Kodi - Open Source Home Entertainment Software (formerly known as XBMC)

Ejal de Klerk
Martijn Kaijser

Who Are we?

Martijn Kaijser (“*Martijn*”)
Martijn@kodi.tv

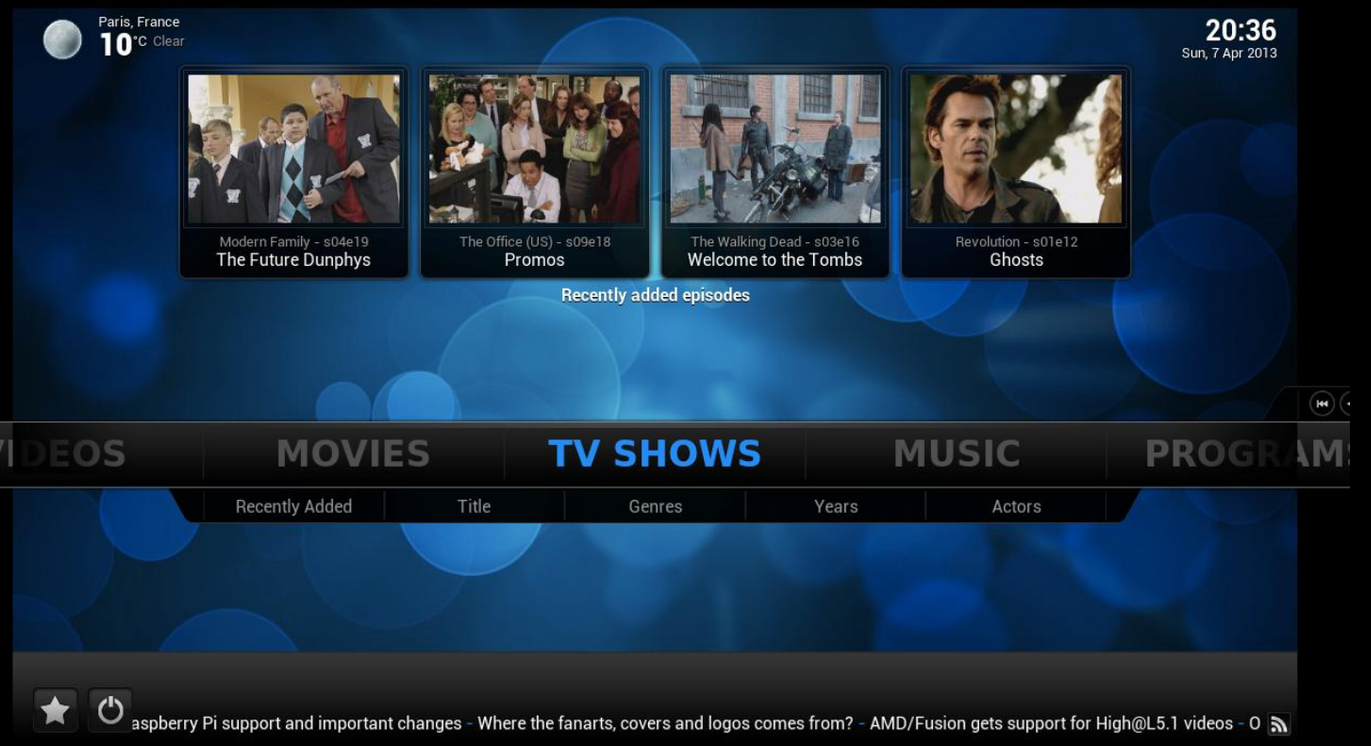
- User since Dharma 10.1 ~2011
- Helped with support in forums/social media
- Started with writing python add-ons
- Joined the team in 2011 and XBMC Foundation in 2012
- Release manager for 14.x and 15.x

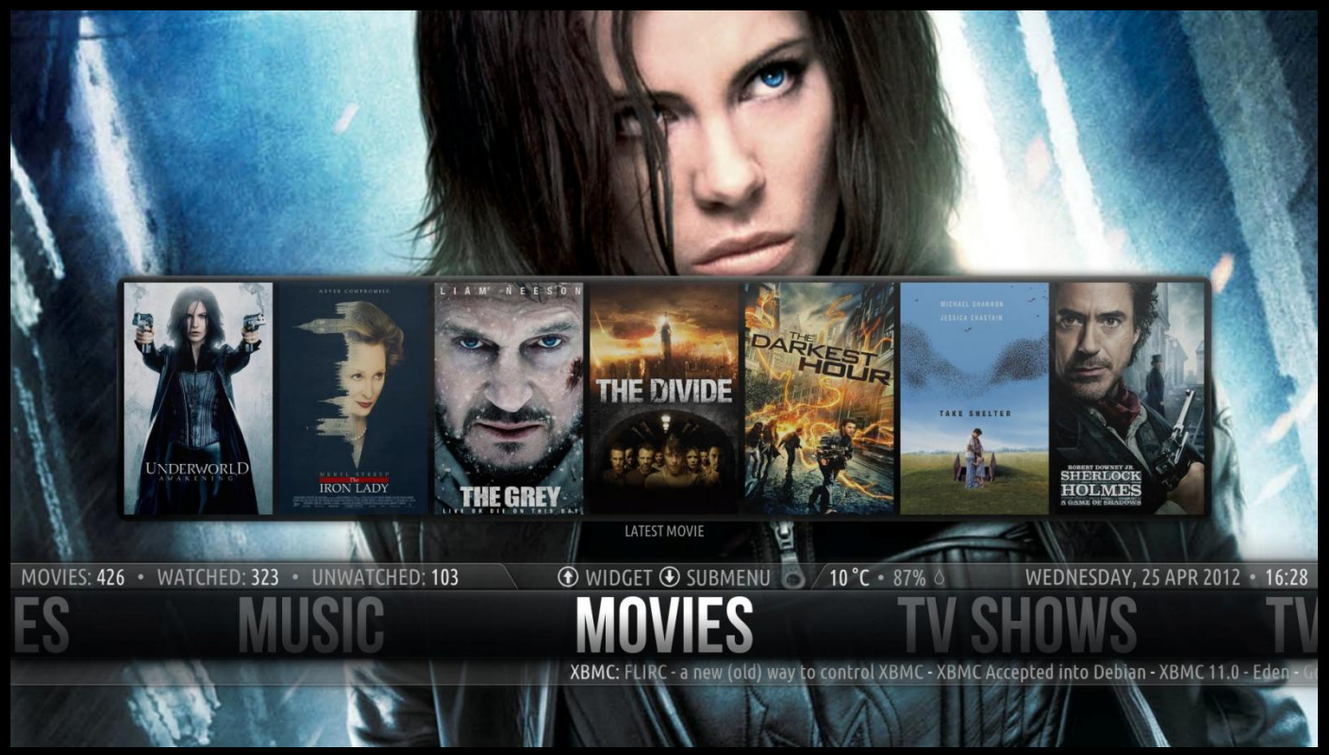
Ejal de Klerk (“*Kib*”)
Kib@kodi.tv

- User since XBMC 10 ~2011
- Helped with support in IRC and forums
- Started with modifying a abandoned skin (Neon) for Frodo and Gotham
- Joined the team and Foundation in 2013
- Focused on server infrastructure

