OS交換

# メディア

## USBHDD

システムイメージ格納

Cドライブの容量以上

## 回復ドライブUSBメモリ

回復ドライブ

32GB

## Windows10インストール用USBメモリ

16GB

## Chrome OS (CloudReady)インストール用USBメモリ

16GB

# バックアップ

## OSだけでなくアプリケーションおよびデータも全てバックアップする。

## システムイメージのバックアップ

システムメニューから「ファイル名を指定して実行」で「control」を入力する。

コントロールパネルで「システムとセキュリティ」→「バックアップと復元(Windows 7)」を選択する。

「システムイメージの作成」を選択し、バックアップ先のUSB HDDドライブを指定する。

「システム修復ディスク」は作成しない。（これはCDまたはDVDドライブが必要になる）

## 回復ドライブの作成

システムメニューから「ファイル名を指定して実行」で「control」を入力する。

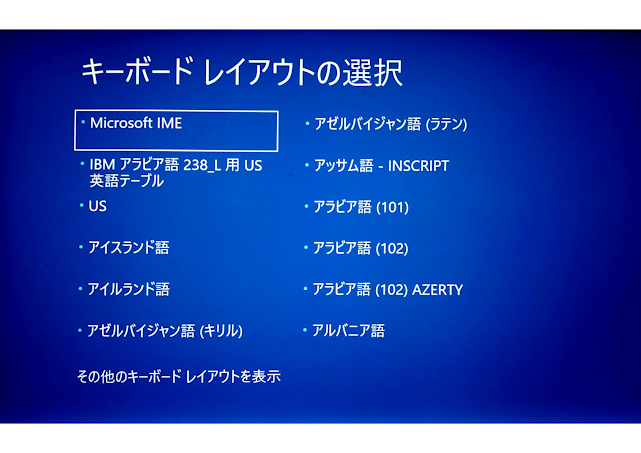
コントロールパネルで「システムとセキュリティ」→「セキュリティとメンテナンス」→「回復」→「回復ドライブの作成」を選択する。

## 復元

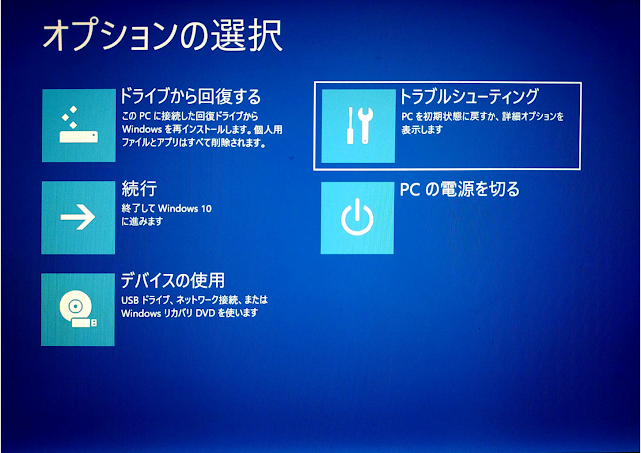
HZ750にSanDiskの64GB USBメモリおよびUSB HDD(HDPX-UTS 2TB)を接続した状態で電源を入れる。



