

# ARM Chromebookで Linuxを色々と遊んでみよう

- 1、自己紹介
- 2、Chromebookの紹介、全体像
- 3、Chromebookのカスタマイズ
  - ・Chromeストアと主なアプリ
  - ・Androidアプリ Chrome-apkとARCon
- 4、ChromebookでOS載せ替えテスト
  - ・Crouton
  - ・Chrubuntu
  - ・Linuxディストリビューション、FreeBSD
  - ・Windows(参考 x86のみ:Acer C720の報告事例抜粋)
  - ・MacOSX(参考 x86のみ:Acer C720の報告事例抜粋)
- 5、追加カスタマイズ
  - ・Linuxカーネル
  - ・ChromiumOS
  - ・PKGSRC
  - ・エミュレータ動作テスト
- 6、次回出展

講演資料は電子データと紙面を配布しております。詳しくはブースまで。

講演：Kapper

OSC東京Spring 2015/2/28 10:00~  
会場：明星大学 102室

# 自己紹介



- 名前:Kapper
- Twitterアカウント:@kapper1224
- HP:<http://kapper1224.sakura.ne.jp>
- 趣味:Linux、モバイル好きなARMマニア
- 好きな言葉:実験より記録重視。

出来た事、ダメだった事は貴重な財産で記録に残そう。

- 最近の実験機:Netwalker(PC-Z1,T1)、Nokia N900、DynabookAZ、RaspberryPi  
Nexus7(2012、2013)、Hercules eCAFE EX HD、Jetson TK-1、  
OpenPandora、ARM Chromebook、ZTE OPEN C (FirefoxOS)  
Chromecast、台湾Android電子辞書 無敵CD-920、CD-928他
- 最近気になる事:NetwalkerでLinux from Scratchのテスト開始しました。  
台湾Android電子辞書の勉強と解析中です。面白いです。  
Chromebookを色々と遊んでいます。

ARMデバイスに関しては  
超マニアです。  
大好きです。

よろしく!

# Chromebookの長所

- 1.安い(ただし無料Windows型と同等?)
- 2.速い。特にブラウザ高速
- 3.ローリングリリース長期サポート
- 4.Googleドライブ100GB無料



- ・chrootでLinuxデベロップメント動作
- ・USB、SDブートも可能
- ・Androidのアプリも移植中
- ・ハッキングしやすい遊べる

## Linuxマシン

Linuxカーネル、Xorgドライバ、ブートローダー、GPUなど全部メーカー側でサポートしてくれて、chrootスクリプトやインストーラーまでGoogle開発者が用意してくれてる  
改造するのに大変楽な素晴らしいマシン

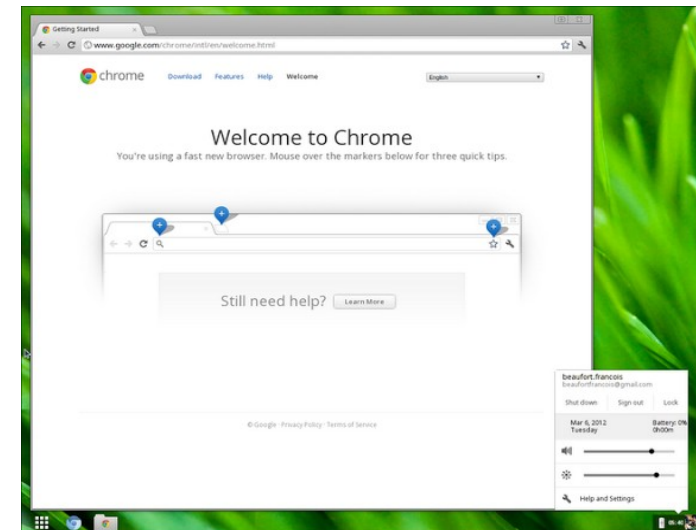
Chromebook



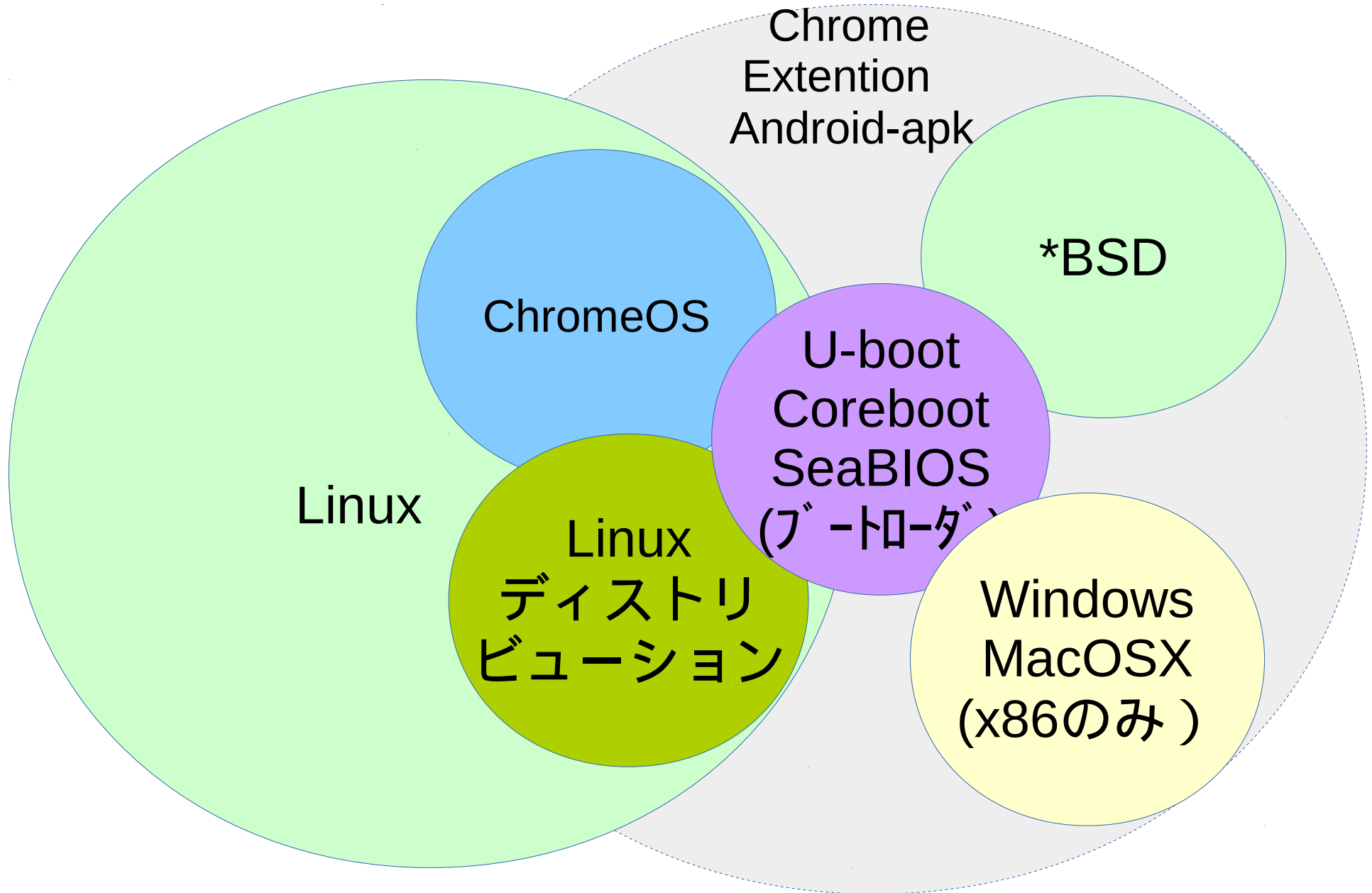
# ChromeOSとは

- 2009年11月Chromium OS公開。  
2010年12月プロタイプ「Cr-48」  
2011年6月Chromebook「Series 5」発売  
2014年11月日本発売開始
- パッケージマネージャ: **Portage**  
インターフェイス: Google Chrome ブラウザ  
アプリ: Chrome Extension, Flash, HTML5, JavaScript  
リリース: ローリングリリース  
ウィンドウマネージャ: Ash-WindowManager  
Aura-hardware accelerated UI framework  
ブートローダー: ARM **U-boot**  
x86 Coreboot ⇒ **U-boot** (ペイロード)  
⇒ **SeaBIOS** (対応機種のみ)
- ChromiumOS (OSS) との違い  
ファームウェアの追加: verified boot と easy recoveryモード  
**ハードウェア最適化**、セキュリティ対策  
**ローリング・リリース**によるオートアップデート  
**Googleとパートナー、ベンダーサポート**  
バイナリパッケージの追加
  - **Adobe Flash**
  - Netflix Instant
  - Google Talk
  - プラグインなしでのWebRTC
  - 3G対応 (対応機種のみ)
  - Google ChromeOS 純正ロゴマーク

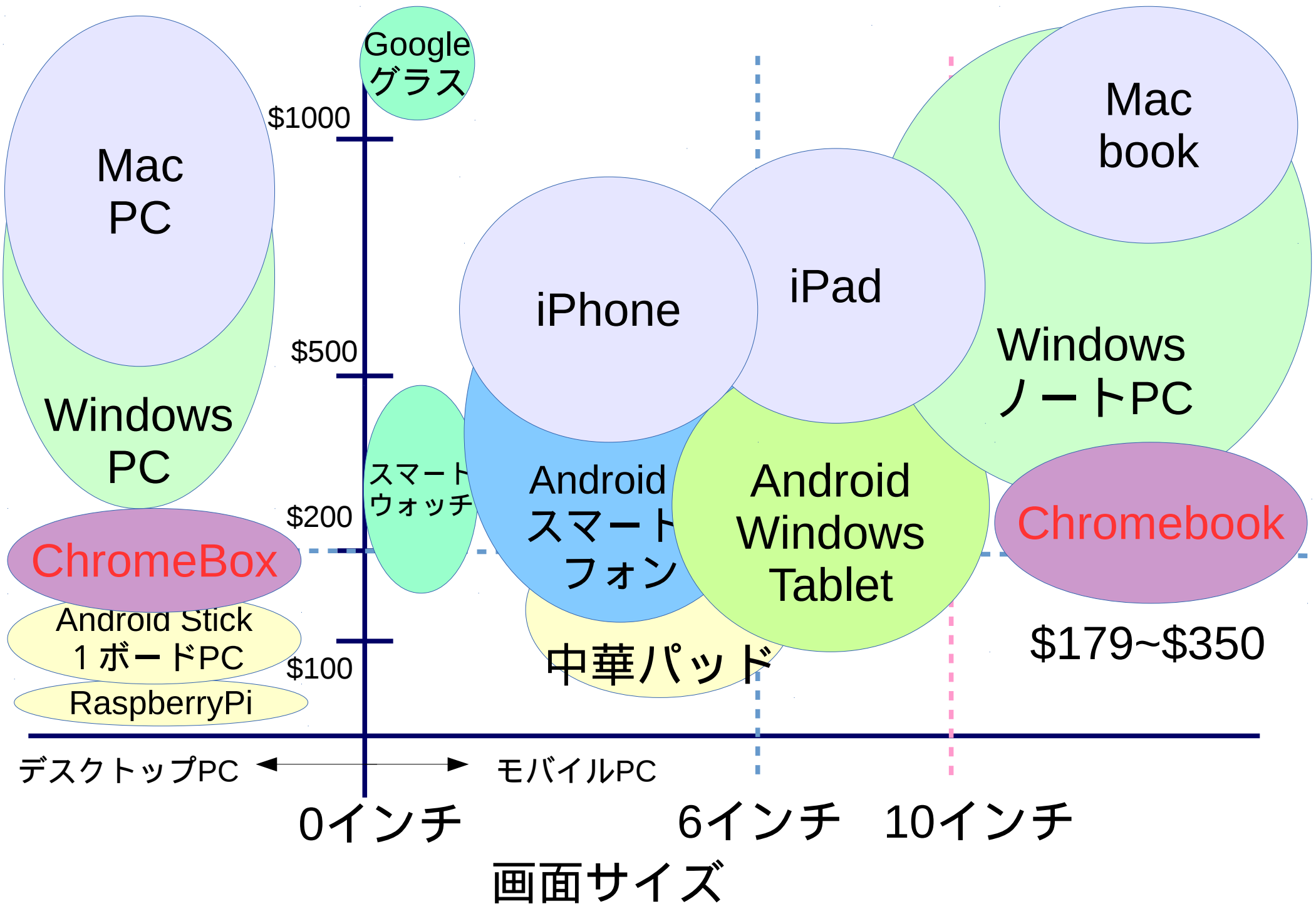
Ash WindowManager



# Chromebookの全体像



# PC市場の中のChromebook



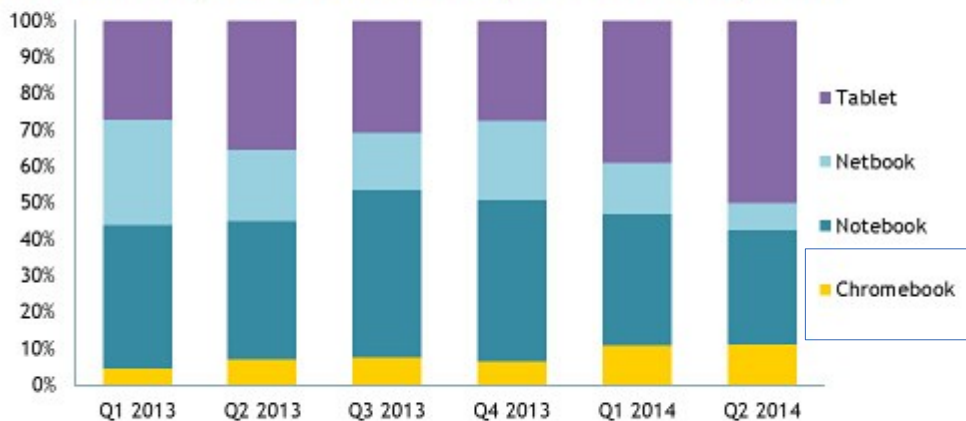
# マーケティング・リサーチ

- Chromebookの市場は多数がUSAとEU。2014年はMSの無償Win8と競争激化。教育部門ではiPadとの競争に優勢でシェアUP。

## TABLET REACHES OVER 50% OF VOLUMES IN QTR 2 2014

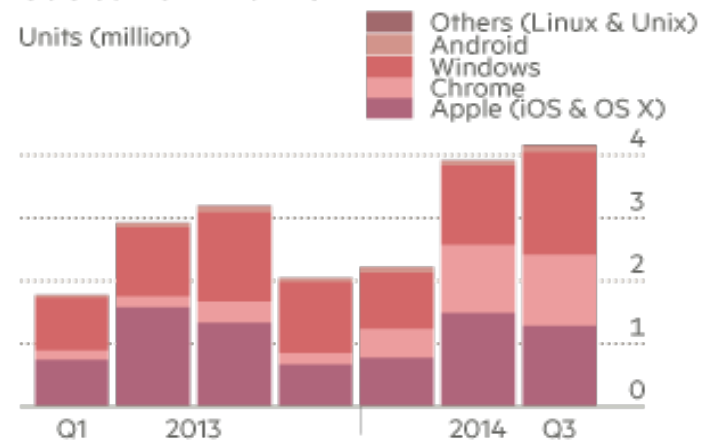
Large Scale Deployments In Turkey and Thailand Aid Tablet Sales. Chrome Continuing To Gain Share But Vast Majority Of Sales Currently In The US.