What is Kodi

- Award winning software media player and entertainment hub
- Worlds largest open source multimedia project
- Free and open source (GPL2)
- Supports every common media retrieval method, local to network to internet

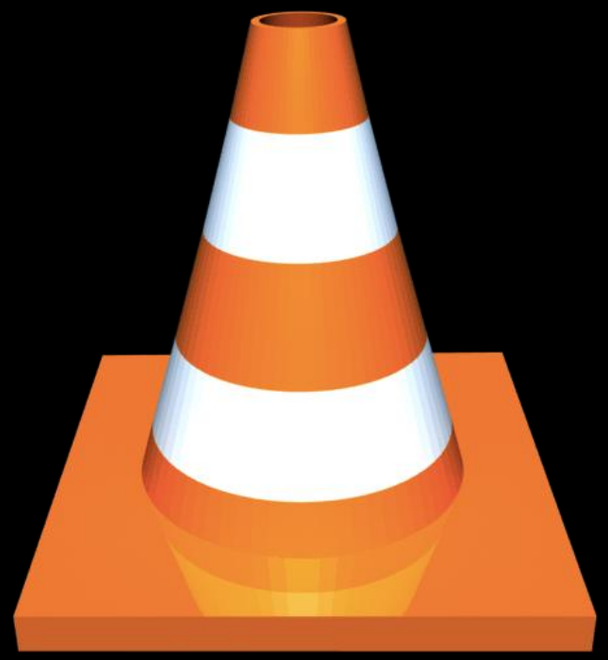




Platforms Kodi runs on

- Linux (x86, ARM, Freescale, MIPS)
- Android (4.0+, ARM, x86, Freescale)
- iOS (iOS5-8, a4+ does full 1080p)
- OSX
- Windows (Vista+)
- FreeBSD
- Almost any hardware that is capable
- Original Xbox (xbmc4box is still unofficially maintaining this port)

How is Kodi different?



- Unified interface to view pictures, videos and music
- Pioneered the 10 foot interface for optimal couch viewing
- Fully skinable using XML
- Same codebase on all platforms (almost)
- Extensible add-on system written in Python
- JSON-RPC interface for almost total control
- First to integrate features related to media consumption

Kodi Distributions

- OpenELEC: The undisputed champion distro for Kodi. Runs on nearly any x86/ARM devices w/ Open GL 2.0+
- Kodibuntu: Standard Ubuntu w/ Kodi on top of it maintained by the team
- Raspbian: Only for the Raspberry Pi based off Debian
- SPMC: an unofficial test fork from Koying, the only Android dev, which is currently in Android-based App Stores
- Many others. List is available on our wiki (incomplete)

Demo!

Recommended Hardware

- None.
  As a team, we don’t recommend anything as we are multi-platform and hardware independant
- With that said…

Hardware we would avoid

- Apple TV’s (ended with 14.1 release)
- Allwinner-based Android devices
  (…or any non name brand Android device in general)
- Slower hardware which reduces the snappier experience of the user interface we like to see.

Hardware we suggest

- x86 device running OpenELEC such as a Zotac, NUC, Gigabyte BRIX or Chromebox
- Android devices such as Fire TV or Nexus Player
- Low-end ARM devices provide a basic experience, but x86 for the ‘real’ experience.
- FLIRC IR adapter, learns any remote control and exposes USB-HID to the computer
- HDMI CEC Adapter, interfacing with existing remotes on supported devices

Where’d XBMC go?

- Devices were being sold as ‘XBMC’ without our permission and tainting our name
- Groups claiming to be official ‘XBMC’
- Applied for Trademark far too late
- Received opposition on our XBMC trademark application in Europe, came to an agreement to change our name

History - Xbox

- Nov 2001 - Xbox released in the US
- Jan 2002 - Xbox hacked by Andrew Huang (Bunnie)
- Nov 2002 - xbplayer and YAMP started by d7o3g4q, RUNTiME and Frodo
- Dec 14 2002 - Xbox Media Player 2.0 released

History - Why Xbox?

- Networked x86 Appliance with TV-Out
- Cheap - 733Mhz p3 Celeron, Ethernet, 64Mb RAM, 8GB HDD, GeForce 3MX graphics for $299 in 2002.
- Easy - running a stripped down Windows Kernel, based off Windows NT and DirectX 8.1
- Hackable - 90 day warranty and any mistakes made in security meant easy/cheap mods.

History - opensource on a closed system?

- All initial homebrew required the Microsoft XDK
- You could distribute the source, but needed the XDK to compile
- Essentially forced an opensource ecosystem
- Binaries were ‘illegal’

History - XBMC is born

- Developed in C/C++
- Structured around a game loop
- GUI library defines widgets from XML files and textures
- An embedded python interpreter allows easier extension via “plug-in scripts” , which we call “add-ons”
- Multiple player cores (dvdplayer, paplayer, mplayer)
- Relies on many open source libraries

vgmstream
libcmyth
libmad
libshout
StSoundLib
libsid
Platinum
libXmu
libpng
libsamplerate
liblzo2
libid3tag
libogg
MACDll
libXinerama
libfaac
libXt
libHAL
UnrarXLib
Python
TinyXML
libvorbis
libhts
wavpack
libfribidi
libvorbisfile
libGLEW
libhdhomerun
libNoseFart
libsqlite
libasound
libavahi
libmpcdec
libjasper
libRTMP
libfaad2
SDL-mixer
libexif
timidity
libass
FFmpeg
flac
libbz2
liba52
libXBMS
libfreetype
SDL-image
libRTV
SDL
libXDAAP
libvorbisenc
libpcre
libmpeg2
foo_gym
libXrandR
libcdio
libmysqlclient
cximage
libjpeg
libfontconfig
zlib
libcurl
libdca
SNESAPU
libGLU

