch
PoliArch
Sonar
Manjaro
Netrunner Rolling
ArchBang
CTKarchLive
CTKArch
AL-AMLUG
Archie
LinHES
MCC Interim
TAMU
Yggdrasil
DLD
LST
Bogus
Xdenu
Linux-FT
Mini
Jurix
Trans-Ameritech
Unifix
Linux Universe
Craftworks
DILINUX
DOSLINUX
mkLinux
Monkey
Linux Router Project
LEAF
Weaver
Nitix
µClinux
ROCK
T2
Project Ballantin
FREESCO



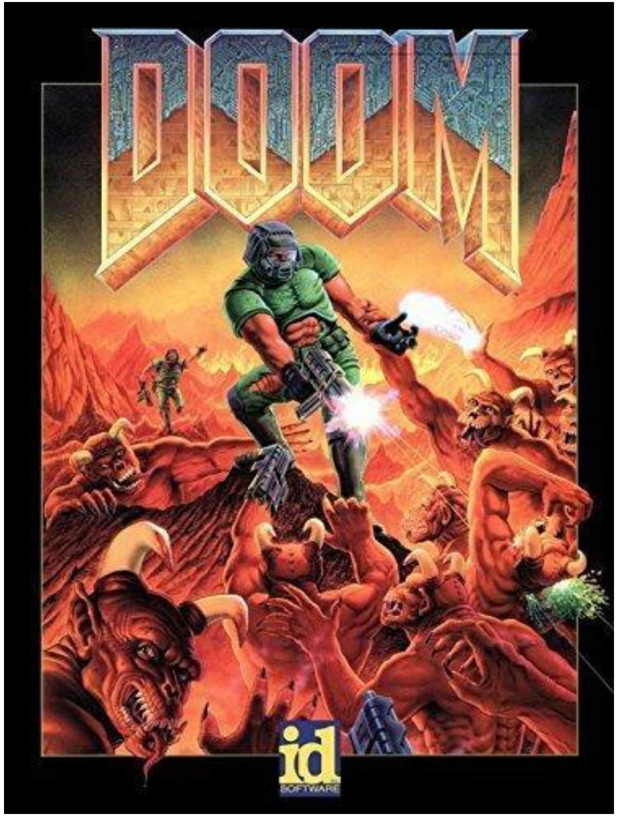
tomsrtbt
Coyote
eIT easyLinux
ELinOS
Peanut
aLinux
BluePoint
NuTyX
AryaLinux
Linux From Scratch
ZENIX
KaarPux
KaeilOS
SmoothWall GPL
SmoothWall Express
IPCop
IPFire
CRUX
Beehive
Midori
Leka Rescue Floppy
Openwall
dyne:bolic
OpenWRT
Lede Project
Ark
NetStation
Thinstation
LPS
TENS
Source Mage
Sorcerer
Lunar
Tinfoil Hat
LinuxConsole
GoboLinux
Yoper
UHU
GeeXboX
Macpup
Simplicity
Puppy
Sage Live CD
TEENpup
Legacy OS
Quirky
Devil
NixOS
GuixSD
QiLinux
Natures Linux
Openfiler
Octoz
Hedinux
Specifix
rPath
Foresight
Paldo
BrazilFW
Jarro Negro
Ophcrack
Alpine
Everest
Qomo
Zeroshell
Parted Magic
SliTaz
Tiny SliTaz
openmamba
Syllable Server
Ångström
PLoP
Exherbo
dCore
Tiny Core


Sage Live CD
TEENpup
Legacy OS
Quirky
Devil
NixOS
GuixSD
QiLinux
Natures Linux
Openfiler
Octoz
Hedinux
Specifix
rPath
Foresight
Paldo
BrazilFW
Jarro Negro
Ophcrack
Alpine
Everest
Qomo
Zeroshell
Parted Magic
SliTaz
Tiny SliTaz
openmamba
Syllable Server
Ångström
PLoP
Exherbo
dCore
Tiny Core
Nanolinux
piCorePlayer
Dragora
webOS
LuneOS
LibreELEC
OpenELEC
Lakka
CloudLinux OS
0Linux
4M
noop
Bedrock
AOSC OS
KaOS
Minimal Linux Live
Pisi Linux
Evolve OS
Solus
Clear Linux
RancherOS
Super Grub2 Disk
Void
OviOS
Phoenix OS
Android-x86
Remix OS
Android Wear
Android
Project Brillo
Android Things
Android-IA
LineageOS
CyanogenMod
Cyanogen OS
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017

Carnegie Mellon University


The Oregon Trail
Introduction
Options
Quit
Travel the Trail


Instant Play
New Game
Endless telegraph
THE OREGON TRAIL



Carnegie Mellon University

Overview of Workshop

1. Why Software Preservation?
2. What is Software Curation?
    a. Scope and Description
    b. Migration
    c. Reproduction and Access
    d. Outreach and Culture
3. Trends
4. Efforts at CMU