Global PC/Tablet Shipments To K-12 Education Sector, Does Not Include 'Bring Your Own'

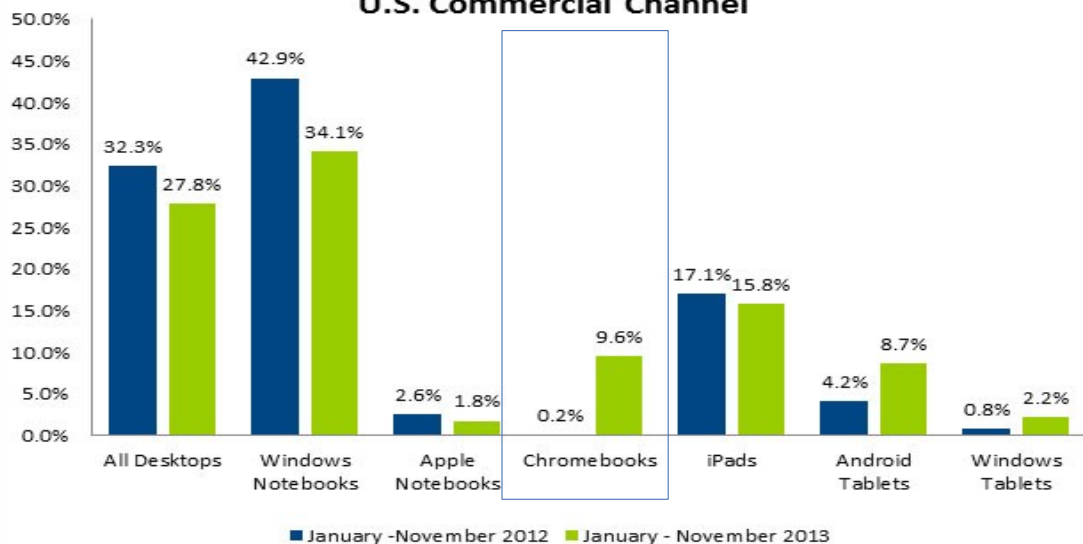


## PC and tablet sales in US education market\*

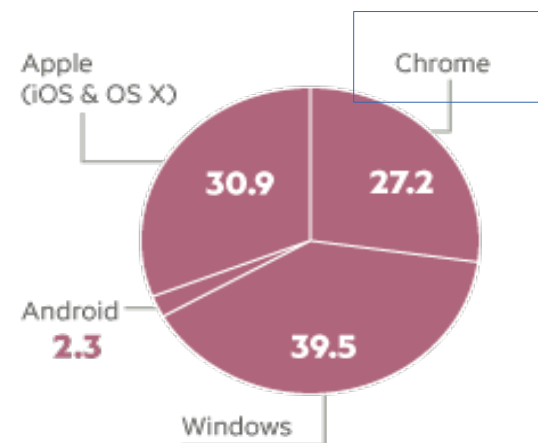
Units (million)



## Share of Unit Sales U.S. Commercial Channel



2014 Q3 (% of total)



Source: IDC \*Schools and colleges



# 英語圏で売れてたChromeOSモデル



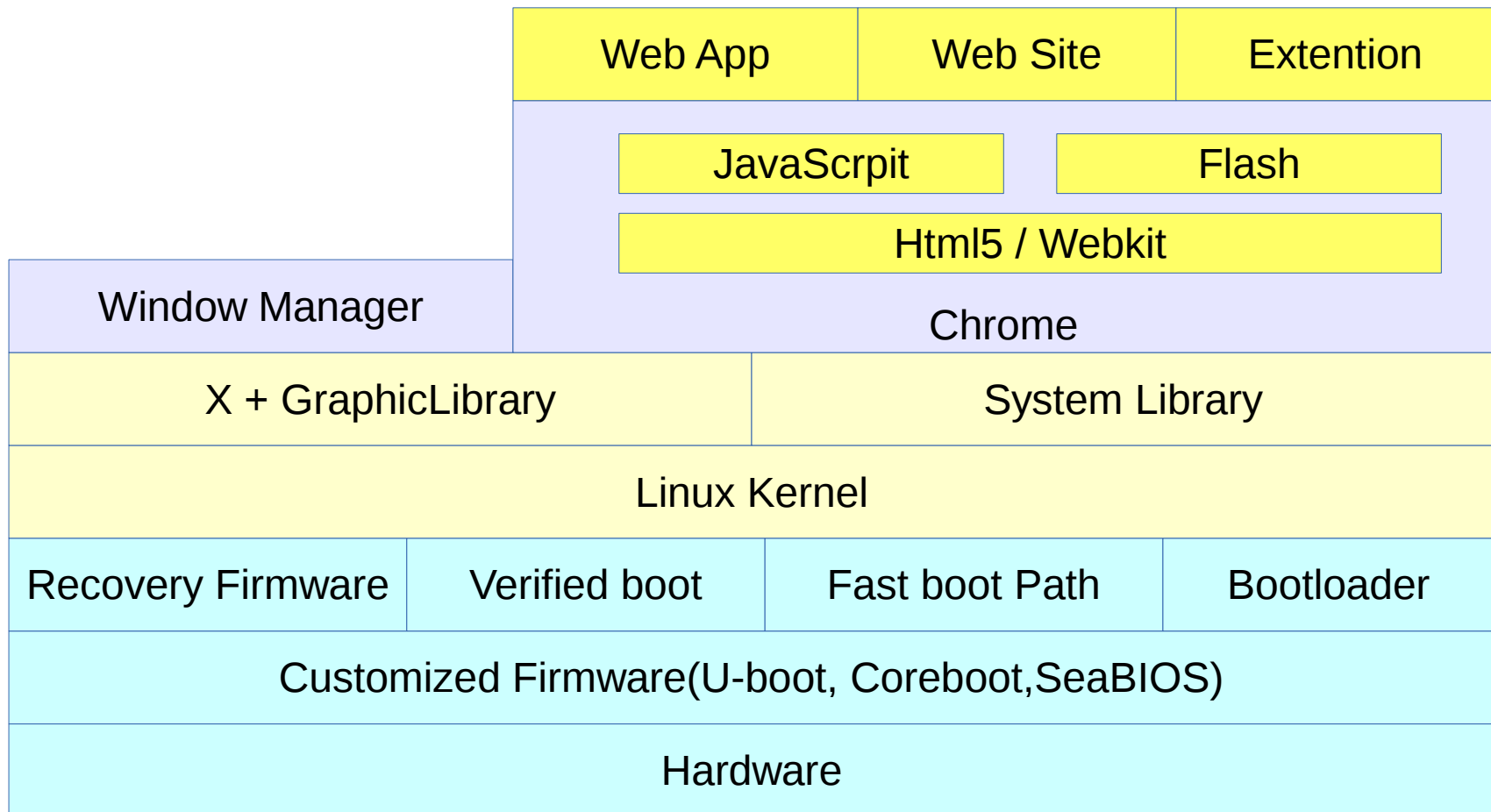
- Acer Chromebook C720  
2013年10月発売のChromebookの代表的存在。  
\$199 ノートPCとして現在でもNo1の売上を誇るx86。  
レガシーブートモードを持ち、SeaBIOSが使えるのでWindowsやMacOSXが起動出来ると評判に。改造機種の名詞として、幅広いユーザー層を持つ名機。

- Samsung Series 3  
2012年10月発売Exynos 5250搭載の初ARM Chromebook。  
ARMにしては当時最新鋭でありUltrabookスタイルで\$249と非常に安価な機種。iPadの半額として有名になった。  
カスタマイズしやすい設計で非常に遊びやすいARM初期モデル。  
Raspberry PiやNetwalkerの4倍/1コア程度。

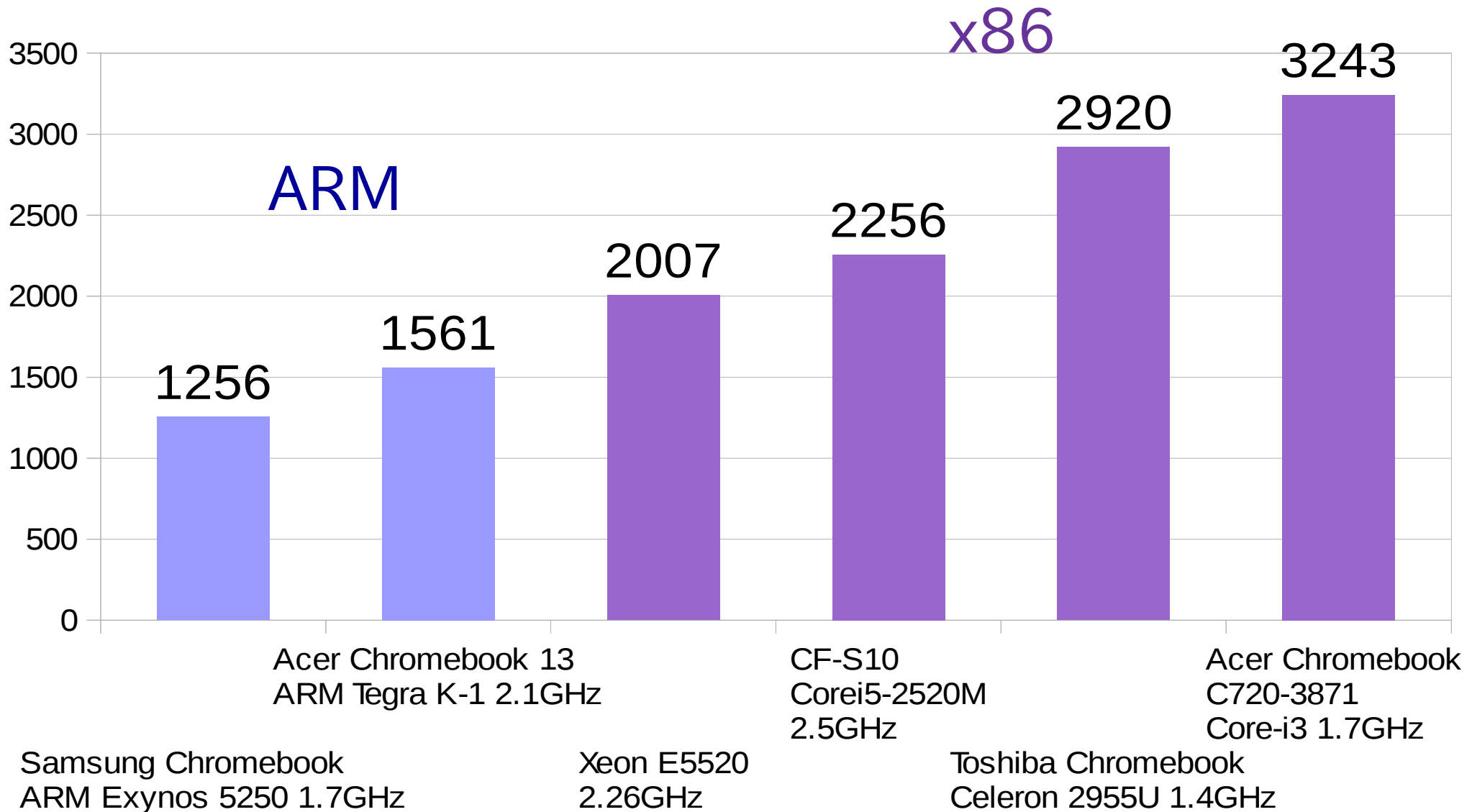
- Asus Chromebox-M004U  
2014年3月にリリースしたAsusの\$179格安Chromebox。  
Chromebookより一回り安い値段設定。  
通常のWindows PCの半値近い価格設定でCeleron、Core i5を搭載した。中古値段を考慮すると非常に安く、小型Linux BOXとしても有望。