XBMC on Xbox: 2003-2007

- Success due to openness and community
- Designer/skinners/scripters pushing development forward
- Xboxes become cheaper and easier to hack - more users
- Hard drives are cheaper - much more focus on video
- Streaming media becomes usable – add-on directory services are developed

Becoming platform agnostic

- Xbox was too slow to do HD, which was fast becoming a standard
- Early 2007, Yuval Tal starts looking at a Linux port
- Uses SDL/OpenGL for input, graphics, etc
- Initial port up and running in May 2007, usable in June
- Tons of work to emulate Win32 API
- Windows SDL/OpenGL port in late 2007
- OSX port in late 2007/early 2008
- Nov 14th, 2008, XBMC 8.10 (Atlantis) is released on Linux, OSX and Windows

2008/2009 - Growing up

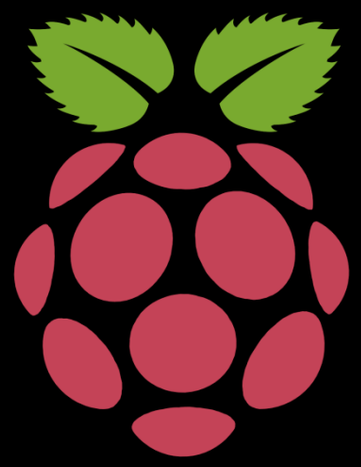
- The ports meant the development team grew, as did the community
- We realized we had to start looking towards the future
- Donations were still being held in a personal paypal account
- What happens if key members leave?
- Companies want to do business with us, how do we ensure XBMC stays independent?
- The XBMC Foundation was born

XBMC Foundation

- Non-profit in the US
- Lawyer costs sponsored by Boxee initially
- Difficult due to no board members being in the US
- It cost a LOT of money to setup and took a LOT of time

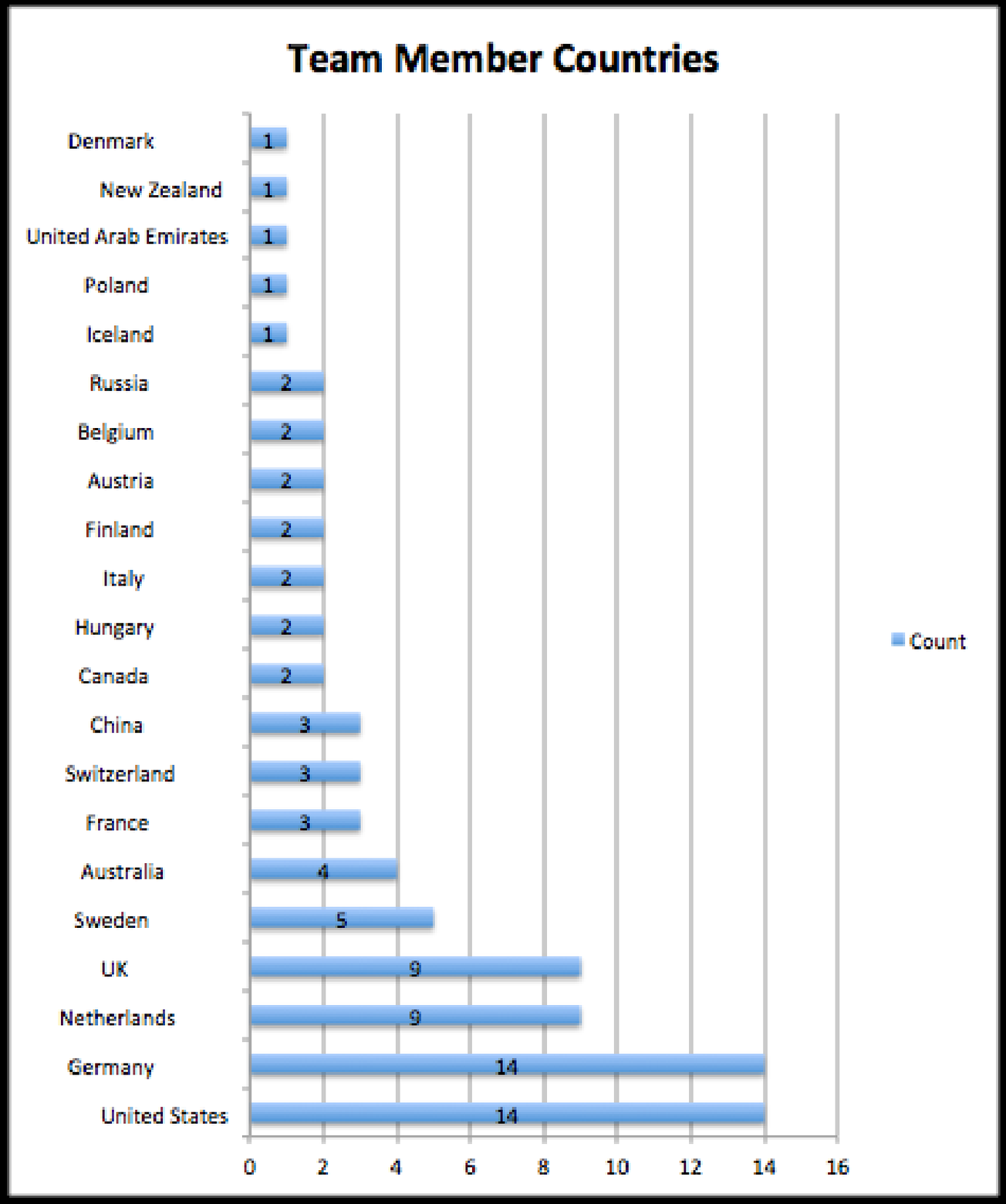
XBMC-ARM: Next chapter

- Oct 2009 - OpenGL ES 2.0 was demo'd by team member McGeagh, on a BeagleBoard
- Jan 2011 - After much hard work by Scott Davilla and others, XBMC on iOS was officially released!
- Jan 2012 - At SCALE 10x, XBMC brought a Raspberry PI running native XBMC. Much thanks to Edgar Hucek (gimli) and Davilla's hard work getting this up and running!
- July 2012 - Heavily sponsored by Pivos, XBMC for Android was officially released
- Jan 2013 - XBMC 12.0 (Frodo) was released, bringing together all the platforms, but Android was still very rough
- May 2014 - XBMC 13.0 (Gotham) was released, bringing full parity across all platforms
- December 2014 – Kodi 14.0 (Helix) was released.