Carnegie Mellon University

Migration

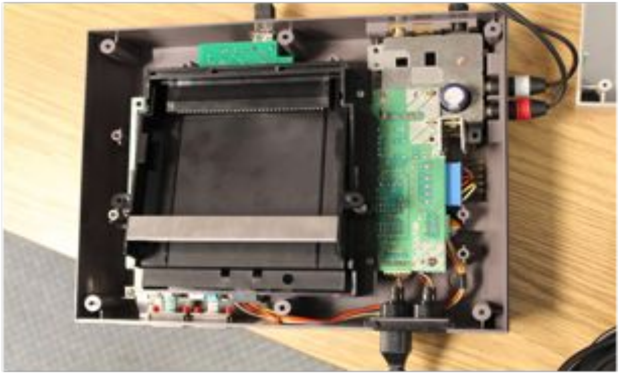
Imaging and Storage

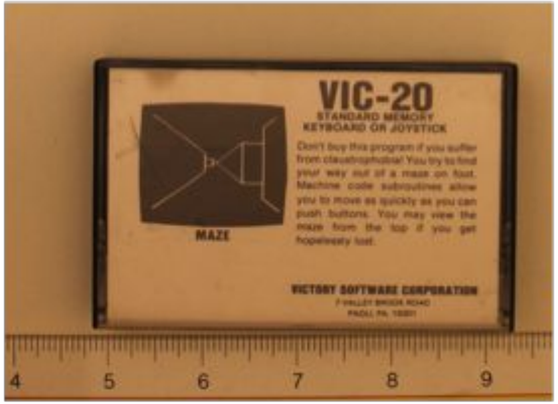
Digital Forensic Workflows

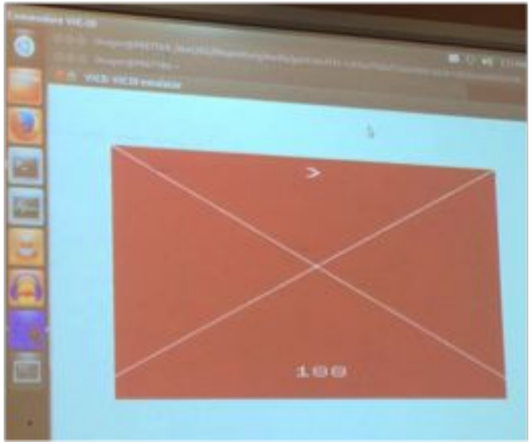
Digital Rights Management

Digital Migration Strategies

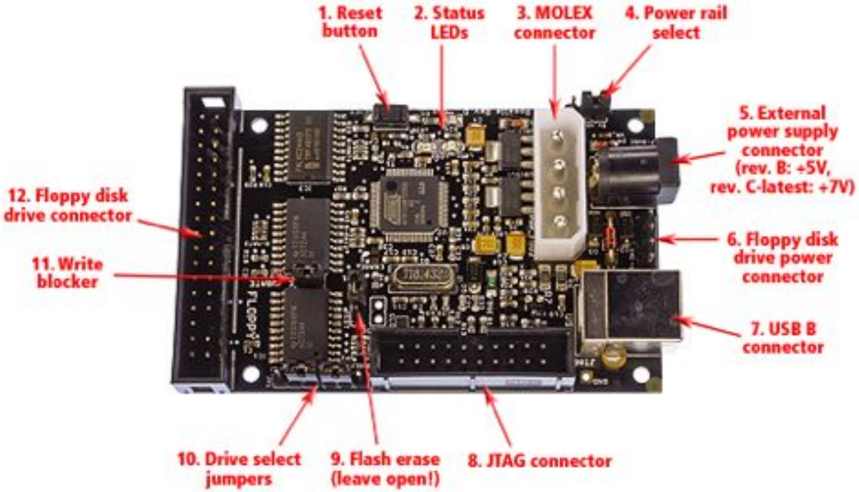
Migration







Carnegie Mellon University

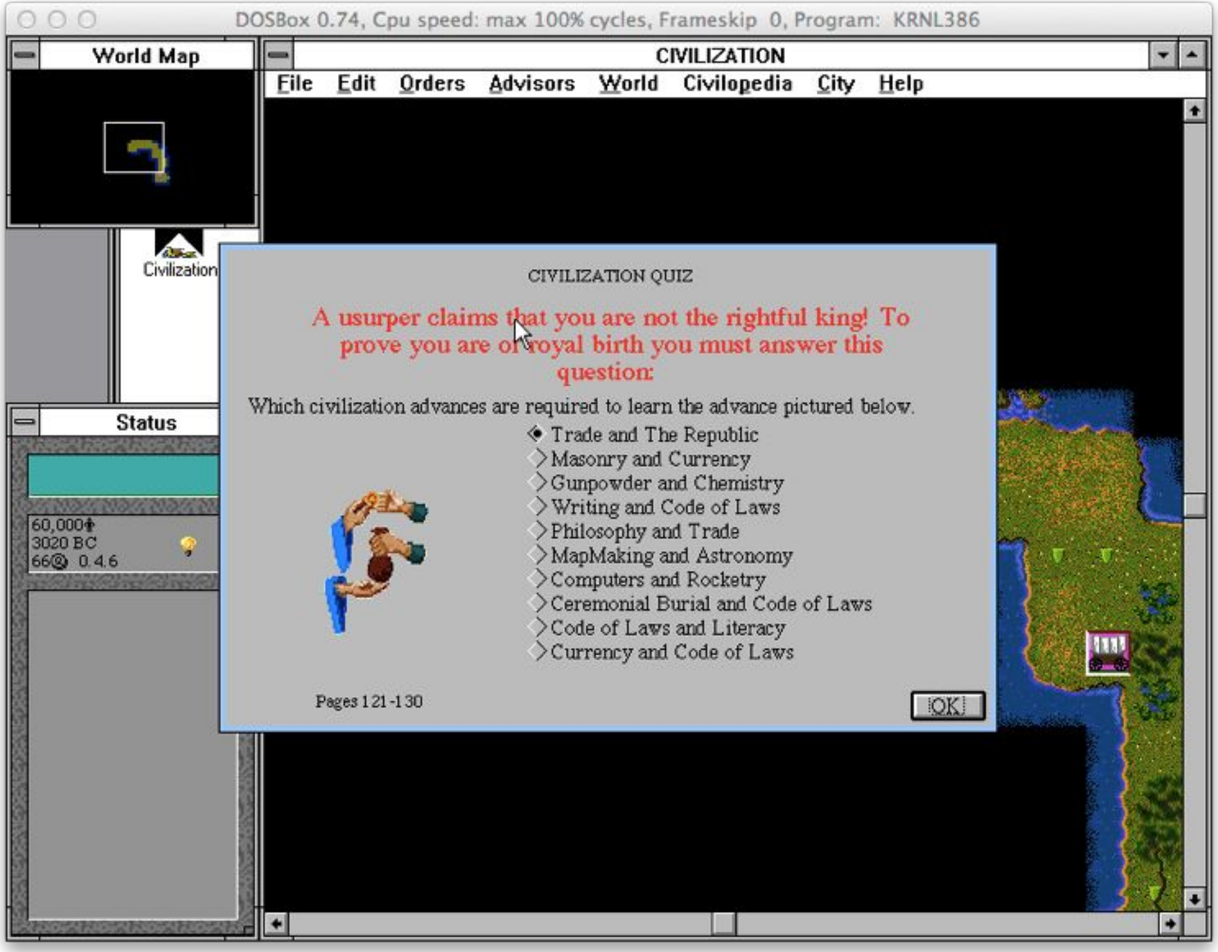


Carnegie Mellon University

BitCurator
CONSORTIUM

FTK

Carnegie
Mellon
University



Carnegie Mellon University

Digital Migration Strategies

Once imaged or ingested, need a *forever* storage strategy

- Repositories change and evolve, data needs to migrate with those changes
- Provenance information becomes more important

Current repositories are still playing significant catch up