# ChromeOSの仕組み

- ファームウェアから4つのブートローダがあり、切り替えてKernelを起動する。XからSingle x-clientとしてChromeを使う構成



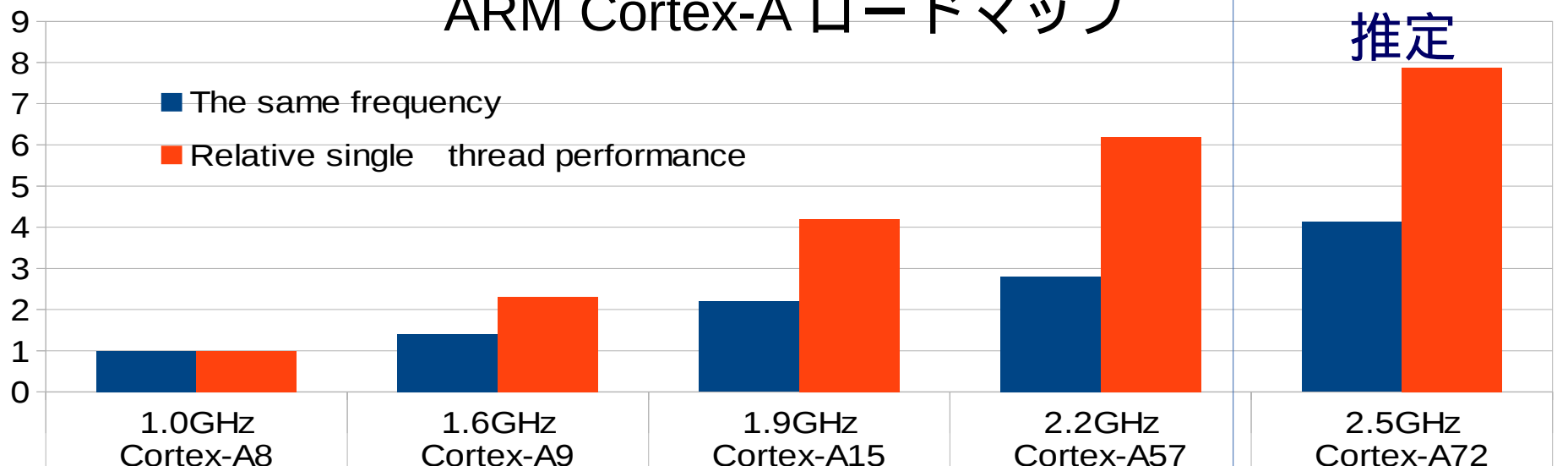
# PeaceKeeperBrowserベンチ比較



- ブラウザの速度面ではARMの優位性は無かったりまだx86の方が速いです。

# ARM端末のベンチマークスコア

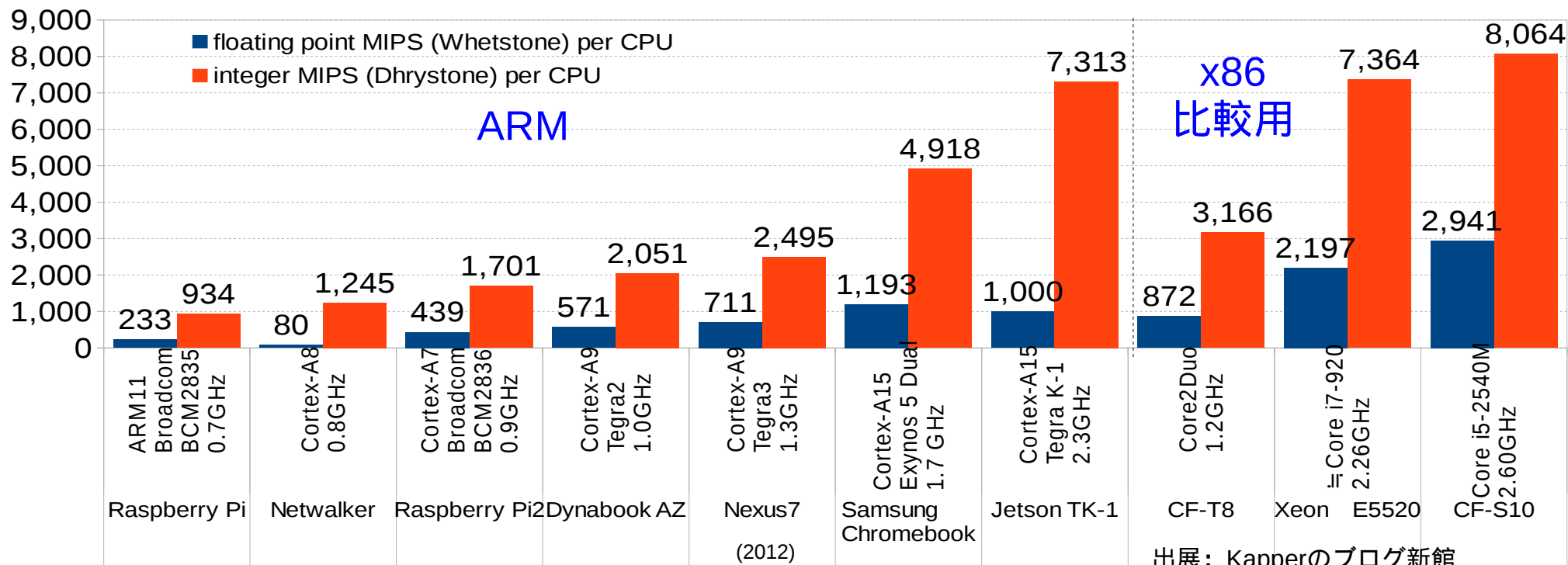
## ARM Cortex-A ロードマップ



出展: A Walk Through the Cortex-A Mobile Roadmap

<http://community.arm.com/groups/processors/blog/2013/11/19/a-walk-through-the-cortex-a-mobile-roadmap>

## ARM端末 BOINCベンチマーク



出展: Kapperのブログ新館

# ChromeOSサポート期間

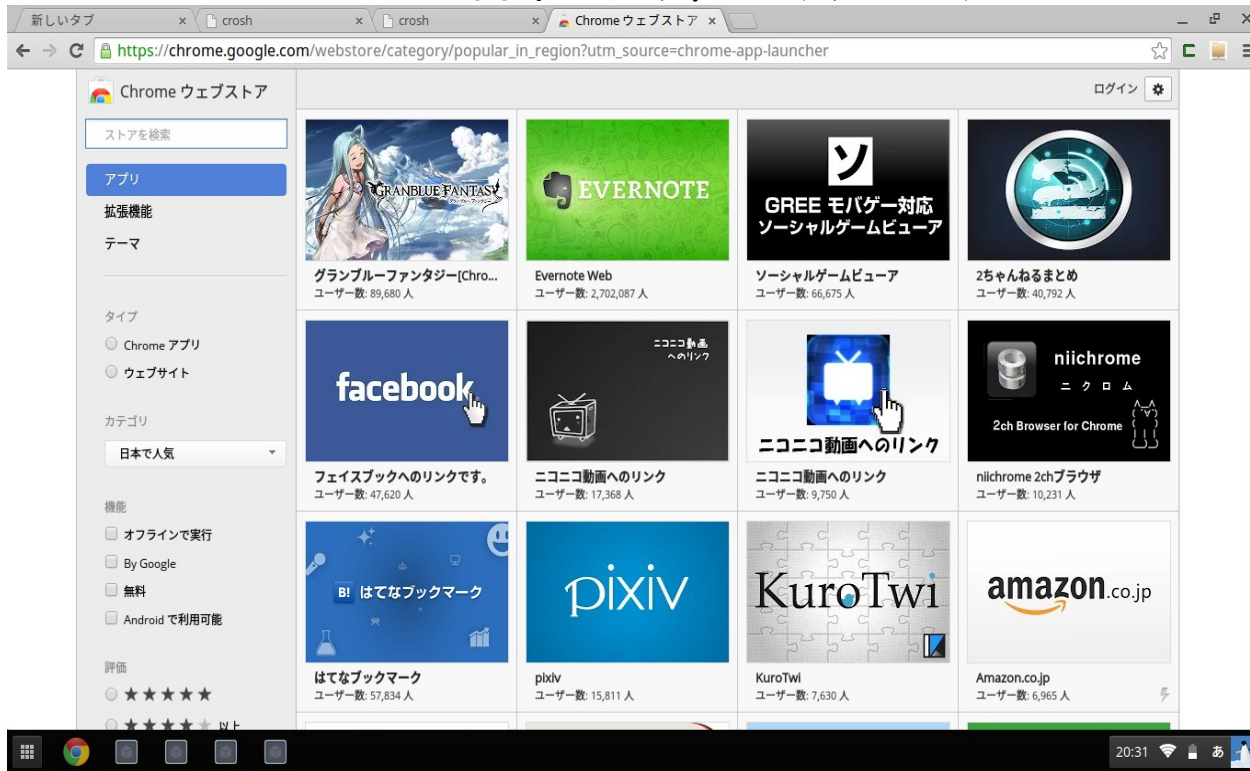
- メーカー情報より抜粋。実質5年サポート
- サポートアウト後はアップデートがこないなので、GentooやDebian、UbuntuLTSなど乗換え必要
- KernelがMainlineに登録されるが将来の鍵

メーカー	機種名	発売日	サポート終了期間	実質サポート期間
Acer	AC700	11年07月	16年08月	05年02月
	C720-2800	13年11月	18年11月	04年12月
Asus	Chromebox	14年06月	19年03月	04年09月
Dell	Chromebook 11	14年04月	19年01月	04年10月
Google	Chromebook Pixel	13年02月	18年04月	05年02月
	Cr-48	10年12月	15年12月	04年12月
HP	Chromebook 11	13年10月	18年10月	04年12月
	Chromebook 14	13年11月	18年11月	04年12月
	Chromebook Pavilion	13年02月	18年02月	04年12月
Lenovo	x131e	13年02月	18年02月	04年12月
Samsung	Series3 Chromebook(XE303C12)	12年10月	17年10月	04年12月
	Series3 Chromebook(XE300M22)	12年12月	18年03月	05年03月
	Series5 (XE500C21)	11年06月	16年06月	04年12月
	Series5 550(XE550C22)	12年05月	17年05月	04年12月
Toshiba	Chromebook	14年01月	19年02月	05年01月

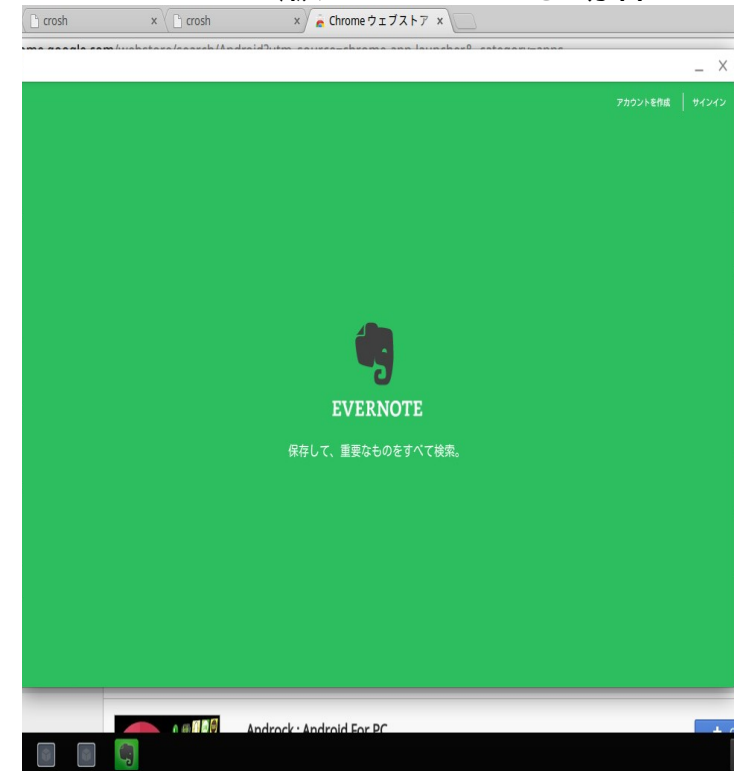
# Chromeストアと主なアプリ

- ChromeOSの基本的な使い方はChromeストアからWebリンクとアプリをダウンロードして使う
- TwitterやFacebook、ニコニコ動画やYoutube、OfficeやPDF、メーカー、RDPまでひと通り対応。
- Androidアプリも移植始まっておりVine、Duolingo、EverNote、Kids Sight Words、Cookpad Recipes、Couchsurfing、HomeAway、PackPointなどの**32つが正式対応**  
**196のアプリとWebリンクがAndroidと共有**出来る形で公開。(15年2月時点)
- **GooglePlayにも対応**しており、Android同様に動画再生などもできる。

## Chromeストア:日本で人気アプリ、Webリンク



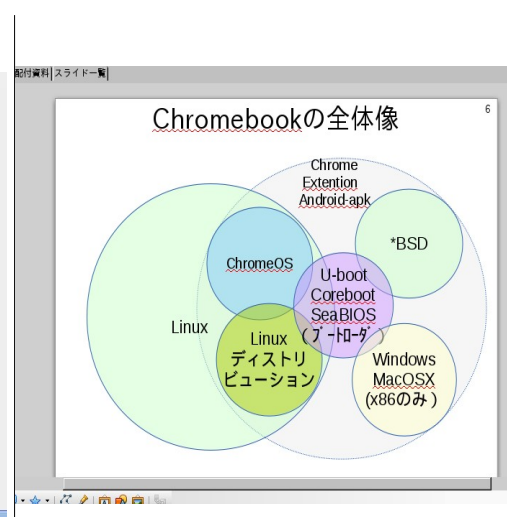
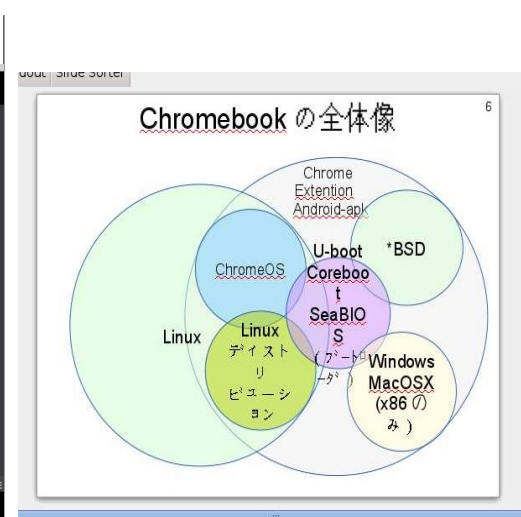
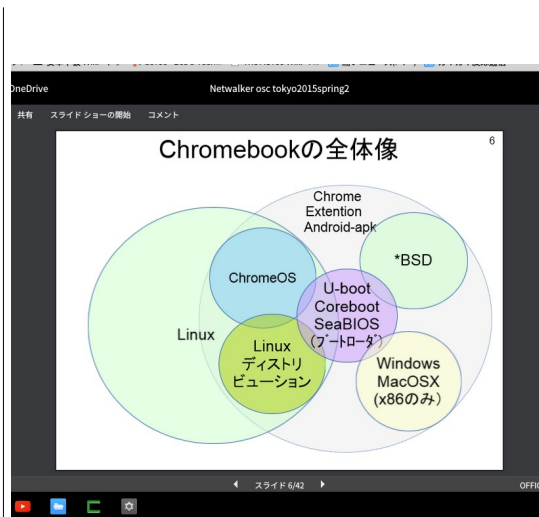
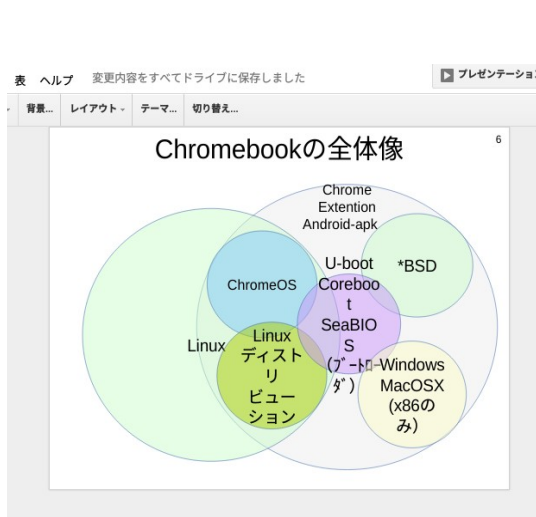
## Android版Evernoteも動作





# ChromeOS上での互換Office比較

- ChromeOS上で動作するOffice環境を比較。  
ChromeOSでフォントの追加修正難しいため、どうしてもフォントズレが発生するCroutonでインストールしたUbuntu、DebianなどのLinuxディストリの場合、LibreOfficeでフォントを追加できるので調整可能。お勧め。



○Googleスライド  
Google標準搭載Webオフィス  
odfやpptが表示、編集可能。  
完成度は高いがChromeOSに  
標準搭載されているフォント  
数が少ないので、他の環境で  
作ったデータはフォントが変  
換されてズレる。  
フォント環境はChromeOSの  
標準環境で追加できないので  
課題。

○MS Office Web App  
MSがサポートしている  
Web Office。pptファイル  
のみ。  
完成度が高く、フォントズ  
レは少ない。僅かにズレは  
発生するので調整修正が必  
要。