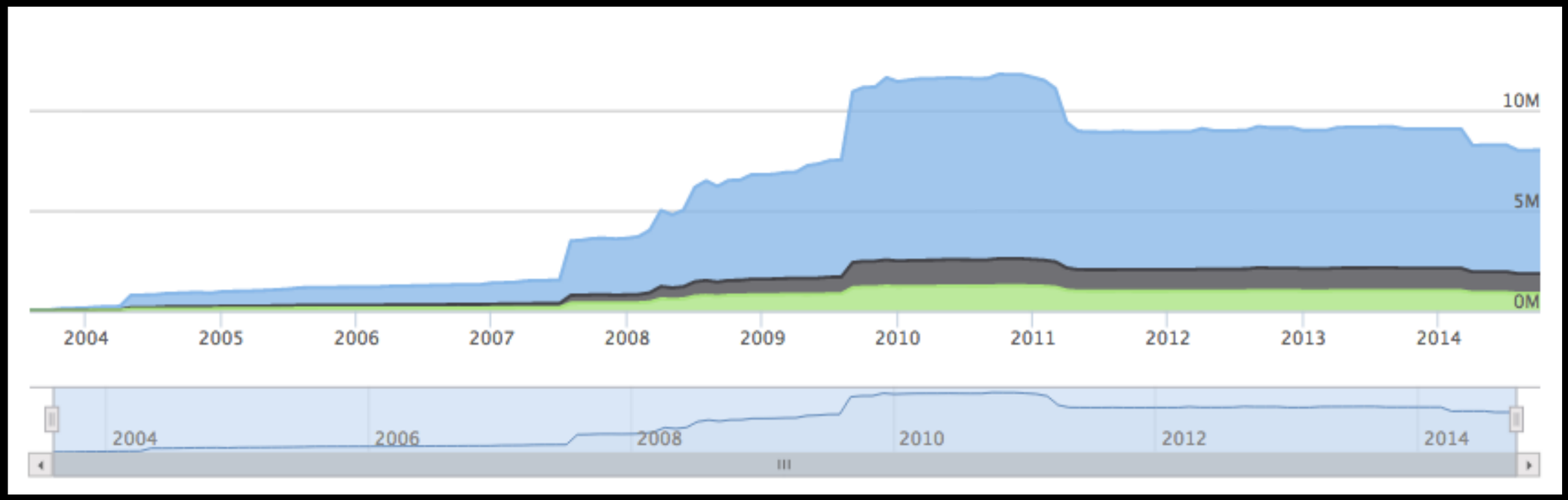
What we failed on

- Dual Licensing: Without signing over the rights of the source to the Foundation, we would have to contact every single developer that's contributed
- Owning our brand: Did not focus on trademarking, anyone remember Mozilla Phoenix/Firebird?
- Backend server: Made great strides in embedding server components into our client, but no focus on a headless build (until now!)
- Communication: Biz side of running a foundation is hard, even in our team the opinion we don’t communicate enough internally is well known
- Non user friendly: We have not done the best encouraging new users by making it easy to start using, but this is slowly changing
- Information is not organized: Between a wiki, forum, trac and github, which is the most up to date (if at all)?



83 Team Members (still increasing)

Lines of Code since 2003



- Code Lines : 6,137,770
- Total Comment Lines : 981,744
- Total Blank Lines : 869,629
- Percent Code Lines : 76.8%
- Percent Comment Lines : 12.3%
- Percent Blank Lines : 10.9%

Codebase declines due to cleaning up code, making it easier to maintain and outsourcing to add-ons.

Estimated Cost

- Codebase Size: 6,137,679 lines
- Estimated Effort:1836 person-years
- Estimated Cost: $100,983,043*

*Using the Basic COCOMO Model at $55k/yr average salary

Taken from OpenHUB

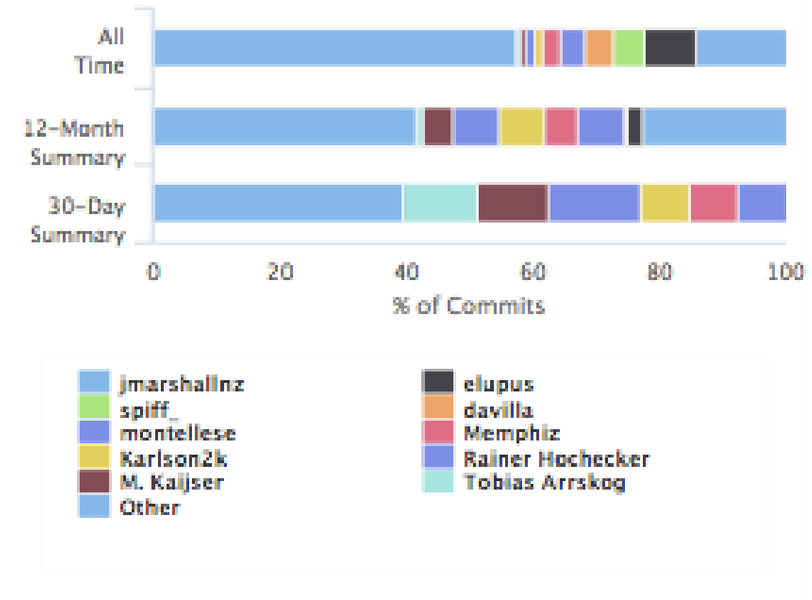
Language Breakdown

Language	Code Lines	Comment Lines	Comment Ratio	Blank Lines	Total Lines	Total Percentage
C	2,804,298	488,538	14.8%	390,068	3,682,904	46.1%
C++	1,269,270	318,655	20.1%	223,176	1,811,101	22.7%
Autoconf	614,840	4,661	0.8%	79,278	698,779	8.7%
XML	426,569	5,851	1.4%	4,810	437,230	5.5%
Python	289,374	77,632	21.2%	60,186	427,192	5.3%
shell script	259,274	38,062	12.8%	28,385	325,721	4.1%
Make	154,716	9,139	5.6%	20,899	184,754	2.3%
TeX/LaTeX	100,133	6,587	6.2%	22,087	128,807	1.6%
HTML	63,988	551	0.9%	6,943	71,482	0.9%
Assembly	49,526	9,770	16.5%	10,812	70,108	0.9%
Objective-C	15,524	4,218	21.4%	3,354	23,096	0.3%
Pascal	13,481	1,540	10.3%	5,250	20,271	0.3%
Perl	13,116	2,843	17.8%	2,397	18,356	0.2%
Modula-2	7,966	0	0.0%	914	8,880	0.1%
Automake	7,888	3,026	27.7%	2,222	13,136	0.2%
C#	7,789	2,905	27.2%	1,577	12,271	0.2%
CSS	7,115	187	2.6%	1,405	8,707	0.1%
Ada	5,043	1,680	25.0%	1,797	8,520	0.1%
Emacs Lisp	4,784	531	10.0%	667	5,982	0.1%
Java	3,328	1,431	30.1%	500	5,259	0.1%
D	2,980	21	0.7%	226	3,227	0.0%
JavaScript	2,827	398	12.3%	290	3,515	0.0%
Ruby	2,741	765	21.8%	568	4,074	0.1%
DOS batch script	2,560	260	9.2%	615	3,435	0.0%
Tcl	2,044	510	20.0%	304	2,858	0.0%
CMake	1,365	517	27.5%	318	2,200	0.0%
DCL	1,335	342	20.4%	39	1,716	0.0%
Groovy	1,059	456	30.1%	191	1,706	0.0%