「Microsoft IME」を選択してEnterを押す。



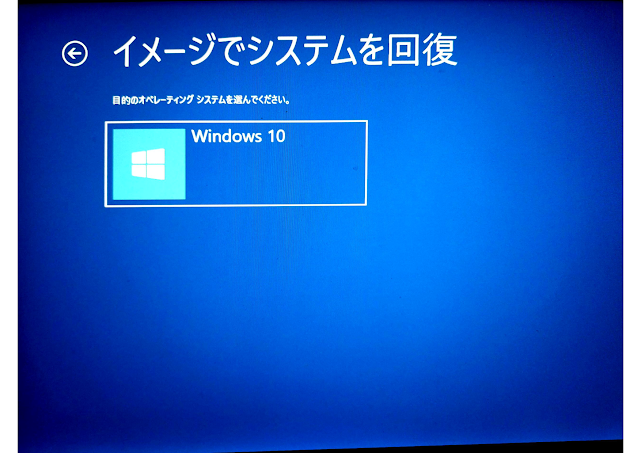
トラブルシューティングを選択してEnterを押す。



「イメージでシステムを回復」を選択してEnterを押す。



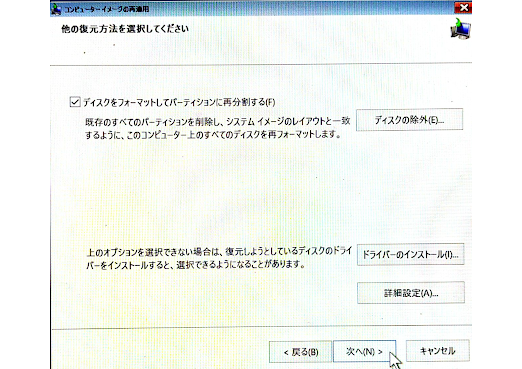
「Windows10」を選択してEnterを押す。



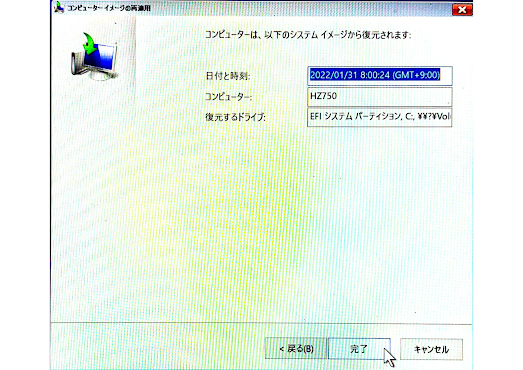
HDDにあるシステムイメージの内、最新のものが選択される。必要に応じて選択をして、「次へ」を押す。



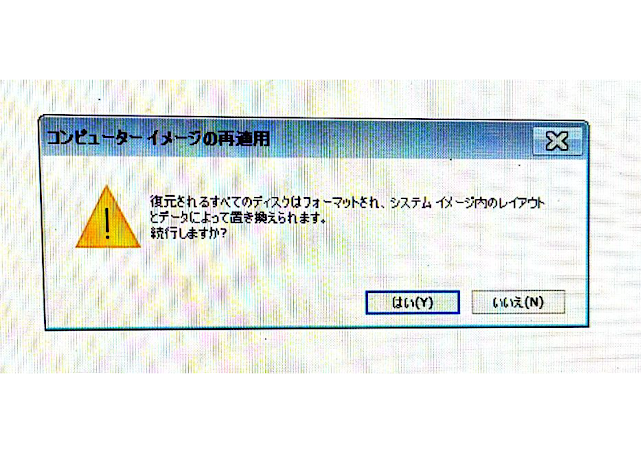
「ディスクをフォーマットしてパーティションに再分割する」をチェックして「次へ」を押す。