△rollAPP Web  
LibreOffice3.3  
RollAPPが供給している  
Web版OSSアプリ。  
LibreOfficeも対応している  
が、英語版なので文字がズ  
レる。フォントが少ない。  
動作もかなり重い。  
今後の多言語化対応に  
期待。

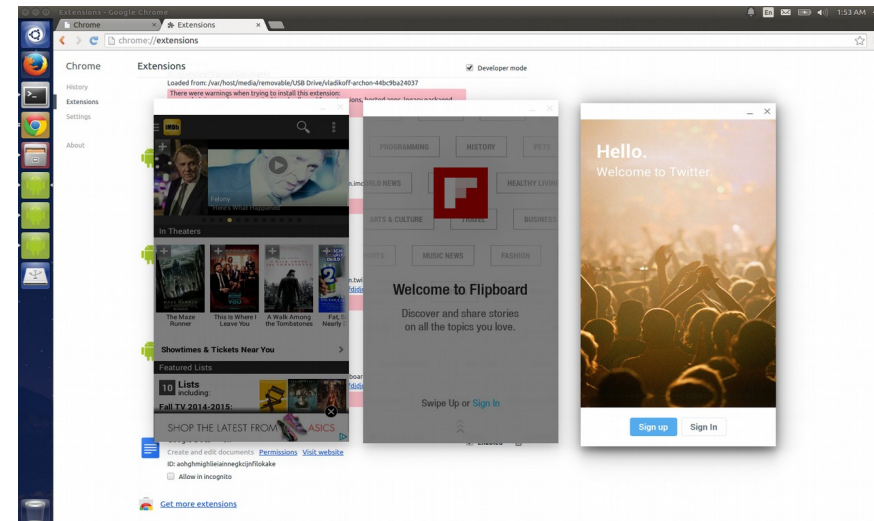
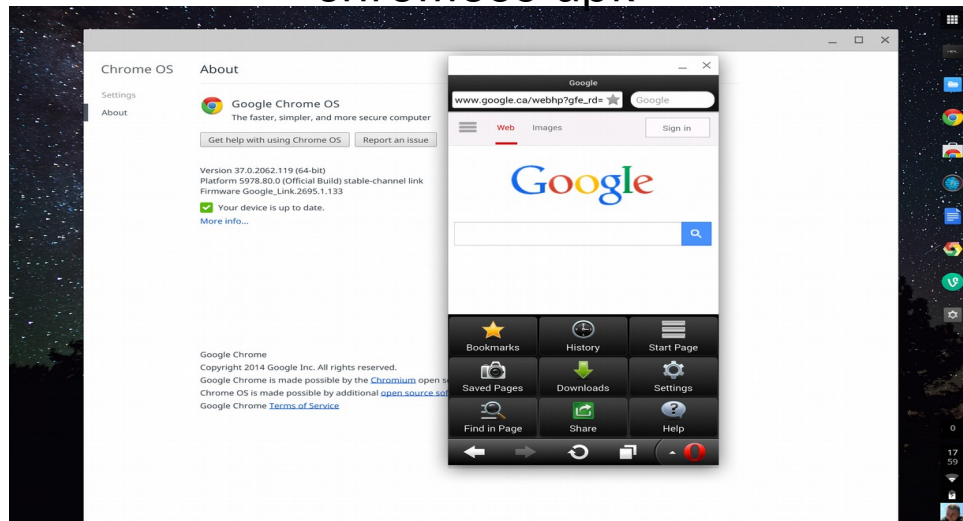
◎Crouton  
Ubuntu用LibreOffice  
Croutonでインストー  
ルするUbuntu、Debian  
版LibreOffice。  
フォントの追加修正が  
簡単なので調整が可能  
お勧め

# chromeos-apkとARChon

- Androidのapkを変換して Chromeで動作  
Chrome OS, OS X, Linux and Windowsなど対応
- サンプルAndroid appをChrome Storeからインストールしてruntimeを入手.
- 母艦がUbuntuの場合、`sudo apt-get install lib32stdc++6`
- インストール Node.js (via <http://nodejs.org/>),  
toolのインストール(might need a sudo prefix):  
`sudo npm install chromeos-apk -g`  
Run `chromeos-apk [path to apk file]`  
スマホアプリの場合:`chromeos-apk com.soundcloud.android.apk`  
タブレットアプリの場合:`chromeos-apk com.soundcloud.android.apk --tablet`
- Chromebookにファイルをコピーしてextensionを読み込む。  
ARChon Extensionをインストールする。

参考出展:  
[github chromeos-apk](#)  
[github ARChon Custom Runtime Guide](#)  
 ARChon

chromeos-apk





# ChromebookやLinux何に使うの？

- OSC浜名湖、名古屋、京都でご意見が沢山ありましたのでご質問

- 皆様、ノートパソコンを日常何に使っていますか？

- 1.動画配信購入やゲームとかしますか？
- 2.MS OfficeかLibreOffice使ってますか？
- 3.UbuntuやDebianとかLinux使った事がありますか？
- 4.RaspberryPiとか使った事がありますか？

- Linuxって何に使うのですか？メリットは？→何個該当しましたか？

- 1.WebやSMS、FlashゲームやOfficeなど
- 2.SkypeやLineなど(Android appやWineなどでも流用)
- 3.古いPCやスマホのリサイクル用OS、リモートデスクトップ端末
- 4.サーバとかクラウド環境。NASやルータにも(アクセス数制限がない)
- 5.3D CAD、3D CGソフトやCAE、Mathematicaなど無料で使いたい。
- 6.エミュレーションで他OS環境(ゲーム機、Android、FirefoxOSなど含む)
- 7.開発言語を無料で使いたい。ソフトを作りたい。
- 8.RaspberryPiで電子工作をDTYしたい
- 9.自宅クラスタ作りたい。超省エネマシンを使って節電したい。
- 10.Aircrack-ngで無線LANをモバイルクラッキングしたい
- 11.BOINCでがんの研究や宇宙人の探索に貢献したいw
- 12.北朝鮮OSやMIPS Linuxなどマニアックに使いたい(おい

~2個 一般人  
 ~6個 ヘビーユーザー  
 ~8個 マニア、開発者  
 ~10個 ギーク  
 ~12個 廃人

# ChromebookにLinuxディストリを入れる

- インストール方法は主に下記3つ。
  - 1、**Crouton**を使ってchroot環境にインストールする
  - 2、**ChrUbuntu**を使ってデュアルブートする
  - 3、ファームウェアを書き換えてブートローダごと各種OSをインストールする
  - 4、Dev\_Channel(開発版)OSを利用して**USBブート**(15年1月新機能追加)

ここからマニアック路線に進みます

# ARM端末のディストリ対応状況

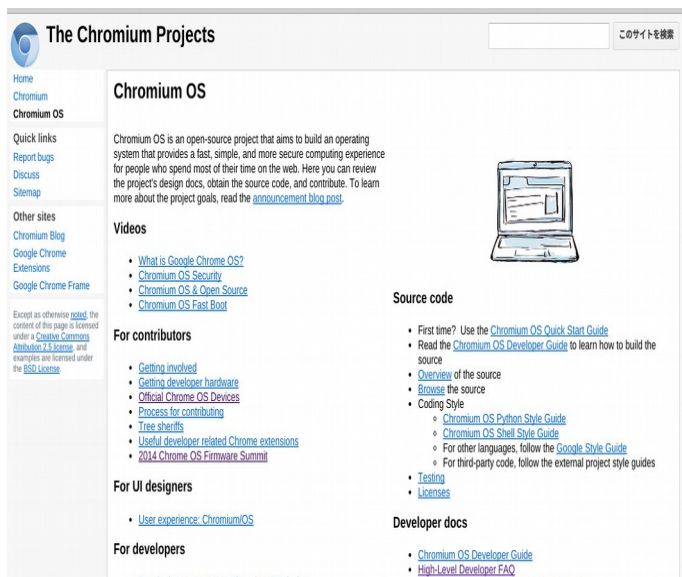
Psionから始まりiPaQ、玄箱～と組み込み機器から普及してきたARMのディストリ近況

ディストリ	Ubuntu	Debian	Fedora	Open Suse	Gentoo	Arch	Open Cocon	Android	Free BSD	Net BSD
バージョン	9.04～	2.2～	20～	12.2～			v8～	1.5～		1.6～
LinuxZaurus	△	◎	?		△			△		◎
Nokia N900	◎	◎	○	?	○	△		○	?	◎
Netwalker	◎	○	△		△			△		◎
DynabookAZ	◎	○	○	◎	◎	◎		◎		
Nexus7 (chroot除く)	◎	?				○		◎		
RaspberryPi		◎	◎	◎	◎	◎	◎	◎	◎	◎
MK802 USBAndroid	◎	○	○	○	△	○		◎	△	?
Samsung Chromebook	◎	◎	◎	◎	◎	◎		△	◎	?
コメント	リファレンス モデル 限定	動作 報告 多数	RasPi レポート	RasPi レポート	動作報告 多数	RasPi レポート	シングル ア ント	スマホ タブレット	ボード PC 中心	40機種 以上サ ポート

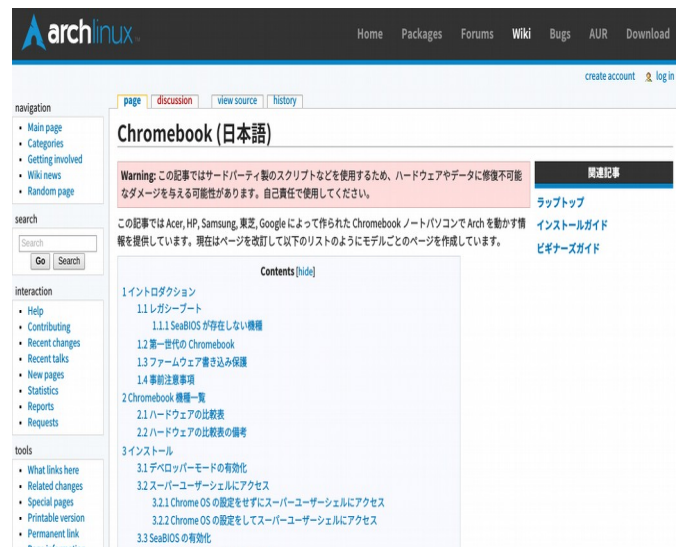
Kapperのネット独断調査（間違っていたらごめんなさい。修正します）

◎：公式サポートあり、○：公式HPに導入報告あり、△：ユーザー導入報告あり


# Chromebookハッキングに重要サイト



The Chromium OS Projects website. The main heading is "Chromium OS". Below it, there is a description: "Chromium OS is an open-source project that aims to build an operating system that provides a fast, simple, and more secure computing experience for people who spend most of their time on the web. Here you can review the project's design docs, obtain the source code, and contribute. To learn more about the project goals, read the [announcement blog post](#)." There is a small image of a laptop. The page is organized into sections: "Videos" with links like "What is Google Chrome OS?", "Chromium OS Security", "Chromium OS & Open Source", and "Chromium OS Fast Boot"; "Source code" with links like "First time? Use the Chromium OS Quick Start Guide"; "For contributors" with links like "Getting involved", "Getting developer hardware", "Official Chrome OS Devices", "Process for contributing", "Tree shenanigans", "Useful developer related Chrome extensions", and "2014 Chrome OS Firmware Summit"; "For UI designers" with a link "User experience: ChromiumOS"; and "For developers" with links "Chromium OS Developer Guide" and "High-Level Developer FAQ".



Arch Linux wiki page for "Chromebook (日本語)". The page title is "Chromebook (日本語)". There is a warning box: "Warning: この記事ではサードパーティ製のスクリプトなどを使用するため、ハードウェアやデータに修復不可能なダメージを与える可能性があります。自己責任で使用してください。" Below the warning, there is a text block: "この記事では Acer, HP, Samsung, 東芝, Google によって作られた Chromebook ノートパソコンで Arch を動かす情報を提供しています。現在はページを改訂して以下のリストのようにモデルごとのページを作成しています。" There is a "Contents" table of contents with sections: "1 インストール", "1.1 レガシーブート", "1.1.1 SeaBIOS が存在しない機種", "1.2 第一世代の Chromebook", "1.3 ファームウェア書き込み保護", "1.4 事前注意事項", "2 Chromebook 機種一覧", "2.1 ハードウェアの比較表", "2.2 ハードウェアの比較表の備考", "3 インストール", "3.1 デベロッパーモードの有効化", "3.2 スーパーユーザーシェルにアクセス", "3.2.1 Chrome OS の設定をせずにスーパーユーザーシェルにアクセス", "3.2.2 Chrome OS の設定をしてスーパーユーザーシェルにアクセス", "3.3 SeaBIOS の有効化".