No current long-term solutions available for software data

Overview of Workshop

1. Why Software Preservation?
2. What is Software Curation?
   a. Scope and Description
   b. Migration
   c. Reproduction and Access
   d. Outreach and Culture
3. Trends
4. Efforts at CMU

Carnegie Mellon University

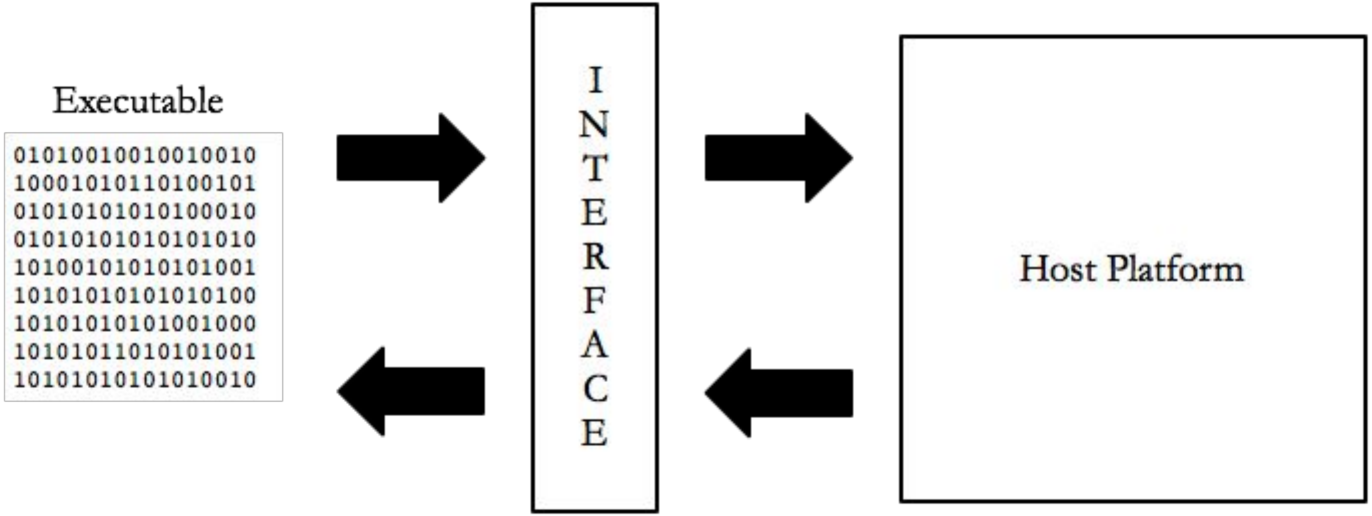
Reproduction Strategies

Virtualization

Containerization

Emulation

Hardware Preservation



Emulator

Container

Virtual Machine

MAME
MULTIPLE ARCADE MACHINE EMULATOR

QEMU

Stella
A multi-platform Atari 2600 VCS emulator

Snes9x
Super Nintendo Emulation

FCEUX

Client Side Emulation

Local executable emulation

JavaScript in browser emulation

Pros:

- Lower latency
- Locally inspectable
- If browser based, shareable and single requirement