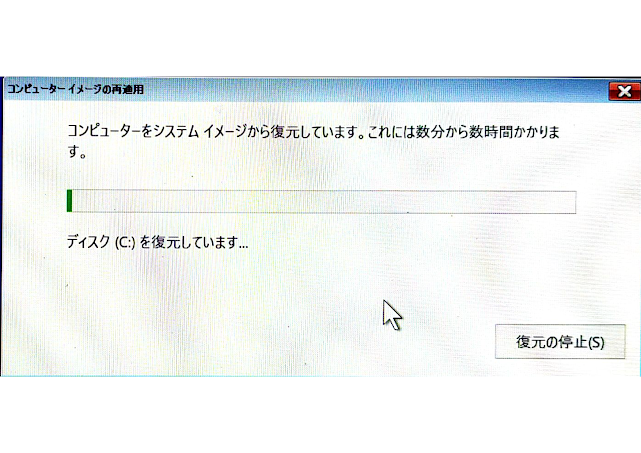
「完了」を押す。



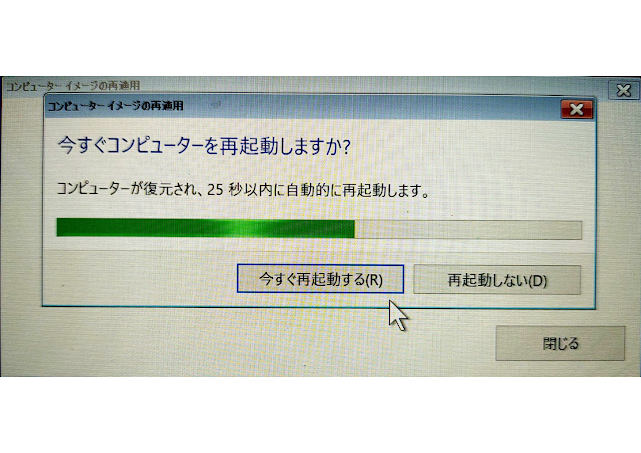
「はい」を押す。



復元が開始される。



復元が終わると「今すぐコンピューターを再起動しますか？」が表示される。USBメモリとHDDを外して「今すぐ再起動する」を押す。



# Linux Mint セットアップ

## タッチパッドの設定

「スクロール方向を反転する」をチェックする。

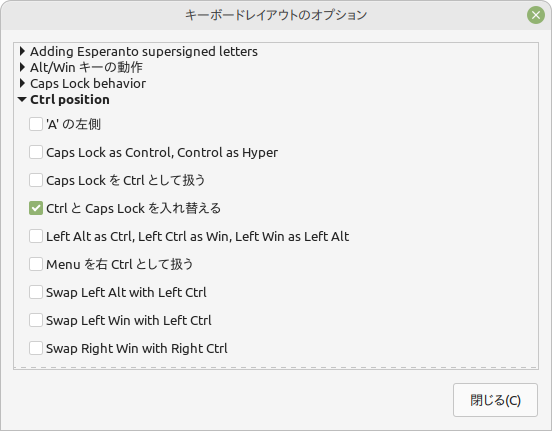
## 

## CapslockとCtrlの交換

/etc/default/keyboardを編集する。

XKBOPTIONS="ctrl:swapcaps"

あるいは「設定」→「キーボード」の「レイアウト」タブの右下にある「オプション(O)…」で



## Github Desktiopインストール

<https://github.com/shiftkey/desktop>

$ wget -qO - https://packagecloud.io/shiftkey/desktop/gpgkey | sudo tee /etc/apt/trusted.gpg.d/shiftkey-desktop.asc > /dev/null

$ sudo sh -c 'echo "deb [arch=amd64] https://packagecloud.io/shiftkey/desktop/any/ any main" > /etc/apt/sources.list.d/packagecloud-shiftkey-desktop.list'

$ sudo apt-get update

$ sudo apt install github-desktop

## USBドライブ上のリポジトリのクローンを作成

途中までうまく行くが、クローン作成に失敗する。

日本語のフォルダ名は問題ないようだが、

## JDK17インストール

sudo add-apt-repository ppa:linuxuprising/java

sudo apt update

sudo apt install oracle-java17-installer

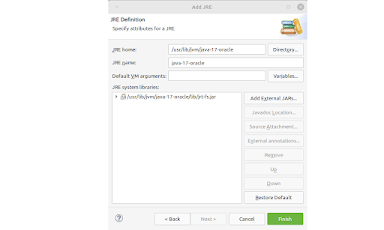
バージョンの切り替え

sudo update-alternatives --config java



「file://」はなくてもよい。

eclipse設定



## eclipseインストール

* Eclipseを下記URLからダウンロードします。

[www.eclipse.org 1 tweet8 users](https://www.eclipse.org/downloads/)

[](https://www.eclipse.org/downloads/)

**[Eclipse Downloads | The Eclipse Foundation](https://www.eclipse.org/downloads/)**

<https://www.eclipse.org/downloads/>

[The Eclipse Foundation - home to a global community, the Eclipse IDE, Jakarta EE and over 350 open source projects, including runtimes, tools and frameworks.](https://www.eclipse.org/downloads/)

* ダウンロードされたファイルを適当なフォルダに移動します。

mv eclipse-inst-linux64.tar.gz ../ドキュメント/java/.