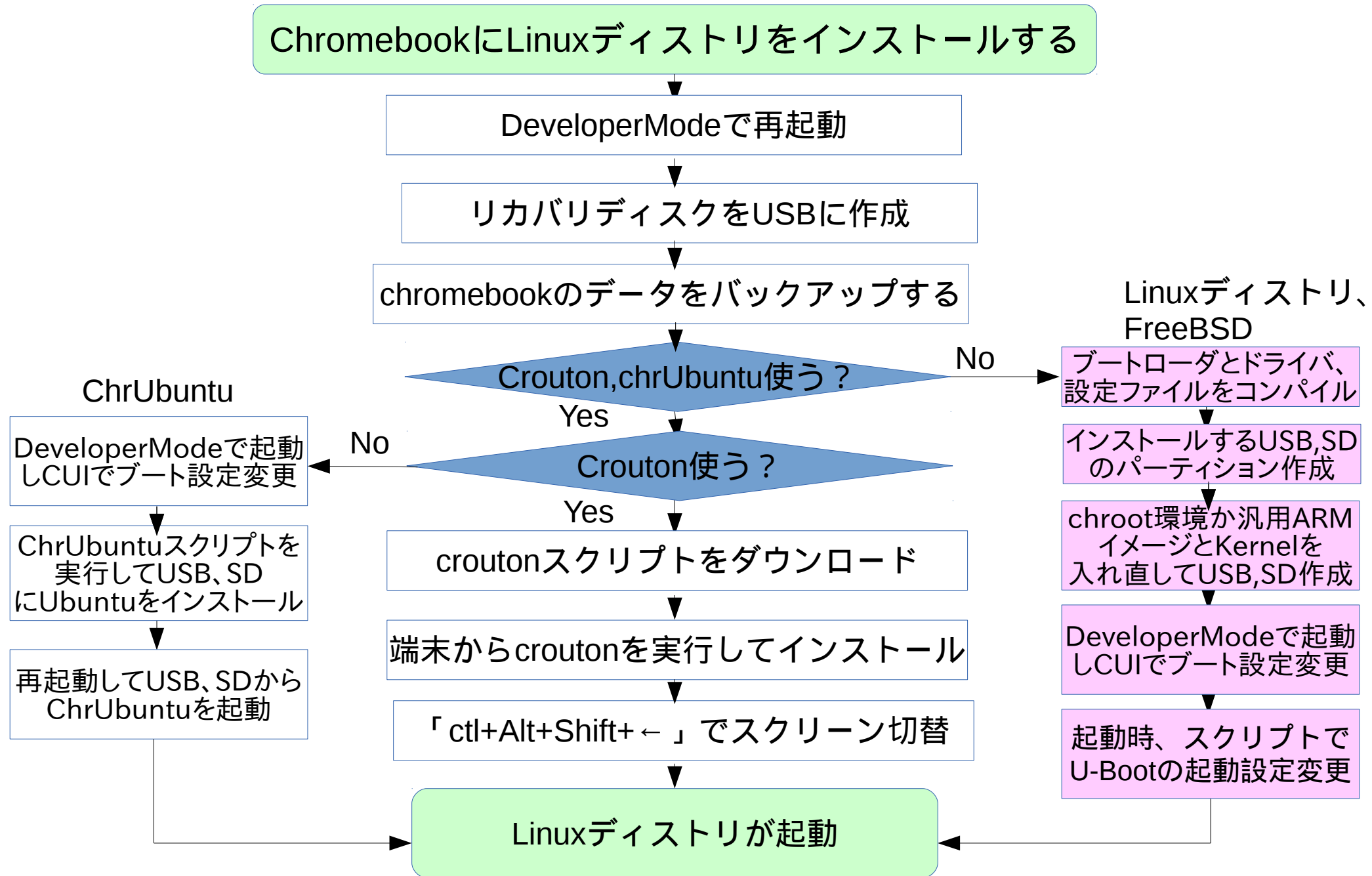
Coreboot on Chromebooks community page. The page features a large Chrome logo with the text "coreboot on Chromebooks" and "一般公開 731人のメンバー". Below the logo, there is a search bar and a list of posts. One post by John Lewis (owner) is highlighted: "I could give you a ROM, +yehanny olivares, but it would almost certainly brick your device, and we wouldn't be able to get any useful feedback". To the right, there is a section titled "このコミュニティについて" (About this community) with the text: "This is a community for people interested in running coreboot + SeaBIOS/Grub2 payloads on Chromebooks, and some form of Linux." Below this, there are "COMMUNITY RULES" listed: "1. It is expected that you have made an effort to read through the information available before posting. You are therefore also expected to demonstrate that you couldn't find the information you need elsewhere when asking your question. Failure to do so may result in your questions being ignored, or worse.", "2. Comments which aren't relevant to the OP will be ignored, or worse.", "3. Comments which are tldr will be ignored, or worse. Please make an effort to be succinct."

• THE Chromium OS Projects ChromeOS開発のオープンソースプロジェクト。ファームウェアからドライバ、コンパイル方法まで。大抵の技術情報がここに集約されているので必見。

• Arch Linux wiki Chromebook ArchLinuxのWikiページでLinux関連全般がこと細かく書いてある必須サイト。Chromebook関連、Linux移植関連も細かくWikiにまとめられており、困ったときには必ず役に立つ。日本語あり。

• Coreboot on Chromebook ChromebookのブートローダーのCorebootを開発している John Lewis氏のプロジェクト。Coreboot+SeaBIOS/Grub 2の関係ならこちら。

# 主な流れのイメージ



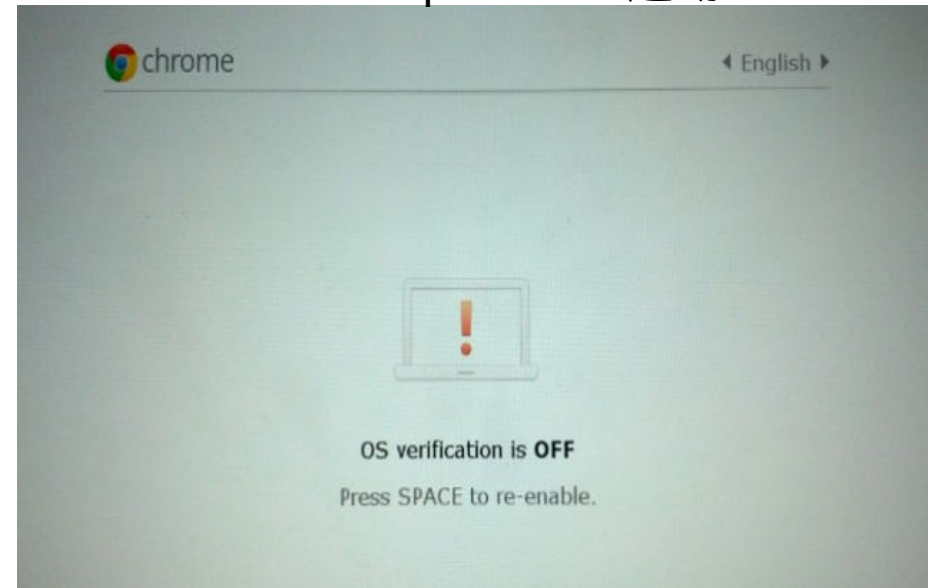
# Developer Modeで起動

- ChromeOSのDeveloperModeはセキュリティ機能をアンロックし内部アクセスする機能。ChromeOSを色々遊ぶ為にはDeveloperModeに切替が必須。
- DeveloperModeは電源を切って「Esc」+「リフレッシュ(F4)」+「電源ボタン」を押します。これはChromebookで言ういわゆる強制リセットというそうです。リカバリモードに入りますので、「Ctrl」+「D」→「Enter」を押して10秒くらいするとビープ音がしてデベロッパーモードに以降します。
- DeveloperModeで起動すると、内部のデータはリセットされるのでリカバリディスク作成とデータのバックアップが必要。
- リカバリディスクの作成はChromebrowserのアドレスバーから  
<chrome://imageburner>  
と入力してUSBかSDにインストール

## Chrome上リカバリディスク作成



## 参考出展: The Chromium Projects Developer Mode DeveloperMode起動

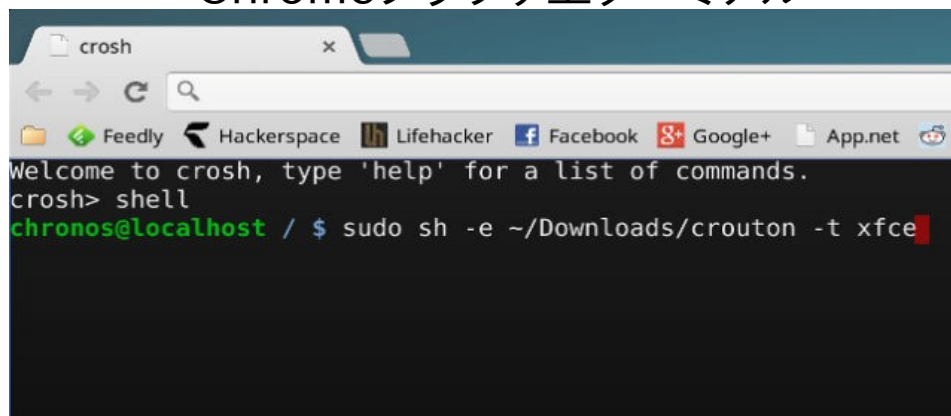




# Crouton

- ChRromium Os Universal chroot envirONmentの略。  
作者はGoogleの開発者、David Schneider氏。
- UbuntuとDebianに対応したchroot自動化ツール <https://goo.gl/fd3zc>
- 「ctl+alt+t」でターミナルを表示させ、「shell」を起動させ端末から使用  
`sudo sh ~/Downloads/crouton -t xfce`
- githubからメインスクリプトを落としてchroot環境をダウンロードする仕組み  
/mnt/stateful\_partition/crouton/(ディストリバージョン)にchroot作成

Chromeブラウザ上ターミナル



```

crash
Welcome to crash, type 'help' for a list of commands.
crash> shell
chronos@localhost / $ sudo sh -e ~/Downloads/crouton -t xfce
  
```

対応Ubuntu、Debianバージョン

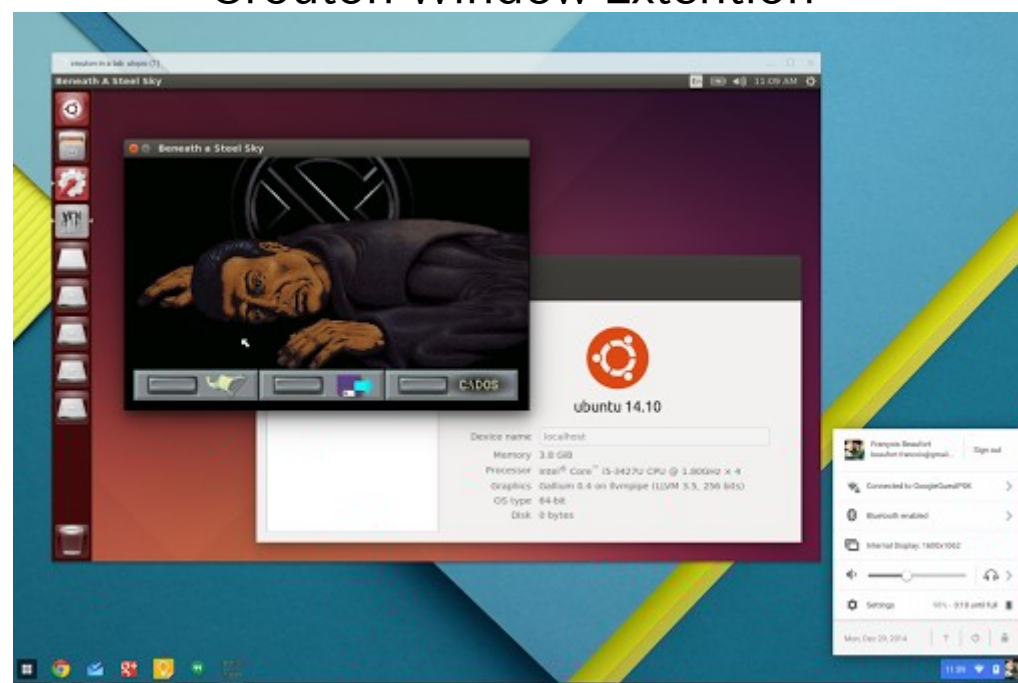
Recognized debian releases:  
potato\* woody\* sarge\* etch\* lenny\* squeeze\* **wheezy jessie sid**

Recognized kali releases:  
**kali**

Recognized ubuntu releases:  
warty\* hoary\* breezy\* dapper\* edgy\* feisty\* gutsy\* hardy\* intrepid\* jaunty\*  
karmic\* lucid\* maverick\* natty\* oneiric\* **precise** quantal\* raring\* saucy\*  
**trusty** utopic\*

Releases marked with \* are unsupported, but may work with some effort.

Crouton Window Extention



# Croutonのソースコードを読んでみて

- githubにあるソースコードのinstall/ubuntu/bootstrap

```
# Grab the latest release of debootstrap
echo 'Downloading latest debootstrap...' 1>&2
d=
'http://anonscm.debian.org/gitweb/p=di/debootstrap.git;a=snapshot;h=HEAD;s=f=tgz
```

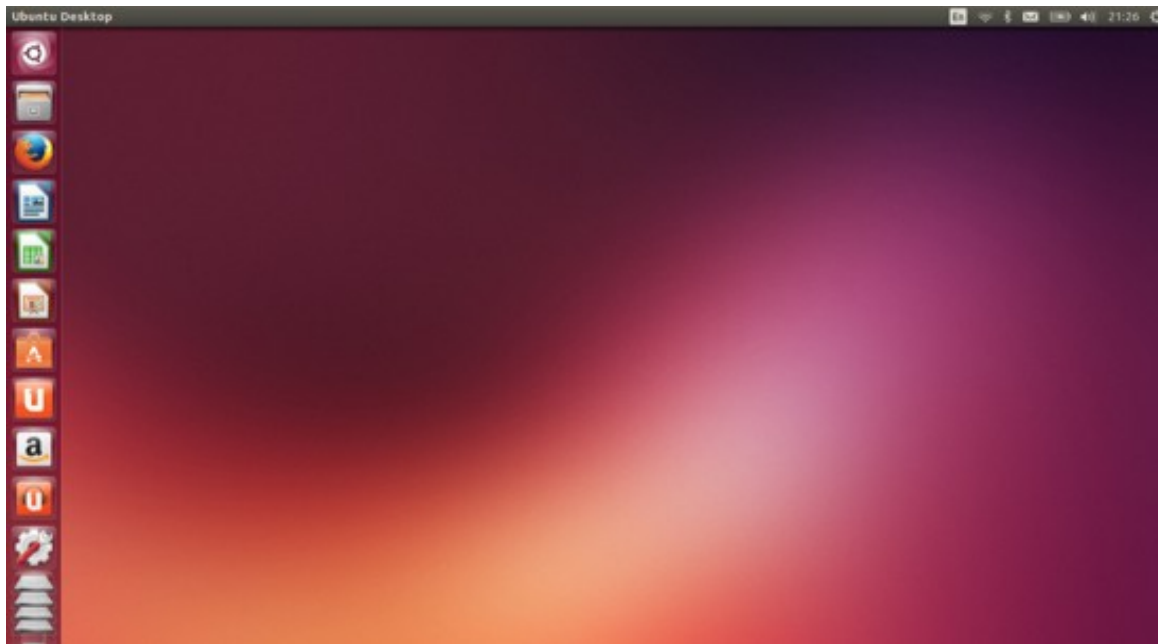
```
if ! wget -O- --no-verbose --timeout=60 -t2 "$d" ¥| tar -C "$tmp" --strip-components=1 -zx 2>/dev/null; then
echo 'Download from Debian gitweb failed. Trying latest release...' 1>&2d='
http://ftp.debian.org/debian/pool/main/d/debootstrap/
'f="` wget -O- --no-verbose --timeout=60 -t2 "$d" ¥| sed -ne 's ^.*¥
(debootstrap_[0-9.]*.tar.xz¥).*$ ¥1 p' ¥| tail -n 1 `"
```

- おやー？  
やっぱりdebootstrapとapt-getのスキプトの様ですねー  
ここを書き換えればchroot環境で色々と遊べそうです。



# ChrUbuntu

- Ubuntuに対応したデュアルブート&SD USB自動化インストールツール。  
作者はGoogleの開発者、Jay Lee氏。  
ChromeOSのLinux Kernelとドライバモジュールをそのまま使用する方式。  
x86対応でARMでは正式対応していない。動作しない機種もあるので注意。
- 「ctrl+alt+→キー」を押すと、Developer ConsoleというCUIの画面に切替え  
Ubuntuをインストールする仕組み。  
`curl -L -O http://goo.gl/9sgchs; sudo bash 9sgchs (34v87 /dev/mmcblk1)`  
パーティションを切り分けて  
`curl -L -O http://goo.gl/9sgchs; sudo bash 9sgchs`
- ブートする際にOSをChromeOSとUbuntuを「ctl+L」で切り替えて使用  
githubからメインスクリプトを落としてchroot環境をダウンロードする仕組み
- 通常にUbuntuが起動するが、起動が手動切替えで若干時間がかかる方式。