Cons:

- Dependent on specific system configuration
- Not easily shareable if local executable
- Legally dubious

Server Side Emulation

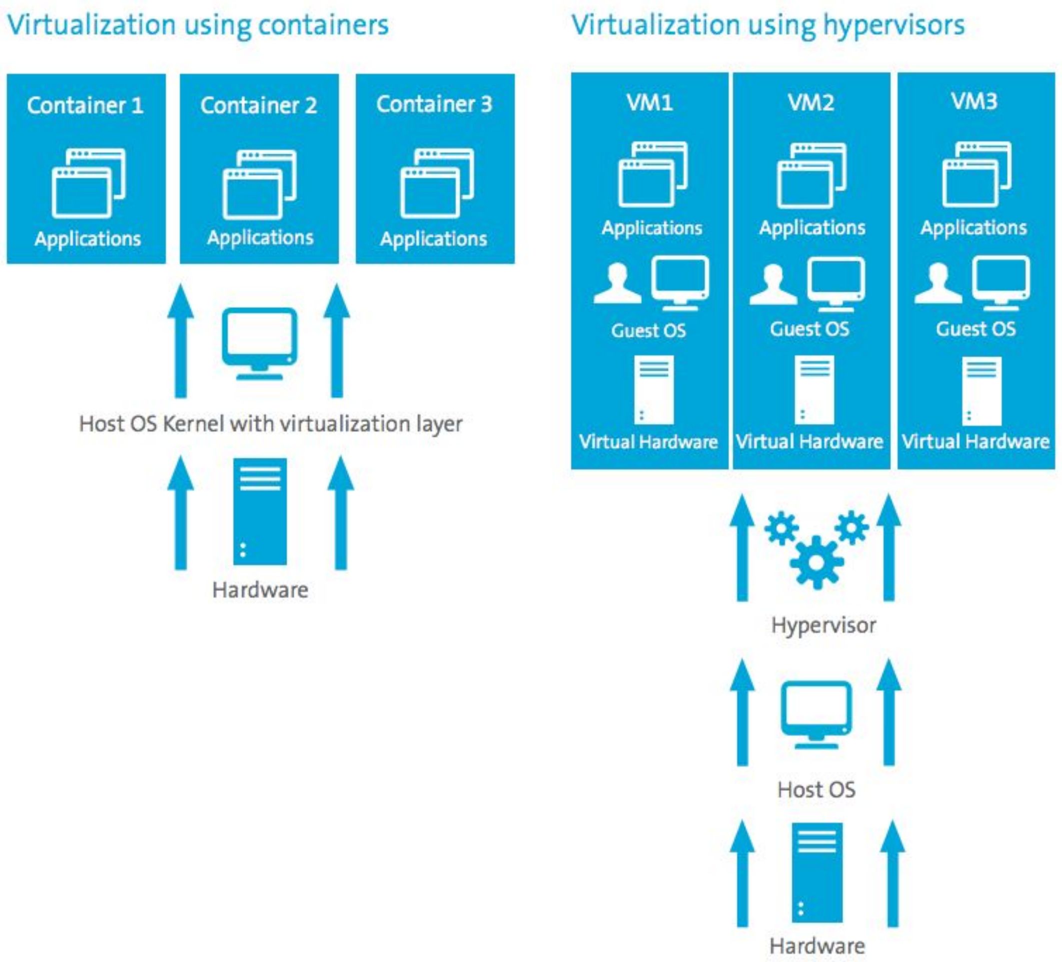
Emulation in cloud

Pros:

- Legally more appealing
- Management and maintenance are centralized
- More easy to roll into services

Cons:

- Latency issues
- Much less introspection
- Cloud is a preservation issue in of itself