* 圧縮ファイルを解凍します。

tar xvfz eclipse-inst-linux64.tar.gz

* 解凍されたフォルダ内に移動して、実行ファイルを実行します。

cd eclipse-inst

./eclipse-installer

* 「Eclipse IDE for java Developers」をクリックします。

## Windowsとファイル共有

* Windowsマシンのアドレスは192.168.100.159、共有フォルダ名は「d」として以下を実行する。
* sudo mount -t cifs -o user=saka,password=javar00t,uid=saka1029 192.168.100.159/d /mnt/windows/lggram13/d

## フォントの追加

* 「ソフトウェアの管理」で「fonts-migmix」を検索してインストール。

# Ubuntuセットアップ

## CapslockとCtrlの交換

/etc/default/keyboardを編集する。

XKBOPTIONS="ctrl:swapcaps"

## JDKインストール

sudo apt install openjdk-17-jdk-headless

saka1029@HZ750:~$ sudo update-alternatives --config java

[sudo] saka1029 のパスワード:

リンクグループ java に 1 つの alternative のみがあります (/usr/bin/java が提供): /usr/lib/jvm/java-17-openjdk-amd64/bin/java

設定は行いません。

## eclipseインストール

「ソフトウェアの管理」で「eclipse」を検索してインストール。

## 「変換」キーで入力切替

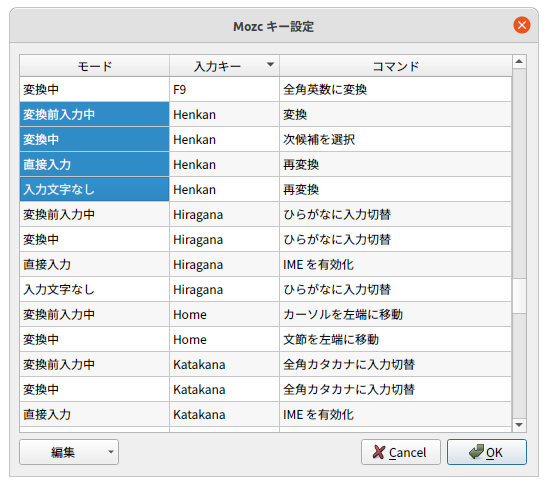
「Mozcの設定」から、かな切り替えをCtrl + Spaceに変更

ここから先はいつもと同じです。

キー設定の選択 > 編集



パッティングを避けるために、「入力キー」が「Henkan」になっているコマンドを削除する。



「入力キー」でソートして、「入力キー」が「Hankaku/Zenkaku」になっているものを「Henkan」に変更します。

変更が終わったら「OK」で設定を終了します。

## Github Desktiopインストール

<https://github.com/shiftkey/desktop>

$ wget -qO - https://packagecloud.io/shiftkey/desktop/gpgkey | sudo tee /etc/apt/trusted.gpg.d/shiftkey-desktop.asc > /dev/null

$ sudo sh -c 'echo "deb [arch=amd64] https://packagecloud.io/shiftkey/desktop/any/ any main" > /etc/apt/sources.list.d/packagecloud-shiftkey-desktop.list'