# emergeを使う dev\_installコマンド

- ChromeOSではshellでemergeが使えます
- DeveloperModeでdev\_installコマンドを使う事で関連アプリをダウンロードできます。
- emergeのレポジトリは各自設定必要。

```
chrome-extension://nkokcljplnhpfnfiajckommmmlphnl/html/crosh.html
Emerg installation complete. Installing additional optional packages.
/usr/bin/python: error while loading shared libraries: libpython2.7.so.1.0: cannot open shared object file
chronos@localhost / $ export LD_LIBRARY_PATH="/usr/local/lib"
chronos@localhost / $ sudo dev_install --reinstall
To clean up, we will run:
  rm -rf /usr/local/
Any content you have stored in there will be deleted.
Remove all installed packages now? (y/N) y
Removed all installed packages.
Reinstalling emerge.
Starting installation of developer packages.
First, we download the necessary files.
Downloading https://commondatastorage.googleapis.com/chromeos-dev-installer/board/daisy/canary-R39-6310.68
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload  Total   Spent    Left   Speed
100 251k  100 251k    0     0   104k    0  0:00:02  0:00:02  --:--:--  109k
Extracting /usr/local/portage/packages/sys-apps/sandbox-2.6-r1.tbz2
Downloading https://commondatastorage.googleapis.com/chromeos-dev-installer/board/daisy/canary-R39-6310.68
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload  Total   Spent    Left   Speed
100 720k  100 720k    0     0   140k    0  0:00:05  0:00:05  --:--:--  177k
Extracting /usr/local/portage/packages/net-misc/rsync-3.0.9-r3.tbz2
Downloading https://commondatastorage.googleapis.com/chromeos-dev-installer/board/daisy/canary-R39-6310.68
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload  Total   Spent    Left   Speed
100 7110  100 7110    0     0   8631    0  --:--:--  --:--:--  --:--:--  8723
Extracting /usr/local/portage/packages/virtual/libintl-0.tbz2
Downloading https://commondatastorage.googleapis.com/chromeos-dev-installer/board/daisy/canary-R39-6310.68
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload  Total   Spent    Left   Speed
100 23057  100 23057    0     0  23922    0  --:--:--  --:--:--  --:--:--  24118
Extracting /usr/local/portage/packages/app-misc/mime-types-9.tbz2
Downloading https://commondatastorage.googleapis.com/chromeos-dev-installer/board/daisy/canary-R39-6310.68
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload  Total   Spent    Left   Speed
0         0     0     0     0     0     0     0  --:--:--  --:--:--  --:--:--  0
```

# Windows、MacOSX (x86のみ)

- 試していませんが、C720でUSBインストールしてWindowsとMacOSXが動いたと動作報告あり。ドライバが動作しないそうなので実用性はないです。  
あくまで実験用とコメントされていました。  
WindowsやMacOSXのインストールはブートローダーでSeaBIOS対応機種のみです。
- Windows
  - 1、Windows8.1のブータブルUSBを作成
  - 2、C720のDeveloperModeにし、Legacybootで起動する
  - 3、「Ctl+L」キーでUSBから起動してWindows8.1をインストール
  - 4、インストール中は全てのドライバーをインストールしない。動作しなくなる。
  - 5、インストール終了後、Windows8.1を起動(勿論各種ドライバは動作しない)
- MacOSX  
OSx86の手法を用いてインストールするそうです。  
こちらもWindows8.1同様にドライバが動作しないそうです。

でもドライバ足りないから使い物にならね？

# Acerの掲示板にWindows8の ドライバよこせと苦情多数(ヒトイ)

ACER STORE

DISCUSSIONS ANSWERS IDEAS COMMUNITY NEWS

**\*\*GUIDE\*\* Install Windows 8 on the Acer C720 [ Edited ]**

The Acer C720 is a great notebook but the limited capabilities of Chrome OS can't way to install Windows 8 for those who do not have a spare machine:

- Make a Windows 8.1 bootable USB (Google this)
- Put the Acer C720 into Dev mode and enable legacy boot. (Google this)
- Install windows 8.1 from the Windows 8.1 USB you made by pressing Ctrl+L
- Do not let ANY drivers auto-install/update as they do not work properly

Problems:  
 Keyboard and Trackpad do not work.  
 Sound does not work.  
 Video Acceleration does not work.  
 Power Management does not work.

To make these work we need the drivers, Acer have officially said they will not release to somehow port over the Linux drivers -- and that seems impossible at this current

Why I did this? Because a user called: [Videomarquantoè](#) , kept writing articles on to prove how it is not but just needs full time developers working on it.

It was not intended for Windows 8 and is more capable of running it then other C

So it is far from hostile, yes it is difficult to upgrade the RAM as it is soldered but harder, what do you accept from a £200 laptop. Can you upgrade the RAM on a

Thanks to Joseph Halder for uploading video running Windows 8:  
<http://www.youtube.com/watch?v=m0IjnLyG9Y>

Follow me back after a few months of hiatus and I'm still posting about the Acer C

community.acer.com/t5/Acer-Ideas/Acer-C720-Driver-Release/id-p/222307

Anlegen vom Wiederherstellungsmedien für Windows 8

✓ Cannot Install windows 8 on Brand newAO 725

✓ Can't install Windows 8 on Aspire V5-431

GUIDE Install ExpressCache for SSD Caching

ASIO driver problem using Windows 8

Windows 8 Upgrade - Slow WiFi

Community Alert!

Please take a moment to review the latest [Community News](#) article to learn about new changes to your community membership.

**Did you know?**

Acer Community is also available in the following languages:

German

Spanish

Portuguese

動かんぞー

# Linuxディストリ、FreeBSD入替え

## ●FreeBSD

- Kernelを移植したソースコードをコンパイル
- インストールするUSB、SDのパーティション作成
- U-Bootのパーティション設定とKernel、Rootパーティションの設定
- ChromebookをDeveloperModeで起動。ログイン前にCUIに移動。  
USB、SDブート出来る様に設定変更。  
localhost ~ # `crossystem dev_boot_usb=1`
- cgptでUSB、SDのパーティションテーブル設定
- 再起動してU-Bootの設定。OSを起動させる

## ●OpenSuse

- もうひとつの方法ではChrUbuntuの仕組みで同様にインストールします。
- ChromebookをDeveloperModeで起動。ログイン前にCUIに移動。  
USB、SDブート出来る様に設定変更。  
localhost ~ # `crossystem dev_boot_usb=1`

⇒Crouton型とChrUbuntu型の両方インストール可能

# Linuxディストリ、FreeBSD入替え

## ●Fedora

ChromeOSの全データを消去して上書きしますので注意。

- ・スクリプトをダウンロード

<http://berrange.fedorapeople.org/install-f18-arm-chromebook-luks.sh>

- ・Optionally edit the script to change the FEDORA\_ROOT\_IMAGE\_URL and UBOOT\_URL env variables to point to a local mirror of the files.
- ・Optionally edit the script to set the ssid and psk parameters with the wifi connection details. If not set, the script will prompt for them
- ・Boot the ChromeBook in Developer Mode and login as a guest
- ・Use Ctrl+Alt+F2 to switch to the ChromeOS root shell (F2 is the key with the forward arrow on it, in the usual location you'd expect F2 to be)
- ・Copy the script downloaded earlier to /tmp in the ChromeOS root and give it executable permission
- ・Run `bash /tmp/install-f18-arm-chromebook-luks.sh`
- ・Watch as it reboots 3 times (keep an eye out for the LUKS key prompts on boots 3 and 4.
- ・Then either rejoice when firstboot appears and you subsequently get a graphical login prompt, or weep as you need to run the ChromeOS recovery procedure.



# Linuxディストリ、FreeBSD入替え

## ●Gentoo

ChromeOSの全データを消去して上書きしますので注意。

```
chronos@localhost / $ cd ~/gentoo
chronos@localhost ~/gentoo $ ls -la
total 32
drwxr-xr-x 3 root  root    4096 Jan 23 14:25 .
drwx--x--- 29 chronos chronos-access 12288 Jan 23 14:28 ..
drwx----- 2 root  root    16384 Jan 23 14:25 lost+found
```

- Download the latest stage 3 archive for armv7a\_hardfp:
 

```
chronos@localhost ~/gentoo $ sudo wget http://distfiles.gentoo.org/releases/arm/autobuilds/latest-stage3-armv7a_hardfp.txt
chronos@localhost ~/gentoo $ sudo wget http://distfiles.gentoo.org/releases/arm/autobuilds/`cat latest-stage3-armv7a_hardfp.txt | grep stage3-armv7a_hardfp`
```
- Extract the downloaded archive right onto the card e.g.:
 

```
chronos@localhost ~/gentoo $ sudo tar xjpf stage3-armv7a_hardfp-20141023.tar.bz2
```

 Clean up:
 

```
chronos@localhost ~/gentoo $ sudo rm latest-stage3-armv7a_hardfp.txt
chronos@localhost ~/gentoo $ sudo rm stage3-armv7a_hardfp-20141023.tar.bz2
```
- ```
chronos@localhost ~/gentoo $ sudo vim ~/gentoo/setup.sh
#!/bin/sh
GENTOO_DIR=/home/chronos/user/gentoo
mount -t proc /proc $GENTOO_DIR/proc
mount --rbind /sys $GENTOO_DIR/sys
mount --rbind /dev $GENTOO_DIR/dev
cp /etc/resolv.conf $GENTOO_DIR/etc
chronos@localhost ~/gentoo $ sudo vim ~/gentoo/enter.sh

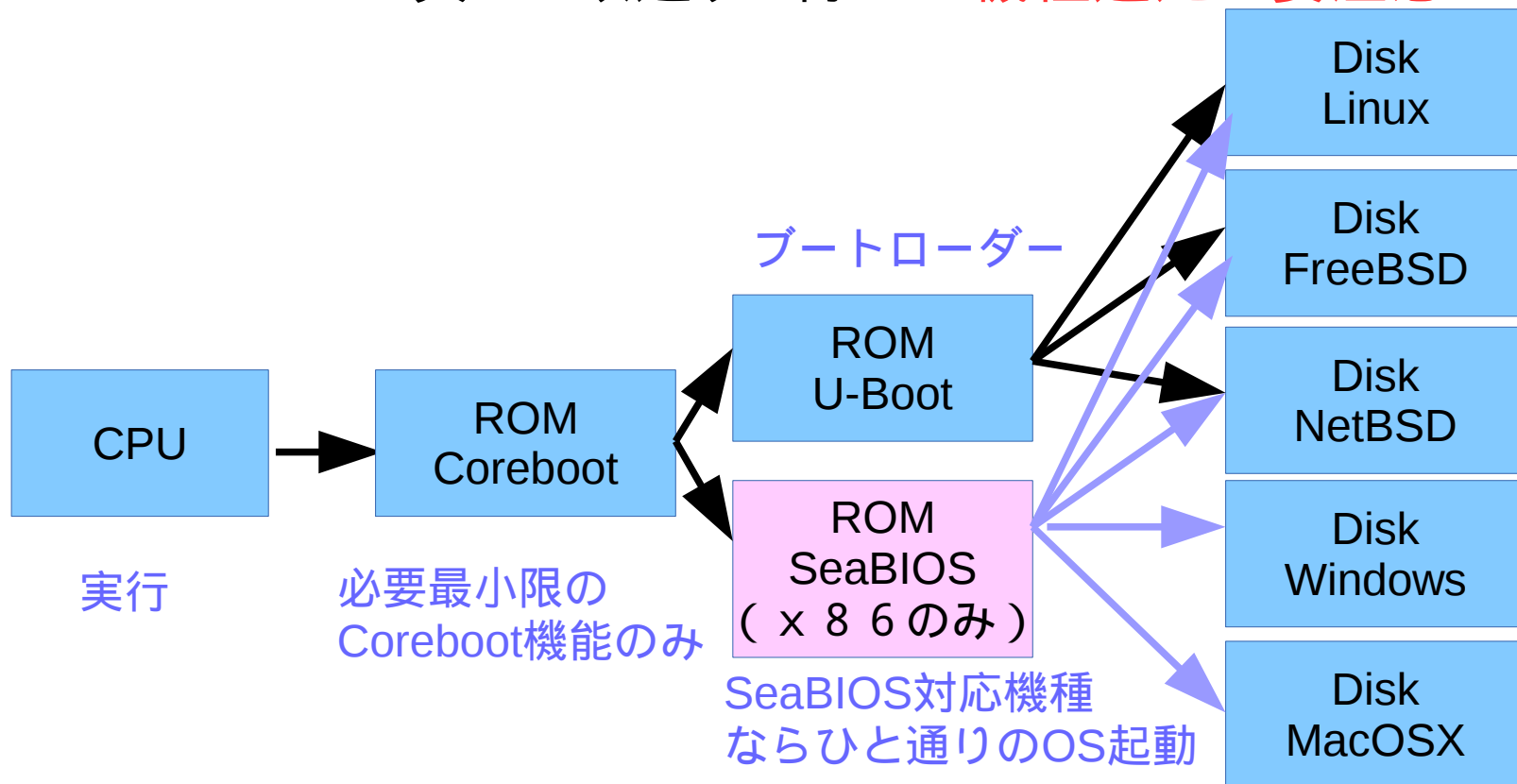
#!/bin/sh
GENTOO_DIR=/home/chronos/user/gentoo
LC_ALL=C chroot $GENTOO_DIR /bin/bash
chronos@localhost ~/gentoo $ sudo chmod u+x ~/gentoo/setup.sh ~/gentoo/enter.sh
```
- Execute the scripts:
 

```
chronos@localhost ~/gentoo $ sudo ~/gentoo/setup.sh
chronos@localhost ~/gentoo $ sudo ~/gentoo/enter.sh
```

# x86のブートローダーって？

## WindowsやMacOSX他を動かすには

- Corebootからペイロードして次のブートローダーから起動
- SeaBIOS対応機種なら様々なOSを起動出来る  
WindowsとMacOSXをChromebookで動かすには必要。  
課題はSeaBIOS対応ハードが少ない。
- SeaBIOS対応はHaswell 世代9機種。Bay-Trail世代は未対応。
- Chromebookを買って改造する際には機種選定に要注意





# Chromebook4つのブートローダー

- **Verified Mode**  
ChromeOSイメージのみ。起動しない場合、リカバリモードに移動
- **RecoveryMode**  
Googleから指定したUSB、SDリカバリディスクから起動
- **DeveloperMode**  
ジェイルブレイクモード。切替時にデータを消去する  
ChromeOSでrootシェルを実行出来る  
USBからOSを起動出来る。設定変更。  
`crossystem dev_boot_usb=1`  
`crossystem dev_boot_signed_only=0`
- **LegacyMode**  
サポートされていない他のOSを起動する方法  
Corebootから他のブートローダにペイロード出来る  
Intel Haswell世代ならSeaBIOSが使える  
DeveloperModeで `crossystem dev_boot_legacy=1` に設定