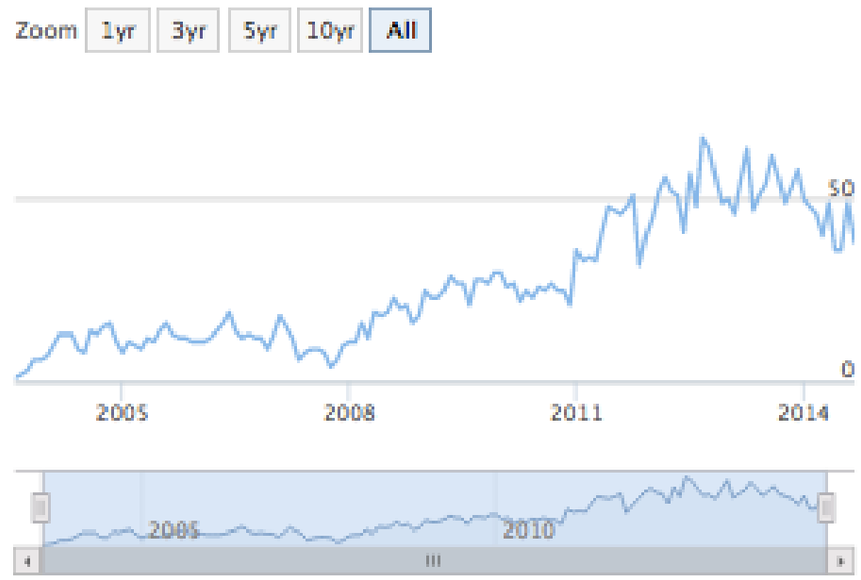
Number of Languages : 38

Contributors

Commits by Top Contributors



Number of Contributors



Top Contributors

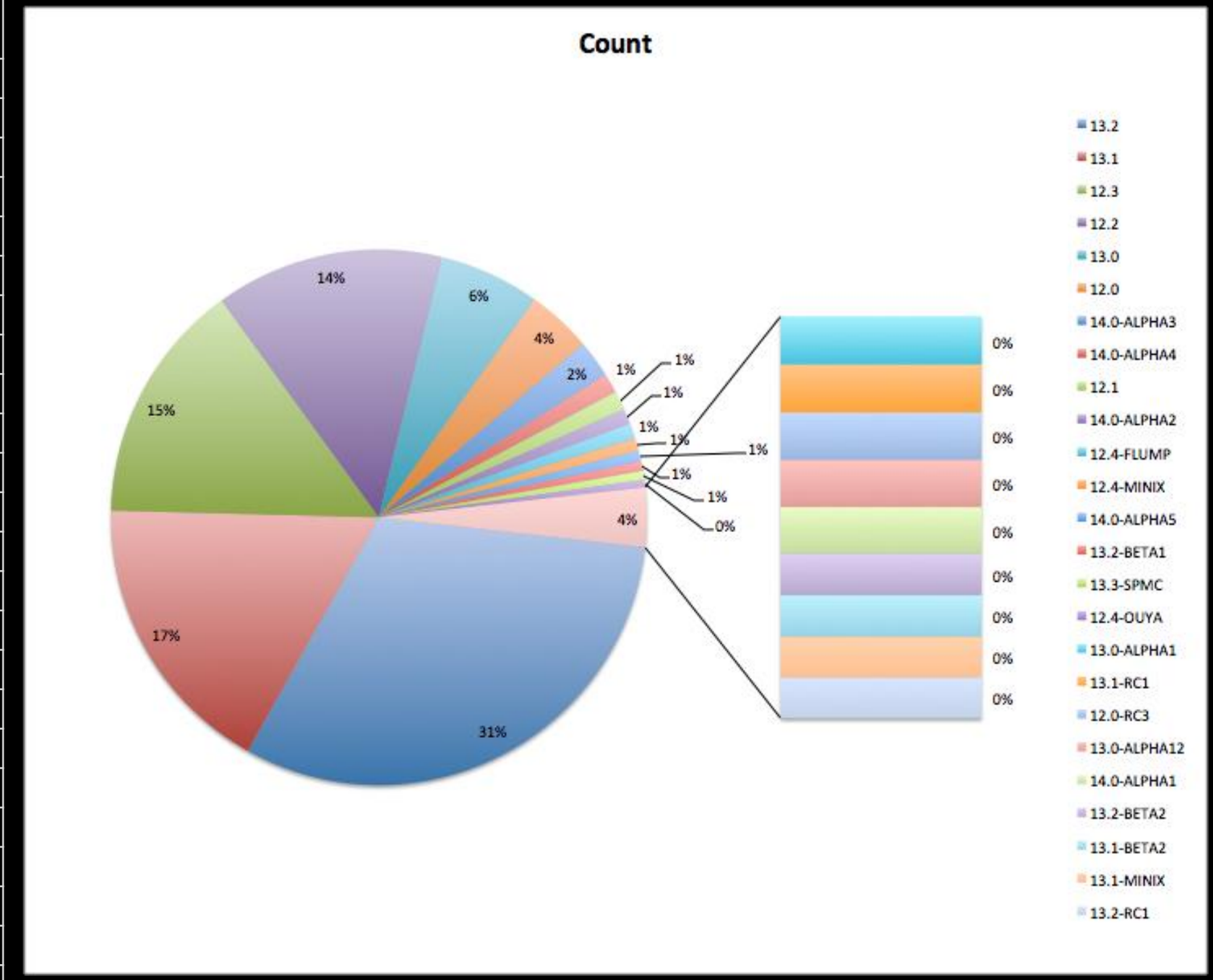
Name	Kudos	12 Month Commits	All Time Commits	5 Year Trend	Primary Language	First Commit	Last Commit
jmarshall (Developer)	9	1205	6828		C++	almost 11 years ago	about 1 month ago
Sascha Montellese (Developer)	9	405	1892		C++	almost 4 years ago	7 days ago
Rainer Hochecker	8	394	594		C++	over 3 years ago	4 days ago
Karlson2k	8	384	695		C++	over 2 years ago	3 days ago
Memphiz	8	266	1174		C++	over 3 years ago	4 days ago
Martijn Kaijser (Manager)	8	248	569		C++	over 3 years ago	2 days ago