Carnegie Mellon University

docker

kubernetes

MESOS

ORACLE VirtualBox

vmware®

Hardware Preservation

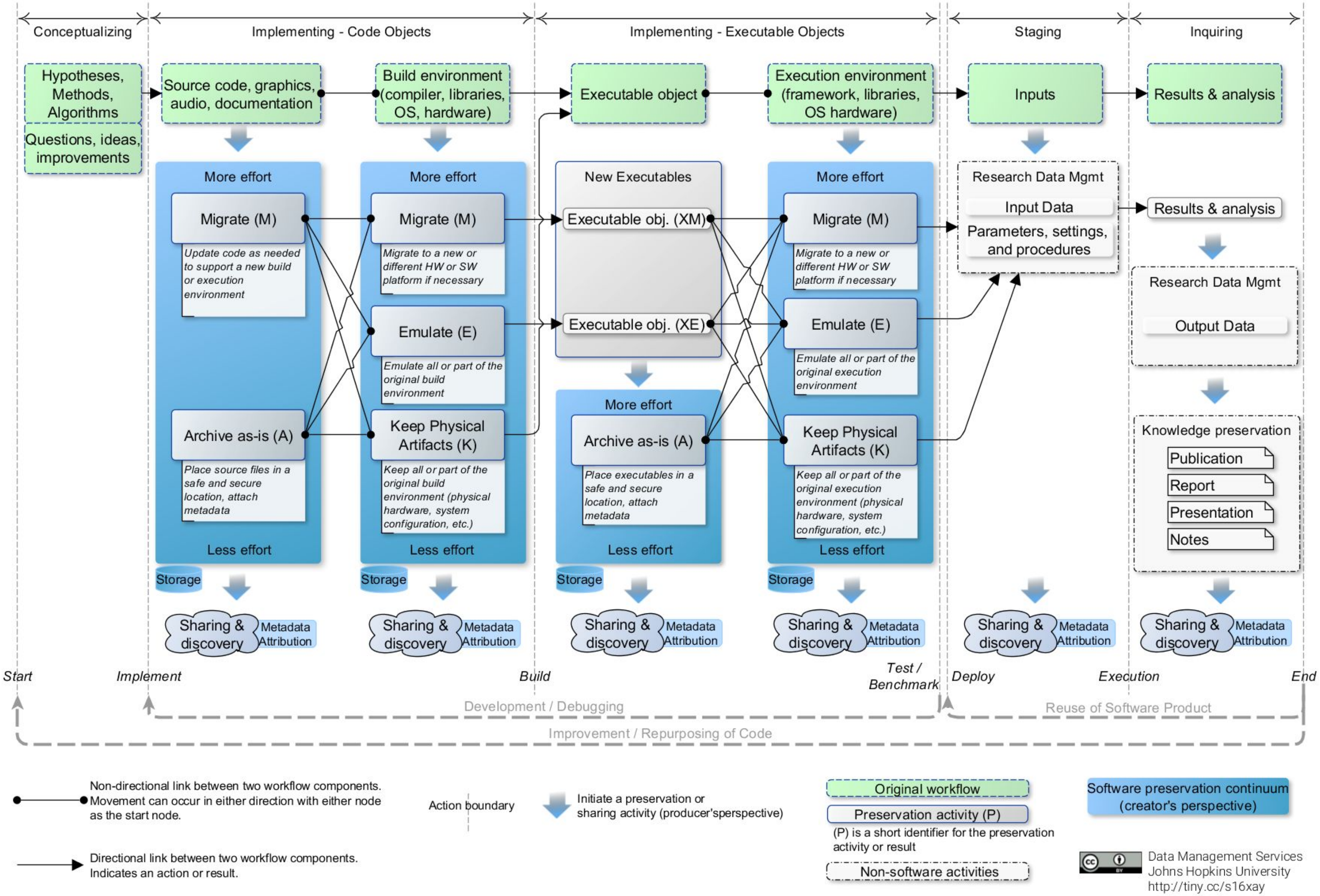
Cultural software objects are designed for specific hardware and interactions

- Computer Games
- Interactive Art Installation
- Digital Art
- Other digital media works

Socio-cultural context is not in a VM or container

Hardware peripheral and displays are not replicated

Carnegie Mellon University


Conceptualizing
Implementing - Code Objects
Implementing - Executable Objects
Staging
Inquiring
Hypotheses, Methods, Algorithms
Questions, ideas, improvements
Source code, graphics, audio, documentation
Build environment (compiler, libraries, OS, hardware)
Executable object
Execution environment (framework, libraries, OS hardware)
Inputs
Results & analysis
More effort
Migrate (M)
Update code as needed to support a new build or execution environment
Archive as-is (A)
Place source files in a safe and secure location, attach metadata
Less effort
More effort
Migrate (M)
Migrate to a new or different HW or SW platform if necessary
Emulate (E)
Emulate all or part of the original build environment
Keep Physical Artifacts (K)
Keep all or part of the original build environment (physical hardware, system configuration, etc.)
Less effort
New Executables
Executable obj. (XM)
Executable obj. (XE)
More effort
Archive as-is (A)
Place executables in a safe and secure location, attach metadata
Less effort
More effort
Migrate (M)
Migrate to a new or different HW or SW platform if necessary
Emulate (E)
Emulate all or part of the original execution environment
Keep Physical Artifacts (K)
Keep all or part of the original execution environment (physical hardware, system configuration, etc.)
Less effort
Research Data Mgmt
Input Data
Parameters, settings, and procedures
Results & analysis
Research Data Mgmt
Output Data
Knowledge preservation
Publication
Report
Presentation
Notes
Storage
Sharing & discovery
Metadata Attribution
Start
Implement
Build
Test / Benchmark
Deploy
Execution
End
Development / Debugging
Reuse of Software Product
Improvement / Repurposing of Code
Non-directional link between two workflow components. Movement can occur in either direction with either node as the start node.
Directional link between two workflow components. Indicates an action or result.
Action boundary
Initiate a preservation or sharing activity (producer'sperspective)
Original workflow
Preservation activity (P)
(P) is a short identifier for the preservation activity or result
Non-software activities
Software preservation continuum (creator's perspective)
Data Management Services
Johns Hopkins University
http://tiny.cc/s16xay

Overview of Workshop

1. Why Software Preservation?
2. What is Software Curation?
   a. Scope and Description
   b. Migration
   c. Reproduction and Access
   d. Outreach and Culture
3. Trends
4. Efforts at CMU

Carnegie Mellon University

Outreach

Locating significant materials around CMU community

Implement reproducible practices inside research labs and departments

Sustainable software development

Overview of Workshop

1. Why Software Preservation?
2. What is Software Curation?
    a. Scope and Description
    b. Migration
    c. Reproduction and Access
    d. Outreach and Culture
3. Trends
4. Efforts at CMU

Carnegie Mellon University

Sustainable Software Development

Testing oriented

Well documented

Instrumentation

- Continuous integration
- Package management
- Configuration management
- Version Control