# ARMブートローダの仕組みとx86比較

ARMの場合、FlashROMからRAMにブートローダを転送し、Kernelと rootfsを読み込む構成です。ROMに書き込んでるブートローダは「ファームウェア」と呼ぶそうです。Androidの場合、Unlockが必要です。

| 種類                                  | ROM                              | RAM          | MBR<br>(ディスク)            | Kernel                     | init                                        | ログイン                              | 特徴                            |
|-------------------------------------|----------------------------------|--------------|--------------------------|----------------------------|---------------------------------------------|-----------------------------------|-------------------------------|
| x86~<br>Linux                       | BIOS<br>デバイス認識<br>ブートディスク選択      |              | GRUB<br>LILO他            | Kernel<br>ドライバ             | Daemon<br>他                                 | CUI<br>Xorg                       | BIOSと<br>DISKブー<br>トロー<br>ダ構成 |
| ARM<br>Android                      | 初期化<br>ドライバ<br>RAM転送<br>MLO      | Kernel<br>転送 | Kernel<br>rootfs<br>読み込み | Kernel<br>ドライバ<br>rootfs起動 | デバイス<br>Daemon<br>コンテキスト<br>マネージャ<br>Zygote | Dalvik<br>仮想マシン<br>Android<br>GUI | ブート<br>ローダー<br>のUnlock<br>が重要 |
| Android ブートローダー (ROM)               |                                  |              |                          |                            |                                             |                                   |                               |
| x86-<br>Chrome<br>book              | 初期化<br>ドライバ<br>RAM転送<br>CoreBoot | Kernel<br>転送 | Kernel<br>rootfs<br>読み込み | Kernel<br>ドライバ             | Daemon<br>他                                 | CUI<br>Xorg                       | ブート<br>ローダー<br>以降は共<br>通      |
| ブートローダー (U-boot、SeaBIOS)            |                                  |              |                          |                            |                                             |                                   |                               |
| ARMLinux<br>ディストリ<br>Chrome<br>book | 初期化<br>ドライバ<br>RAM転送<br>MLO      | Kernel<br>転送 | Kernel<br>rootfs<br>読み込み | Kernel<br>ドライバ             | Daemon<br>他                                 | CUI<br>Xorg                       | ブート<br>ローダー<br>以降はx8<br>6と同じ  |
| ブートローダー (U-boot他)                   |                                  |              |                          |                            |                                             |                                   |                               |

# ARM関連のブートローダ比較

明確な定義は決まってないですが、ROMブートローダとDiskブートローダの2つに分類される事が良くあります。

| 種類              | 対応MPU                                               | 対応OS                             | 対応ストレージ                                         | 対応フォーマット                                                                   | 起動モジュール             | ライセンス          |
|-----------------|-----------------------------------------------------|----------------------------------|-------------------------------------------------|----------------------------------------------------------------------------|---------------------|----------------|
| GNU GRUB        | x86                                                 | Linux<br>*BSD<br>MacOSX<br>MSDOS | HDD,floppy<br>USB,LAN<br>TFTP,Serial            | 全種類                                                                        | DISK他               | GPLv3          |
| LILO            | x86                                                 | Linux<br>*BSD<br>MSDOS           | HDD,floppy<br>USB,LAN<br>TFTP,Serial            | 全種類                                                                        | DISK他               | BSD<br>Licence |
| eCos<br>Redboot | ARM,x86,68k,<br>MIPS,Altera,P<br>owerPC,Super<br>H他 | Linux<br>*BSD<br>MacOSX<br>MSDOS | HDD,floppy<br>USB,LAN<br>TFTP,Serial            | JFFS2、EXT2、E<br>XT3、EXT4、FAT<br>他                                          | ROM、<br>RAM         | Mod<br>GPLv2+  |
| Das<br>U-boot   | ARM,x86,68k,<br>MIPS,Altera,P<br>owerPC,Super<br>H他 | Linux<br>*BSD<br>Android他        | HDD,floppy<br>USB,LAN,Zip<br>TFTP,Serial<br>NFS | Cramfs、EXT2、E<br>XT3、EXT4、FAT<br>、FDOS、JFFS2、<br>ReiserFS、UBIFS<br>、YAFFS2 | ROM、<br>RAM         | GPLv2          |
| SeaBIOS         | x86                                                 | Win,MAC<br>BSD他                  | SecondaryROM<br>Disk他                           | EXTx、FAT32、<br>NTFS他                                                       | ROM、RAM<br>Coreboot | LGPLv3         |
| Multirom        | ARM                                                 | Android<br>Linux                 | Android<br>SecondaryROM<br>USB                  | EXTx、FAT32、<br>NTFS                                                        | ROM、<br>RAM         | GPLv3          |

# ARM ChromebookのMainline Kernel

## 1、U-bootをコンパイルしてインストール

- パーティションを作る

  - 1st partition: kernel partition with nv u-boot.

  - 2nd partition: boot partition (must be formatted as FAT32 or Ext2).

  - 3rd partition: root partition.

- U-bootをコンパイル

  - `wget -O - http://commondatastorage.googleapis.com/chromeos-localmirror/distfiles/nv_uboot-snow.kpart.bz2 | bunzip2 >`

  - `nv_uboot.kpart`

  - `/Kernel`

  - `dd if=nv_uboot.kpart of=/dev/mmcblk1p1`

  - `cgpt add -i 1 -S 1 -T 5 -P 10 /dev/mmcblk1`

## 2、Mainline Kernelをコンパイルしてインストール

- `/git clone --depth 1 https://github.com/linux-exynos/linux.git -b dts-config linux`

- `cd linux`

  - `make exynos_defconfig`

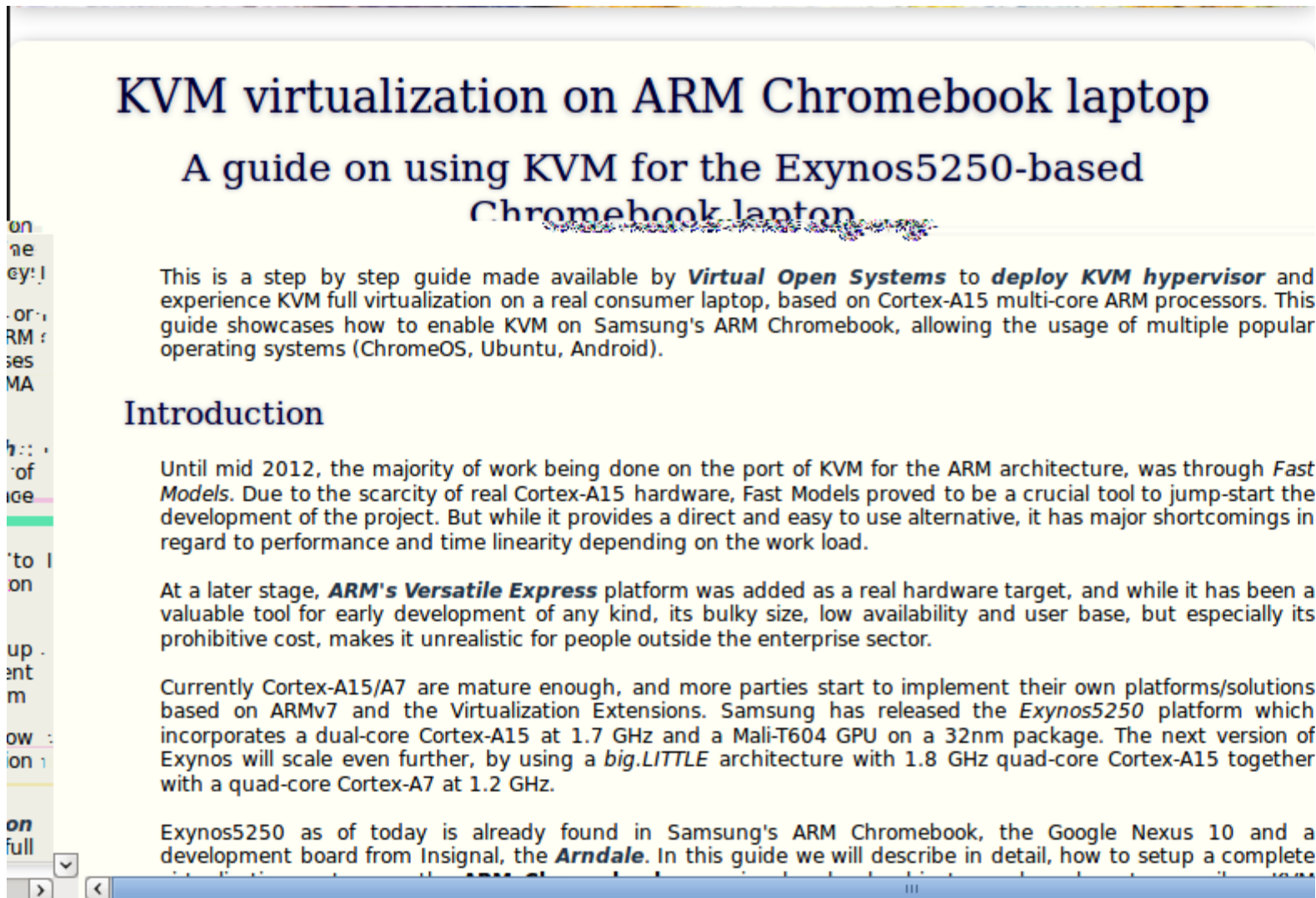
  - `make menuconfig`

- `mount /dev/mmcblk1p2 /mnt/externcp arch/arm/boot/{ulmage,dts/exynos5250-snow.dtb} /mnt/extern`  
`umount /mnt/extern`

- `mount /dev/mmcblk1p3 /mnt/extern`  
`INSTALL_MOD_PATH=/mnt/extern make modules_install`  
`umount /mnt/extern`

# KVMもChromebookで出来るそうなの

- まだ試していませんが、一応



The image shows a screenshot of a document titled "KVM virtualization on ARM Chromebook laptop". The document is displayed in a window with a sidebar on the left and a bottom navigation bar. The main content area has a light yellow background. The title is in a large, dark blue serif font. Below the title is a subtitle in a smaller, dark blue serif font. The main text is in a black serif font. The document is partially obscured by a sidebar on the left and a bottom navigation bar.

## KVM virtualization on ARM Chromebook laptop

### A guide on using KVM for the Exynos5250-based Chromebook laptop

This is a step by step guide made available by **Virtual Open Systems** to **deploy KVM hypervisor** and experience KVM full virtualization on a real consumer laptop, based on Cortex-A15 multi-core ARM processors. This guide showcases how to enable KVM on Samsung's ARM Chromebook, allowing the usage of multiple popular operating systems (ChromeOS, Ubuntu, Android).

### Introduction

Until mid 2012, the majority of work being done on the port of KVM for the ARM architecture, was through *Fast Models*. Due to the scarcity of real Cortex-A15 hardware, Fast Models proved to be a crucial tool to jump-start the development of the project. But while it provides a direct and easy to use alternative, it has major shortcomings in regard to performance and time linearity depending on the work load.

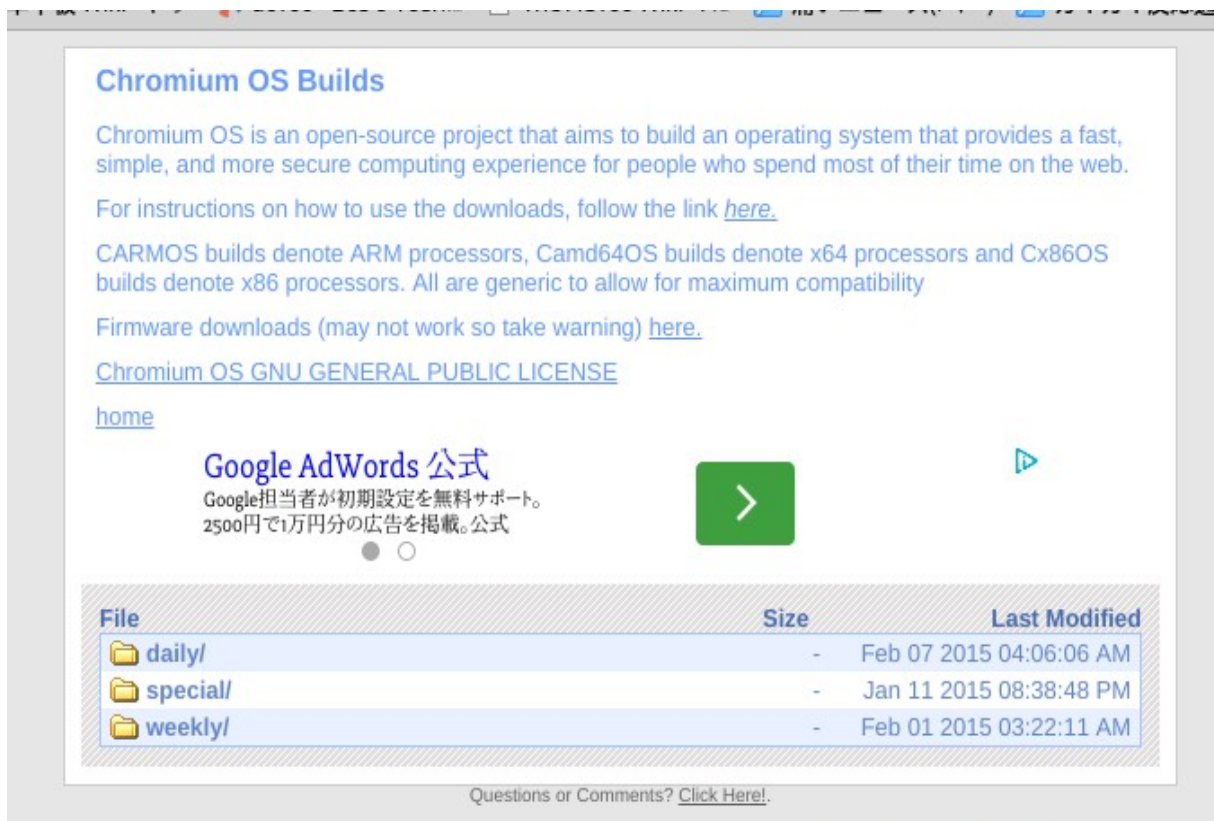
At a later stage, **ARM's Versatile Express** platform was added as a real hardware target, and while it has been a valuable tool for early development of any kind, its bulky size, low availability and user base, but especially its prohibitive cost, makes it unrealistic for people outside the enterprise sector.