Development Cycle

- Code contributions through Github pull requests. Code is reviewed and build on all platforms using Jenkins before merging into master
- Daily build for all platforms for continuity testing by any one who wants
- Merge Window - Merge feature Pull Requests at the beginning of the month, trivial bugfixes can be merged throughout
- Window order are alpha, alpha, beta, release candidate, release
- Use Milestones in Github to track progress
- Only allow API breakage during the alpha windows
- Beta windows allow stuff to be altered outside of API
- Release Candidate window only bugfixes are allowed

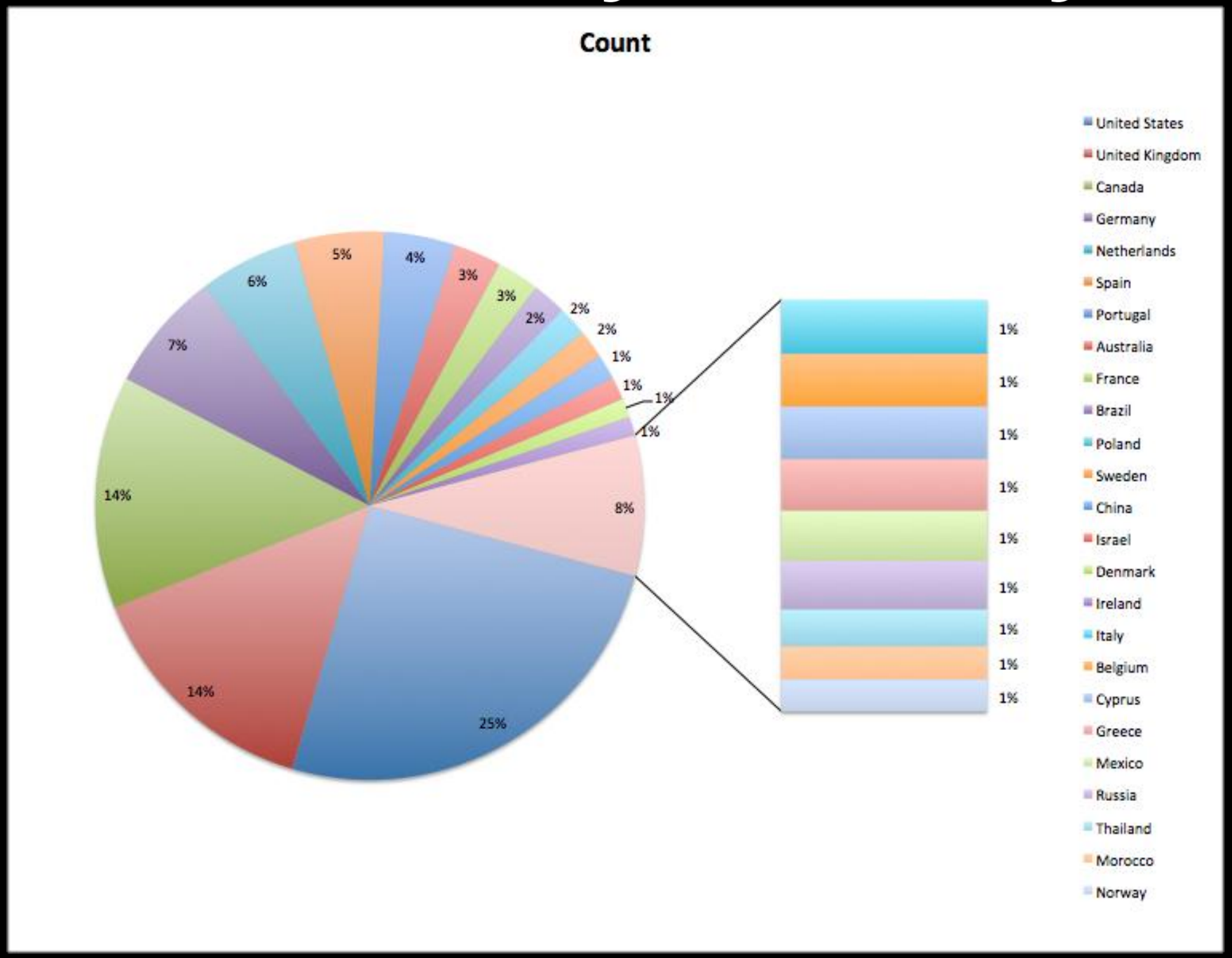
User estimates by version

Version	Count
13.2	2,016,088
13.1	1,109,641
12.3	940,262
12.2	885,499
13.0	392,025
12.0	252,572
14.0-ALPHA3	144,241
14.0-ALPHA4	76,909
12.1	73,318
14.0-ALPHA2	59,552
12.4-FLUMP	58,953
12.4-MINIX	47,881
14.0-ALPHA5	43,093
13.2-BETA1	36,808
13.3-SPMC	35,013
12.4-OUYA	28,429
13.0-ALPHA1	27,532
13.1-RC1	27,532
12.0-RC3	27,232
13.0-ALPHA12	26,933
14.0-ALPHA1	26,634
13.2-BETA2	23,940
13.1-BETA2	23,641
13.1-MINIX	23,342
13.2-RC1	23,342