Tools and Organizations

Reproducibility Tools Supporting Software

- Code Ocean - http://codeocean.com
- Occam - http://occam.cs.pitt.edu
- Collective Knowledge - http://cknowledge.org/
- Umbrella - http://ccl.cse.nd.edu/software/umbrella/
- ReproZip - https://www.reprozip.org/

Organizations supporting software preservation

- Software Sustainability Institute (SSI) - UK Organization - http://www.software.ac.uk
- Data and Software Preservation for Open Science (DASPOS) - CERN - http://daspos.org
- Software Preservation Network (SPN) - US Memory Institutions
  - http://www.softwarepreservationnetwork.org

Carnegie Mellon University

ACM Reproducibility

- Repeatability (Same team, same experimental setup)
- Replicability (Different team, same experimental setup)
- Reproducibility (Different team, different experimental setup)

ACM Badging Levels

Artifacts Available

- Software and data present in publication are available for download and investigation

Artifacts Evaluated - Functional

- Software and data have been audited and validated as working

Artifacts Evaluated - Reusable

- Software and data have been audited by a third party, are functional, and significantly oriented toward reusability through documentation, code / software organization, etc.

Results Replicated

- Software and data have been used to validate results

Results Reproduced

- Different software and data have been used to validate results

Carnegie Mellon University

Overview of Workshop

1. Why Software Preservation?
2. What is Software Curation?
   a. Scope and Description
   b. Migration
   c. Reproduction and Access
   d. Outreach and Culture
3. Trends
4. Efforts at CMU

Tools and Services at CMU

Kilthub Repository

- Source code
- Research data and software executable binaries

Code Ocean (Beta)

- Targeted for August 2019
- Reproducibility platform

Emulation as a Service (EaaS)

- Currently in research beta
- Distributed containerized execution contexts

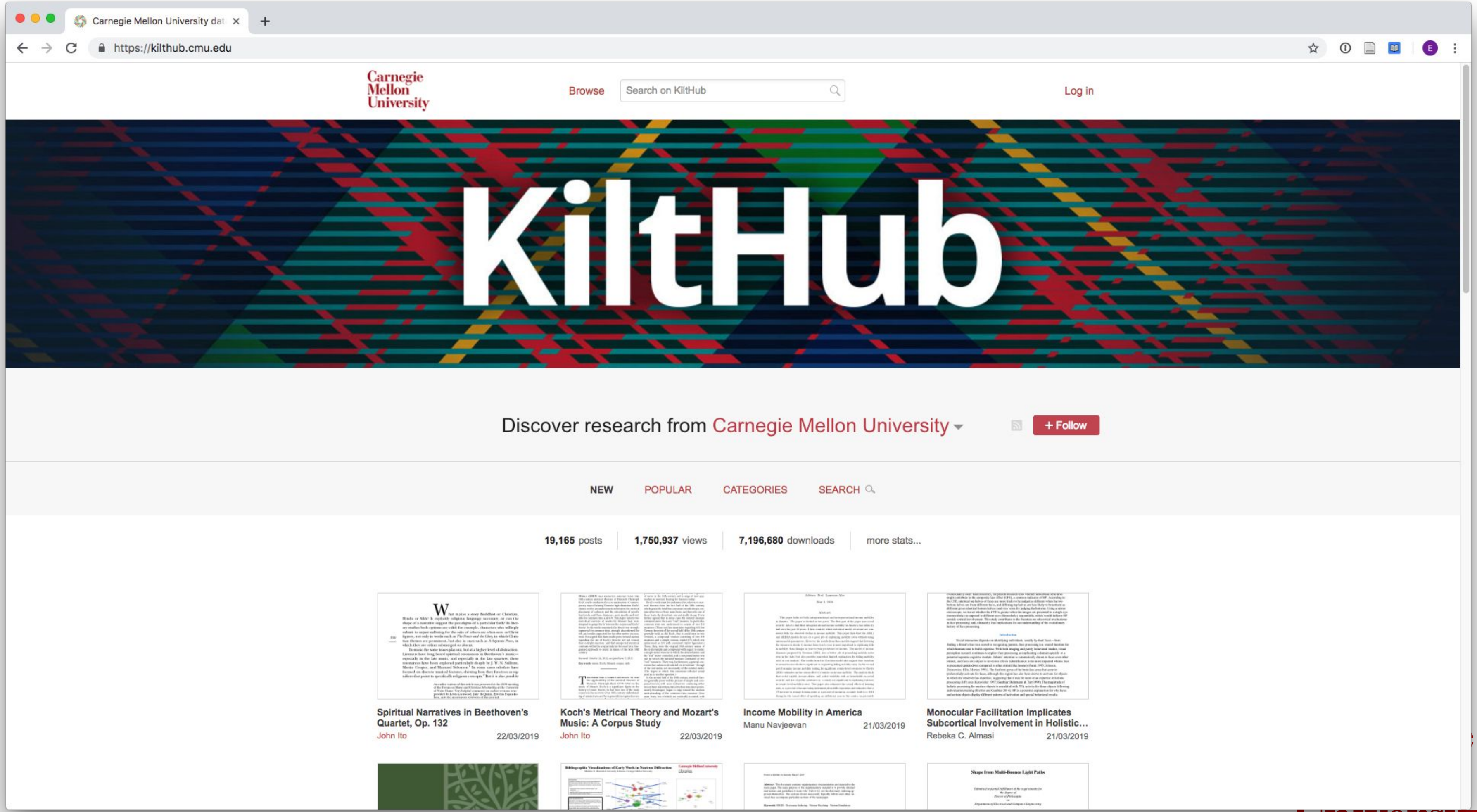
Software and Data Carpentries

- Two day courses on basic research support tools like Python, R, and Git

History of Science and Technology at CMU (HOST@CMU)