$ sudo apt-get update

$ sudo apt install github-desktop

| Cloning into '/home/saka1029/git/history'...  remote: Enumerating objects: 1427, done.  remote: Counting objects: 100% (1427/1427), done.  remote: Compressing objects: 100% (1343/1343), done.  remote: Total 1427 (delta 178), reused 1177 (delta 58), pack-reused 0  Receiving objects: 100% (1427/1427), 463.93 MiB | 25.83 MiB/s, done.  Resolving deltas: 100% (178/178), done.  error: unable to create file backup/internet/GyaOの保存　ダウンロード方法 １　GetASFStreamの使い方（フリーソフト）　GASで最新のギャオを録画　WMVを変換して携帯,PSP,IPODなどで再生して見る（高画質） Vista,Windows7対応の無料ツール紹介.htm: File name too long  Updating files: 100% (1155/1155), done.  fatal: unable to checkout working tree  warning: Clone succeeded, but checkout failed.  You can inspect what was checked out with 'git status'  and retry with 'git restore --source=HEAD :/'  Would you like to retry cloning ? |
| --- |

# 最近のVirtualBoxでAndroid-x86を使う方法

Android-x86は、スマートフォン向けOSであるAndroidを、PC向けに移植したものだ。

そして、それをVirtualBox上にインストールして使う記事がいくつか存在する。私もそれにしたがって、Android 9を仮想マシンに入れることにした。

## しかしながら...

私の環境は、Ubuntu 18.04、VirtualBox 6.1.10

ほかサイトの記事には、VirtualBoxの設定画面で、「3Dアクセラレーションを有効化」にチェックを入れなければならないという記述のある記事があった。

もちろん私も、チェックを入れて起動してみるが...

起動しない。

いや、起動はするのだが、GUIが出ずに、CUIのコンソール画面で止まってしまう。

「Android-x86 VirtualBox 起動しない」で検索してみるも、3Dアクセラレーションを有効にせよの一点張りである。

## 対処法

この場合の対処法である。

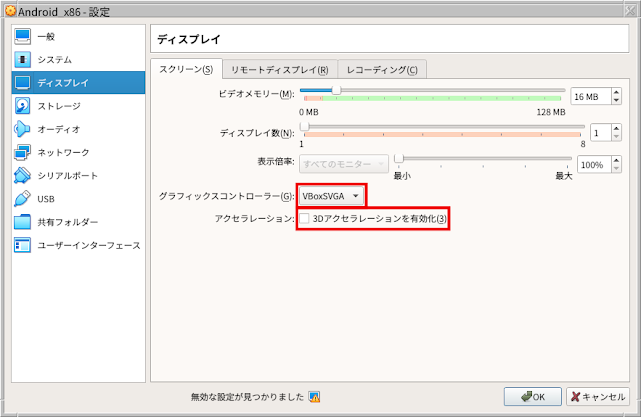
原因は、グラフィックスコントローラが「VMSVGA」に設定されていることが原因である。

まず、VirtualBoxの、Android-x86の仮想マシンの設定を開き、「ディスプレイ」に移動する。そうしたら、あの、「3Dアクセラレーションを有効化」のチェックを外す。

それでもって、グラフィックスコントローラーに、「VBoxSVGA」もしくは、「VBoxVGA」を指定する。

このとき、「無効な設定が見つかりました」と怒られるが、これで普通に動くので無視。

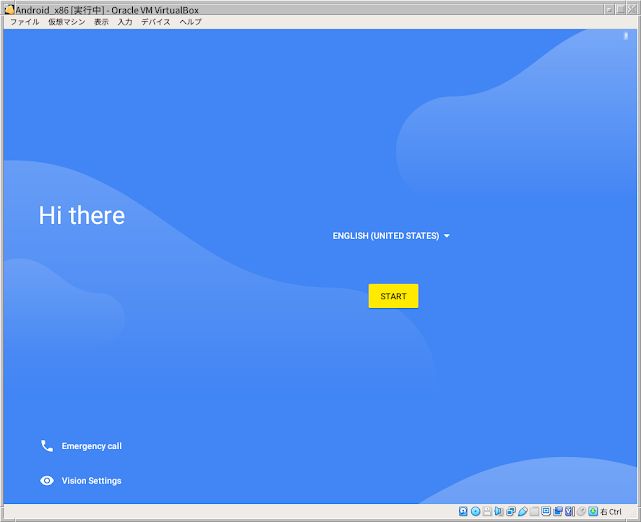
ちなみに、「3Dアクセラレーションを有効化」のチェックを入れたまま、「VBoxSVGA」、「VBoxVGA」を指定してしまうと、強制的にグラフィックスコントローラーが「VMSVGA」に変更されてしまい、動作しなくなるので注意。