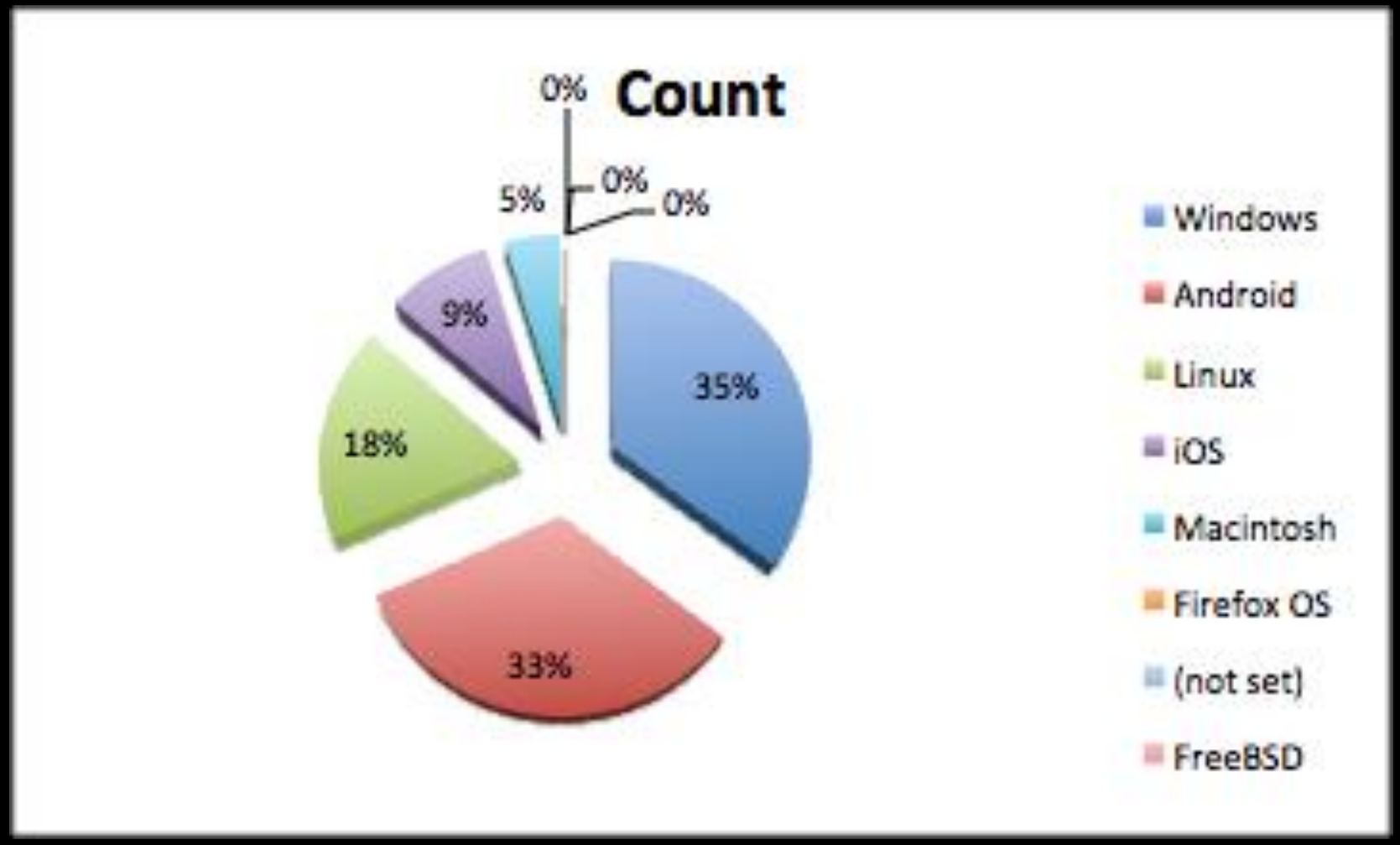
User estimates by Country

Location	Count
United States	1,471,741
United Kingdom	832,231
Canada	793,627
Germany	410,280
Netherlands	336,064
Spain	305,840
Portugal	242,697
Australia	162,197
France	146,635
Brazil	115,812
Poland	100,849
Sweden	97,258
China	85,886
Israel	78,405
Denmark	64,939
Ireland	63,742
Italy	62,545
Belgium	61,347
Cyprus	61,048
Greece	59,851
Mexico	58,355
Russia	56,260
Thailand	42,494
Morocco	38,604
Norway	37,108



User estimates by platform

Platform	Count
Windows	2,361,429
Android	2,194,444
Linux	1,211,987
iOS	589,534
Macintosh	305,540
Firefox OS	2,693
(not set)	599
FreeBSD	599



Android

- A single main developer!
  Chris Browet (“*Koying*“) is amazing, but he could use help

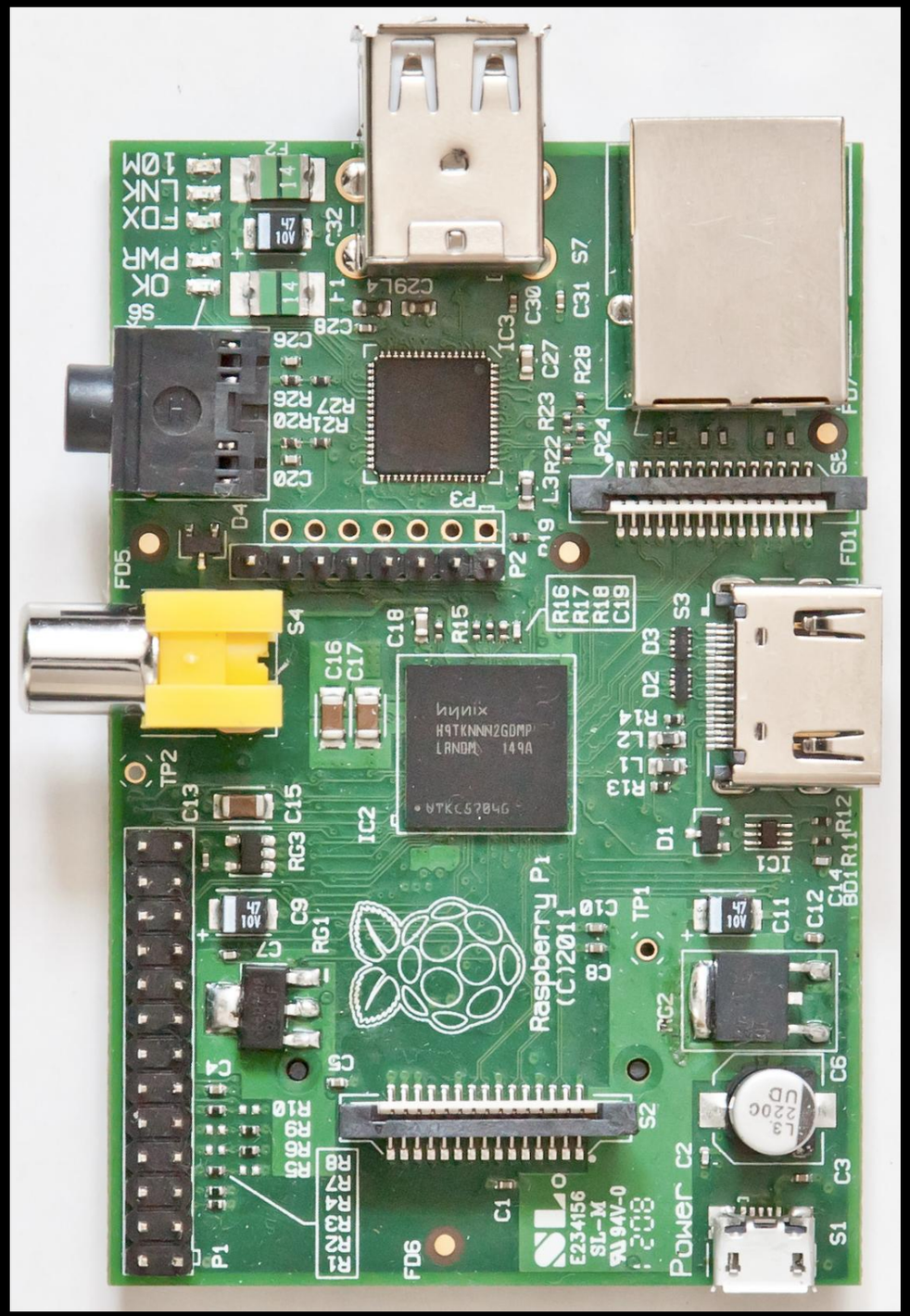
- Fastest growing user base of any platform (in only two years time): 33%