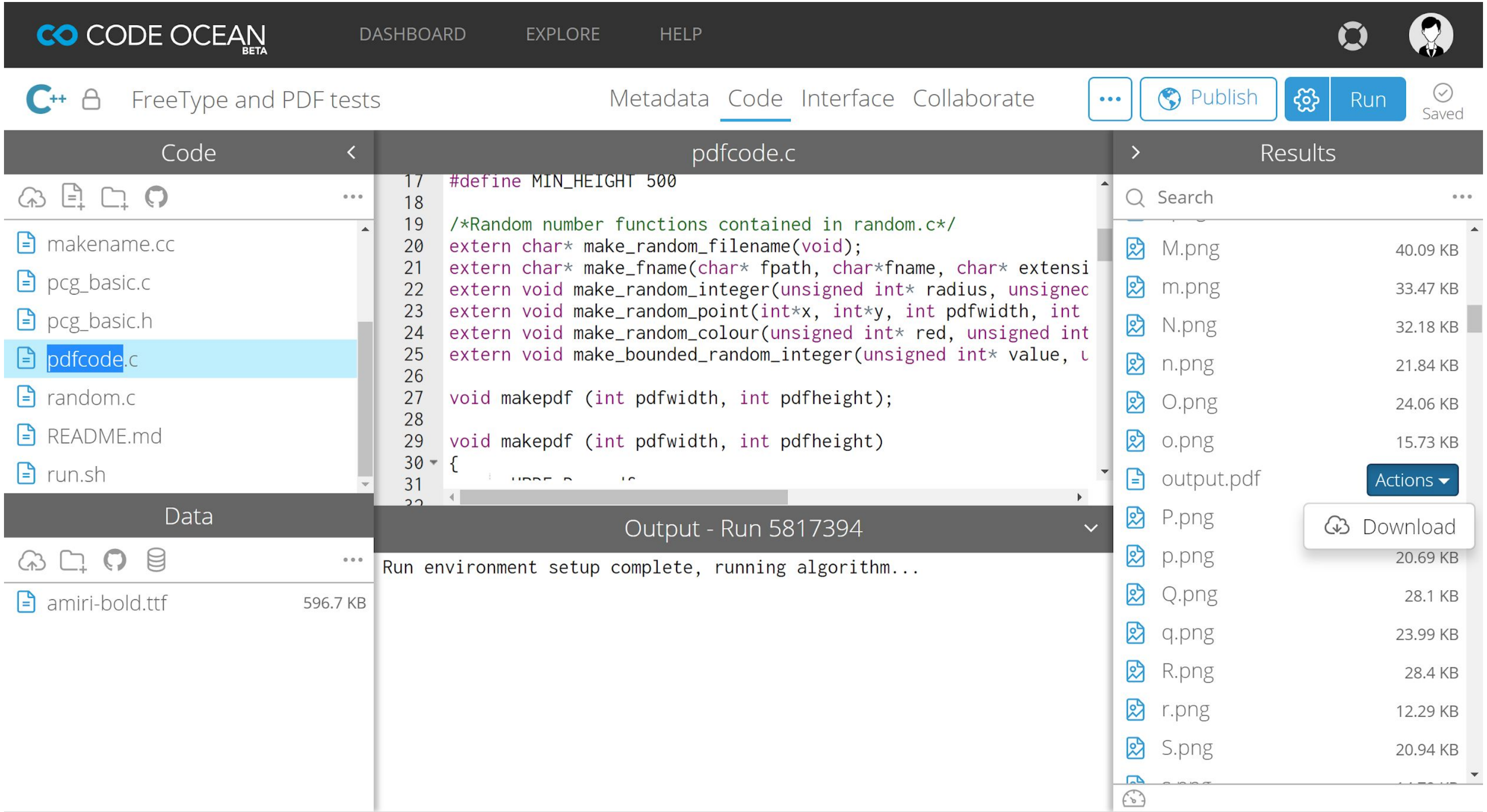
- Interdisciplinary initiative to locate and celebrate CMU technical history

Carnegie Mellon University


Carnegie Mellon University dat
https://kilthub.cmu.edu
Carnegie Mellon University
Browse
Search on KiltHub
Log in
KiltHub
Discover research from Carnegie Mellon University
+ Follow
NEW
POPULAR
CATEGORIES
SEARCH
19,165 posts
1,750,937 views
7,196,680 downloads
more stats...
Spiritual Narratives in Beethoven's Quartet, Op. 132
John Ito
22/03/2019
Koch's Metrical Theory and Mozart's Music: A Corpus Study
John Ito
22/03/2019
Income Mobility in America
Manu Navjeevan
21/03/2019
Monocular Facilitation Implicates Subcortical Involvement in Holistic...
Rebeka C. Almasi
21/03/2019
Shape from Multi-Bounce Light Paths