## 起動してみよう

設定が完了したので、インストールなしモード(Run Android-x86 without installation)で起動してみると...

「android」のロゴが点滅して...



起動した!

あとは、正常に動くことを確認できたので、一旦仮想マシンを再起動して、Android-x86をインストールしていけばいい。Android-x86のインストール方法は、あまり昔と変わりはないようなので、ほかサイトの記事でも参考にすればよいかと。

# Windows以外のバックアップ・リストア

## partimage

Partimageはオープンソースのディスクバックアップソフトウェアです。セクターベースでサポートされているファイルシステムを持つパーティションをイメージファイルに保存します。Linuxで動作しますが、WindowsおよびほとんどのLinuxファイルシステムがサポートされています。イメージファイルは、ディスク容量と転送時間を節約するために圧縮でき、CDまたはDVDにコピーするために複数のファイルに分割できます。

**制限事項**

**Partimageはext4またはbtrfsファイルシステムをサポートしていません。**

## fsarchiver

FSArchiverは、ファイルシステムの内容を圧縮されたアーカイブファイルに保存できるシステムツールです。ファイルシステムは、異なるサイズのパーティションに復元でき、別のファイルシステムに復元できます。tar / darとは異なり、FSArchiverは、データをパーティションに抽出するときにファイルシステムも作成します。データを保護するために、すべてがアーカイブにチェックサムされます。アーカイブが破損している場合は、アーカイブ全体ではなく、現在のファイルを失うだけです。FsarchiverはGPL-v2ライセンスの下でリリースされています。

### ファイルシステムをアーカイブに保存する方法

* FSArchiverを使用してディスクのパーティションをバックアップする方法は次のとおりです。Linuxオペレーティングシステムが/ dev / sda1にインストールされていて、それを/ mnt / backup上のファイルにバックアップしたいとします。このコマンドはlivecdから実行できます。
* fsarchiver savefs /mnt/backup/gentoo-rootfs.fsa /dev/sda1
* 複数のファイルシステムを1つのアーカイブファイルにアーカイブすることもできます。
* **fsarchiver savefs /mnt/backup/gentoo-rootfs.fsa /dev/sda1 /dev/sda2 /dev/volgroup/lv01**
* 2つのファイルシステムをアーカイブに保存した場合の出力例を次に示します。
* # fsarchiver savefs -o /backup/backup-fsa/backup-fsa025-gentoo-amd64-20090103-01.fsa /dev/sda1 /dev/sda2 -v -j4 -A
* filesystem features: [has\_journal,resize\_inode,dir\_index,filetype,sparse\_super,large\_file]
* ============= archiving filesystem /dev/sda1 =============
* -[00][REGFILE ] /vmlinuz-2.6.25.20-x64-fd13
* -[00][REGFILE ] /sysresccd/memdisk
* -[00][REGFILE ] /sysresccd/pxelinux.0
* -[00][REGFILE ] /sysresccd/initram.igz
* -[00][REGFILE ] /sysresccd/boot.cat
* .....
* -[00][DIR ] /mkbootcd-gentoo64
* -[00][REGFILE ] /System.map-2.6.25.20-x64-fd13
* -[00][REGFILE ] /config-2.6.25.20-x64-fd13
* -[00][REGFILE ] /config-2.6.27.09-x64-fd16
* -[00][DIR ] /
* ============= archiving filesystem /dev/sda2 =============
* -[01][SYMLINK ] /bin/bb
* -[01][REGFILE ] /bin/dd
* -[01][REGFILE ] /bin/cp
* -[01][REGFILE ] /bin/df
* .....
* -[01][REGFILE ] /fdoverlay/profiles/repo\_name
* -[01][DIR ] /fdoverlay/profiles
* -[01][DIR ] /fdoverlay
* -[01][DIR ] /