Currently Cortex-A15/A7 are mature enough, and more parties start to implement their own platforms/solutions based on ARMv7 and the Virtualization Extensions. Samsung has released the *Exynos5250* platform which incorporates a dual-core Cortex-A15 at 1.7 GHz and a Mali-T604 GPU on a 32nm package. The next version of Exynos will scale even further, by using a *big.LITTLE* architecture with 1.8 GHz quad-core Cortex-A15 together with a quad-core Cortex-A7 at 1.2 GHz.

Exynos5250 as of today is already found in Samsung's ARM Chromebook, the Google Nexus 10 and a development board from Insignal, the **Arndale**. In this guide we will describe in detail, how to setup a complete

# ChromiumOS

- フリーで開発されているのがChromiumOS。  
各種仕様がThe Chromium Projectsで公開。
- USBかSDカードに書き込みして、普通のx86のPCから起動すれば、ChromiumOSが起動します。特別設定は不要ですがGoogleアカウント必要
- OSサポートやドライバ、Adobe Flash以外は殆ど同じです。  
Chromebookを購入する前に一度試してみてください。
- 推奨スペックは、CPU:1GHz メモリ:512MB～、HDD:4GB  
メモリは沢山あった方が良いでしょう。



**Chromium OS Builds**

Chromium OS is an open-source project that aims to build an operating system that provides a fast, simple, and more secure computing experience for people who spend most of their time on the web.

For instructions on how to use the downloads, follow the link [here](#).




CARMOS builds denote ARM processors, Camd64OS builds denote x64 processors and Cx86OS builds denote x86 processors. All are generic to allow for maximum compatibility

Firmware downloads (may not work so take warning) [here](#).

[Chromium OS GNU GENERAL PUBLIC LICENSE](#)

[home](#)

**Google AdWords 公式**  
Google担当者が初期設定を無料サポート。  
2500円で1万円分の広告を掲載。公式

| File                                                                                                         | Size | Last Modified           |
|--------------------------------------------------------------------------------------------------------------|------|-------------------------|
|  <a href="#">daily/</a>   | -    | Feb 07 2015 04:06:06 AM |
|  <a href="#">special/</a> | -    | Jan 11 2015 08:38:48 PM |
|  <a href="#">weekly/</a>  | -    | Feb 01 2015 03:22:11 AM |

Questions or Comments? [Click Here!](#)



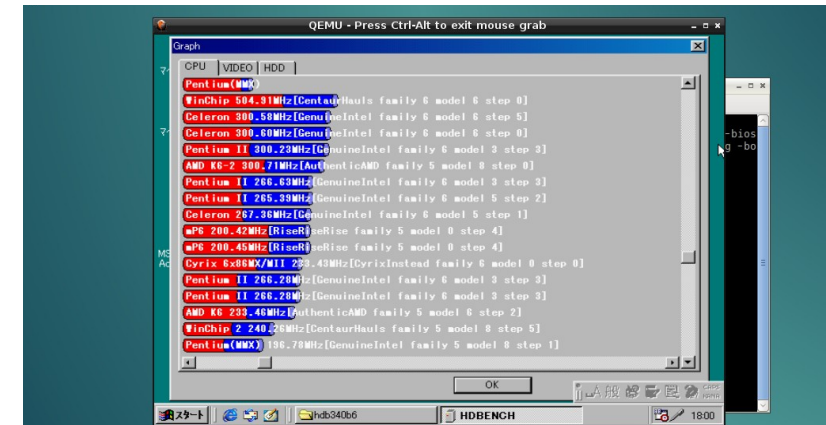
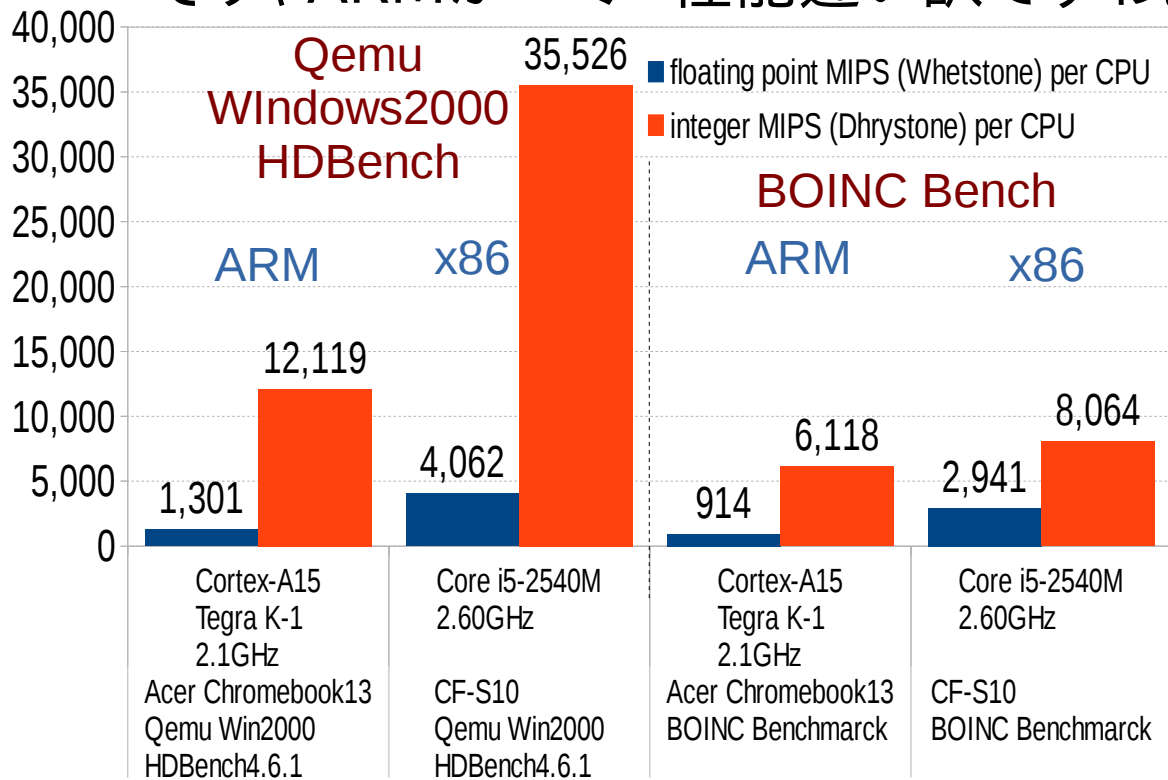
# PKGSRC on Chromebook

- NetBSDのパッケージャーpkgsrcをUbuntu chromebookでも使ってみる⇒何で？  
⇒1, 最新版のパッケージを使いたい。Mikutterとか随時最新版使いたい  
2, UbuntuのPPAでARMバイナリパッケージが殆ど無い。  
ソースパッケージから1個1個手動コンパイル必要。非常に面倒。  
3, OS、アーキテクチャからpkgsrcが独立しておりUnix、Linux、MacやWinでも使える  
ほどの高い移植性がARM環境では大変重宝する  
4, Debian SidとかDeb開発版をLTSとかと混載させたくない。独立させたい。  
5, 仮にUbuntu側とかでARM、Kernelサポートアウトされても継続して最新版使いたい。  
メーカー公式サポートが終わってからも使える最強パッケージャー。
- 最新版Mikutterが光の早さで移植されて使えます。  
最近ディストリ間でMikutterがいかに短い時間でリリースされるか競争してませんか？



# エミュレータで補完してみる

- Ubuntu、Debianを使えばWine、Qemu、Bochs,Steamなどである程度のアプリが動いたりします。
- x86エミュレーションを使えば、WindowsやNetBSDなんかも。TegraK-1でWin98がHDBenchでPentium2 266MHz相当でした。Nexus9でWindowを動かす際も似たようなスペックになるかと。
- ARMでのQemuエミュのベンチマークはx86:ARM=3:1で浮動小数演算能力にほぼ比例そりゃARMがエミュ性能遅い訳ですわ。





# OSC名古屋、京都、香港OSC、台湾OSDC

- OSC名古屋、京都、香港OSC、台湾OSDCに出展予定
- 内容:OSC名古屋、京都 未定  
海外 ARMモバイル機色々
- OSC名古屋 5月23日(土)  
OSC京都 8月?予定  
台湾OSDC ?月?日予定  
香港OSC 6月28日(土)  
台湾COSCUP 8月?予定
- 質問、ご要望はブースまで是非気軽にお聞き願います。  
まだ分からない事沢山あります、ゴメンナサイ。  
技術的な話題や議論、雑談を含め何でも。  
講演資料は会場でも配布しております。  
楽しみにお待ちしております。

# 過去の参考文献

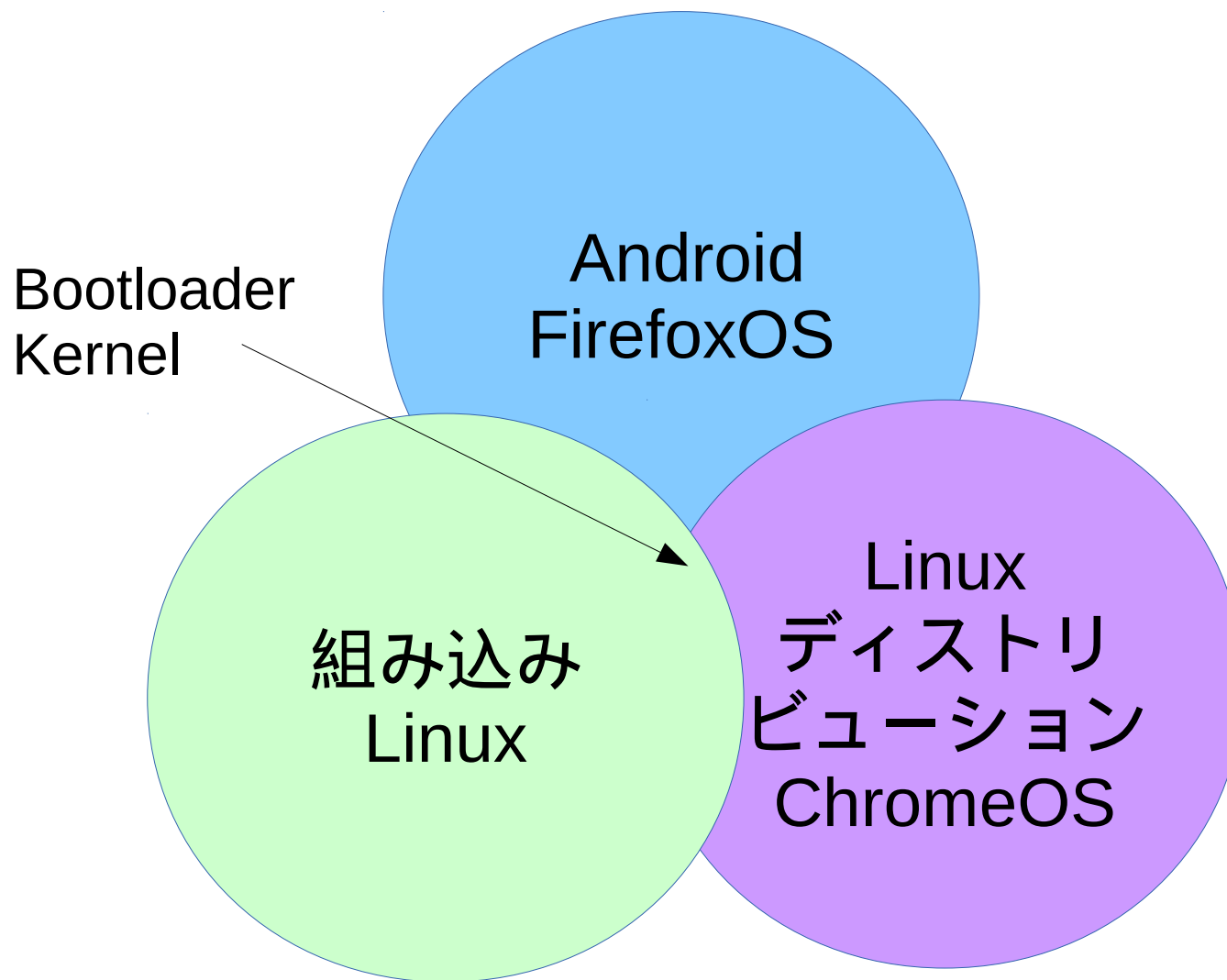
# Chromebookのブラウザ コマンドラインオプション

- `/opt/google/chrome/chrome --ppapi-flash-path=/opt/google/chrome/pepper/libpepflashplayer.so --ppapi-flash-version=16.0.0.305-r1 --ppapi-flash-args=enable_hw_video_decode=1 --enable-fixed-position-compositing --enable-impl-side-painting --max-tiles-for-interest-area=512 --ui-enable-per-tile-painting --ui-prioritize-in-gpu-process --use-gl=egl --enable-webkit-text-subpixel-positioning --enable-accelerated-overflow-scroll --default-tile-width=512 --default-tile-height=512 --gpu-sandbox-failures-fatal=yes --enable-webrtc-hw-vp8-encoding --ignore-resolution-limits-for-accelerated-video-decode --enable-logging --log-level=1 --use-cras --user-data-dir=/home/chronos --max-unused-resource-memory-usage-percentage=5 --system-developer-mode --login-profile=user --has-chromeos-keyboard --ash-default-wallpaper-large=/usr/share/chromeos-assets/wallpaper/oem_large.jpg --ash-default-wallpaper-small=/usr/share/chromeos-assets/wallpaper/oem_small.jpg --ash-default-wallpaper-is-oem --ash-guest-wallpaper-large=/usr/share/chromeos-assets/wallpaper/guest_large.jpg --ash-guest-wallpaper-small=/usr/share/chromeos-assets/wallpaper/guest_small.jpg --enterprise-enable-forced-re-enrollment --enterprise-enrollment-initial-modulus=13 --enterprise-enrollment-modulus-limit=17 --vmodule=screen_locker=1,webui_screen_locker=1,reset_screen_handler=1,update_engine_client=1,*ui/display/chromeos*=1,*ash/display*=1,*ui/ozone*=1,*zygote*=1,*plugin*=2 --login-manager --policy-switches-begin --num-raster-threads=4 --policy-switches-end --termination-message-file=/tmp/.org.chromium.Chromium.xy9vQL/.org.chromium.Chromium.sefxwn --first-exec-after-boot`

# 今回の重要な参考文献



# ARM Linuxの分類



# Das U-Boot解説

- ARM他で良く使われている高性能ROMブートローダ
- 2010年頃からRedbootの代替としてメジャーに
- 起動ディスクとして多数のフォーマット、ストレージに対応している強みあり。C言語で作成。CPUとメモリマップを直接指定、理解が必要。
- Linuxの場合は、mkimageコマンドでKernelをulmage等に変換。圧縮してフラッシュROM焼きしてROMから起動。メモリに展開。
- 第一パーティションにMLOと/boot/を置く
- rootfsをext3などに置く
- Kernelから起動