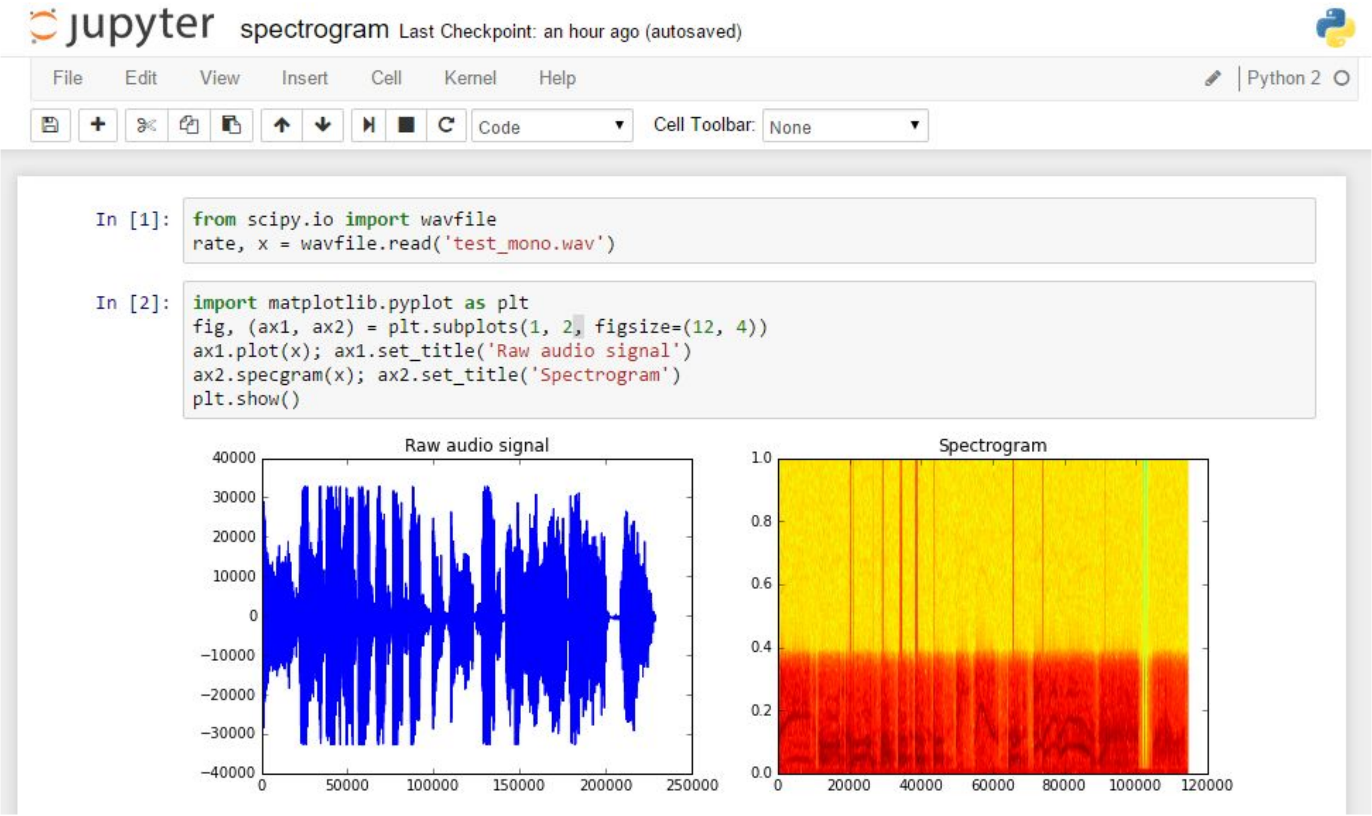
University

Code Ocean



CODE OCEAN BETA
DASHBOARD
EXPLORE
HELP
FreeType and PDF tests
Metadata
Code
Interface
Collaborate
Publish
Run
Saved
Code
makename.cc
pcg_basic.c
pcg_basic.h
pdfcode.c
random.c
README.md
run.sh
Data
amiri-bold.ttf 596.7 KB
pdfcode.c
#define MIN_HEIGHT 500
/*Random number functions contained in random.c*/
extern char* make_random_filename(void);
extern char* make_fname(char* fpath, char*fname, char* extensi
extern void make_random_integer(unsigned int* radius, unsignec
extern void make_random_point(int*x, int*y, int pdfwidth, int
extern void make_random_colour(unsigned int* red, unsigned int
extern void make_bounded_random_integer(unsigned int* value, u
void makepdf (int pdfwidth, int pdfheight);
void makepdf (int pdfwidth, int pdfheight)
{
Output - Run 5817394
Run environment setup complete, running algorithm...
Results
Search
M.png 40.09 KB
m.png 33.47 KB
N.png 32.18 KB
n.png 21.84 KB
O.png 24.06 KB
o.png 15.73 KB
output.pdf
Actions
P.png
Download
p.png 20.69 KB
Q.png 28.1 KB
q.png 23.99 KB
R.png 28.4 KB
r.png 12.29 KB
S.png 20.94 KB

Jupyter Notebooks



Carnegie Mellon University

Emulation as a Service Infrastructure

Interuniversity Distributed Network of Emulation Nodes

Six partner institutions

Environment Contexts with full software installation

Access to legacy files and objects

Carnegie Mellon University

Questions/Comments

Eric Kaltman
Data Curation Fellow
ekaltman@andrew.cmu.edu

Carnegie Mellon University