### アーカイブからファイルシステムを抽出する方法

* FSArchiverは、アーカイブごとに複数のファイルシステムをサポートします。そのため、復元するファイルシステムを指定する必要があります。各ファイルシステムには0から始まる番号があります。アーカイブ内の最初のファイルシステムはファイルシステム番号0、2番目はファイルシステム番号1になります…一度に1つのファイルシステム、または1つのコマンドで複数のファイルシステムを復元できます。
* アーカイブにファイルシステムが1つしかない場合に、アーカイブからファイルシステムを復元する方法は次のとおりです。
* fsarchiver restfs /mnt/backup/gentoo-rootfs.fsa id=0,dest=/dev/sda1
* アーカイブから2番目のファイルシステムを復元する方法があります（2番目=番号1）：
* fsarchiver restfs /mnt/backup/archive-multple-filesystems.fsa id=1,dest=/dev/sdb1
* 1番目と2番目のファイルシステムの両方を同時に復元することもできます:(番号0と1）
* fsarchiver restfs /mnt/backup/archive-multple-filesystems.fsa id=0,dest=/dev/sda1 id=1,dest=/dev/sdb1
* オプション-Fは、古いバージョンのファイルシステムを変換するために使用されました。たとえば、新しいパーティションにreiserfsとして保存されたときにext2であったファイルシステムを復元できます。ここで、宛先パーティションでオプションmkfs = xxxを指定する必要があります。最初のファイルシステムをアーカイブから/ dev / sda1に復元し、同時にreiserfsに変換する方法は次のとおりです。
* fsarchiver restfs /mnt/backup/gentoo-rootfs.fsa id=0,dest=/dev/sda1,mkfs=reiserfs
* アーカイブに関する情報を表示する
* アーカイブに何が保存されているかを知ることは有用かもしれません。これは、 archinfoを使用して行うことができます 。ファイルシステムの数、それらのプロパティ、ファイルシステムの元のサイズ、および使用されているスペースの量がわかります。
* fsarchiver archinfo /backup/backup-fsa/sysimg-t3p5g965-debian-20100131-0716.fsa
* 出力の例を次に示します。
* # fsarchiver archinfo /backup/backup-fsa/sysimg-t3p5g965-debian-20100131-0716.fsa
* ====================== archive information ======================
* Archive type: filesystems
* Filesystems count: 2
* Archive id: 4b610c6e
* Archive file format: FsArCh\_002
* Archive created with: 0.6.6
* Archive creation date: 20100131-07:16:35
* Archive label: debian-backup
* Compression level: 7 (lzma level 1)
* Encryption algorithm: none
* ===================== filesystem information ====================
* Filesystem id in archive: 0
* Filesystem format: ext3
* Filesystem label: boot
* Filesystem uuid: d76278bf-5e65-4568-a899-9558ce61bf06
* Original device: /dev/sda1
* Original filesystem size: 961.18 MB (1007869952 bytes)
* Space used in filesystem: 356.86 MB (374190080 bytes)
* ===================== filesystem information ====================
* Filesystem id in archive: 1
* Filesystem format: ext3
* Filesystem label: debian
* Filesystem uuid: 4b0da78f-7f02-4487-a1e2-774c9b412277
* Original device: /dev/vgmain/snapdeb
* Original filesystem size: 11.81 GB (12682706944 bytes)
* Space used in filesystem: 7.11 GB (7635599360 bytes)