- Most diverse set of hardware and (very)slowly maturing API’s

- Inconsistent feature sets (hardware decoding, audio passthrough, etc).

Raspberry PI

- Lowest barrier of entry for a dedicated Kodi device
- Game changer, especially in the entertainment world
- In spite of the devices low power, it’s one of the most used platforms
- Lucky enough to have one of the main R-Pi guys on our team, contributing optimizations and bug fixes
- Helped greatly in improving our codebase in general for all platforms due to low power

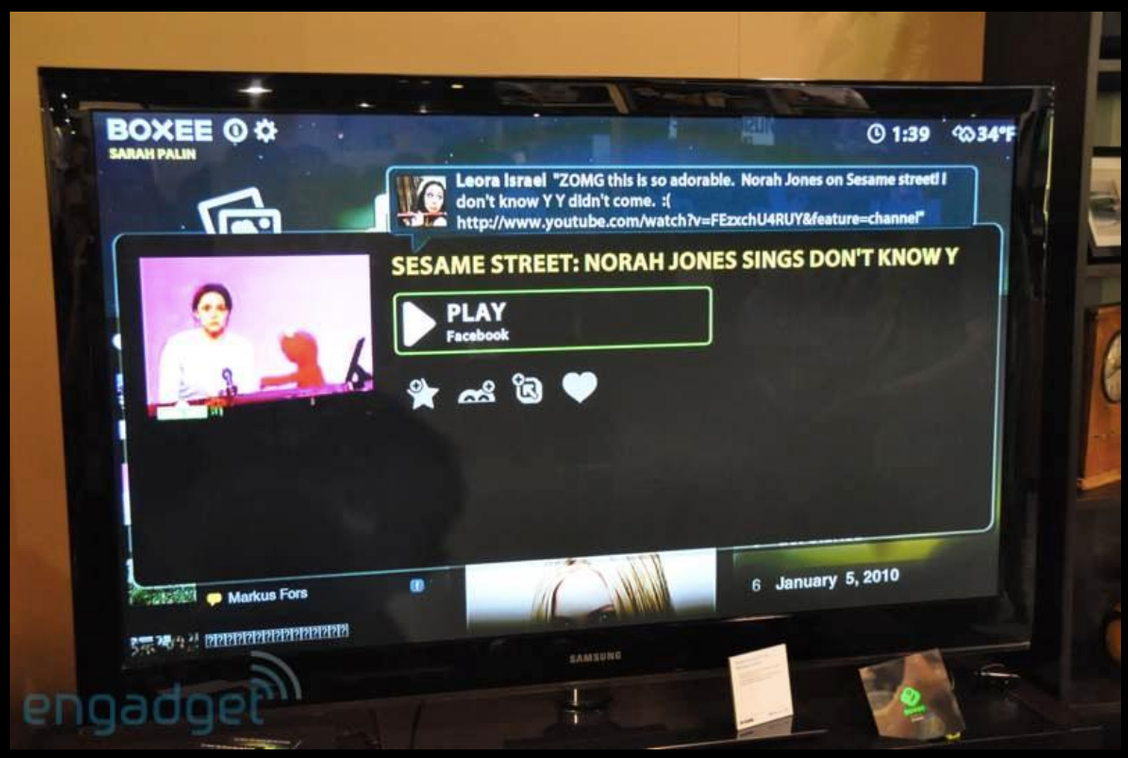


OpenELEC

- Kodi is the centerpiece
- Designed to be extremely small and fast booting
- Live CD style, can be installed or ran from flash
- Simple install and reduces complexity as much as possible for the end user
- Supports both ARM and x86

Commercial Port - Boxee

- Received $26.5 Million in funding
- Initially wanted to combine XBMC with social networking
- Goal was to commercialize XBMC while giving back
- Sponsored first XBMC Developer Conference in 2008
- Combined multiple sources of media in an innovative manner
- Created the Boxee Box, a Intel CE4100-based hardware device w/ a QWERTY keyboard remote
- In July 2013 acquired by Samsung for $30 Million



Commercial Port - Plex

- For profit corporation started by a couple ex team members
- Ecosystem consists of 3 types of software:
  - Media Server, a closed source application that runs on all platforms
  - Plex Home Theater, an open source application based off XBMC 12.0 and runs on Windows and OSX
  - Plex thin clients, such as iOS, Android, Roku, Samsung and other devices, utilizing the server to do transcoding if the end device does not support the format
- Freemium model, where users can pay for Plexpass subscription which allows them early access to builds and new features that aren’t rolled out yet to the public

Kodi’s Future

- Binary add-ons:
  Extend existing add-on functionality to include compiled C/C++ code that would not have to be compiled or shipped with the rest of the application. These can also be closed source

- Media Importing:
  Method to define media providers, currently using UPnP but can be extended and have information shared across instances

- Content intergration:
  Total integration of any installed content providing plug-in / add-on into a single media library instead of separate browsing

Kodi’s Future

- Retroplayer:
  Built around RetroArch, allows you to play emulated ROM’s. Features saving, rewind and start playing again from the point you choose. One single library of all games.

- Kodi Server:
  Headless instance of Kodi, controlled through JSON-RPC and webinterface. Intended to be central media hub where other Kodi clients get their data (library) and media from.

- Audio DSP processing add-ons:
  Hook in DSP related add-ons into our Audio Engine. Bringing EQ and FreeSurround like features

How you can contribute

- We need developers!
- Skinners, testers, python developers, documentation, anyone can contribute!
- Did I mention Developers and testers?
- Android, iOS and OSX all only have a single developer currently

Questions?