### マルチスレッド圧縮

* FSArchiverは、マルチスレッド圧縮もサポートしています。マルチコアプロセッサ（例：デュアルコアまたはクアッドコア）を使用している場合は、すべてのコアが使用されるように、いくつかの圧縮ジョブを作成する必要があります。これにより、圧縮または解凍がはるかに高速になります。たとえば、デュアルコアを使用している場合は、オプション-j2を使用して、2つのコアの能力を使用する2つの圧縮スレッドを作成する必要があります。クアッドコアCPUを使用している場合は、他のプログラムのために1つのコアをアイドル状態のままにする場合を除いて、オプション-j4をお勧めします。その場合、-j3を使用できます。マルチスレッド圧縮の例を次に示します。
* fsarchiver -j3 -o savefs /mnt/backup/gentoo-rootfs.fsa /dev/sda1
* アーカイブをいくつかのボリュームに分割する
* アーカイブファイルが非常に大きい場合は、いくつかの小さなファイルに分割することをお勧めします。たとえば、バックアップのサイズが8GBで、DVD + RWディスクに保存する場合は、アーカイブを4.3GBのボリュームに分割すると便利な場合があります。ファイル分割はFSArchiver-0.3.0以降でサポートされています。アーカイブを作成するときに使用するには、オプション-sを使用して、各ボリュームに必要なサイズをメガバイト単位で指定する必要があります。
* fsarchiver savefs -s 4300 /data/backup-rhel-5.2-fsa033.fsa /dev/sda1
* 最初のボリュームには常に.fsa拡張子が付いています。次のボリュームの名前は.f01、.f02、.f03、…で終了します。アーカイブを復元するときは、コマンドラインで最初のボリュームへのパスを指定するだけで、次の場合は自動的に次のボリュームが使用されます。それらは同じディレクトリにあります。それ以外の場合は、ボリュームの別の場所を指定できるプロンプトが表示されます。

### 実行環境

* FSArchiverでは、ファイルシステム属性を保存するためにファイルシステムツールをインストールする必要があり（fsarchiver savefsを実行する場合）、 fsarchiverestfsを実行するときにファイルシステムを再作成するためにこれらのツールも必要です。とにかく、インストールする必要があるのは現在のファイルシステムのツールだけです。つまり、ext3ファイルシステムでのみ作業する場合は、xfsprogsをインストールする必要はありません。
* これらの理由から、すべてのシステムツールがインストールされている環境からFSArchiverを実行することをお勧めします。最適な環境は SystemRescueです。これは、すべてのLinuxファイルシステムツールとごく最近のFSArchiverバージョンが付属しているためです。
* SELinuxでラベル付けされたファイルシステムを保存するときは、FSArchiverを実行しているカーネルでSELinuxが有効になっていないことを確認することも重要です。実行中のコンテキストにSELinuxに関連する拡張属性を読み取るのに十分な権限があることが確実な場合は、SELinuxを有効にしてFSArchiverを使用できます。もう1つのオプションは、selinux = 0オプションを使用して一時的に起動することですが、SELinuxを再度有効にして起動したときに、（少なくともRHELでは）完全なファイルシステムのラベル変更を強制するという副作用があることに注意してください。その他の場合、カーネル はsecurity.selinuxの実際の値の代わりにラベルなしを返す可能性があります属性。その場合、FSArchiverはこれらの属性を保持せず、ルートファイルシステムを復元するときにシステムが機能しないか、SELinuxにラベルを付け直すように依頼する必要があります。SystemRescueからFSArchiverを使用する場合、SELinuxサポートはデフォルトで無効になっているため、その環境からFSArchiverを使用する場合、SELinuxラベルは保持されます。

### ファイルシステムの検出

* FSArchiverは、コンピューターのすべてのディスクにインストールされているファイルシステムを検出できます。これは、デバイス名がわからないときにパーティションで作業する場合に非常に便利です。
* # **fsarchiver probe simple**
* [=====DEVICE=====] [==FILESYS==] [=====LABEL=====] [====SIZE====] [MAJ] [MIN]
* [/dev/sda1 ] [ext3 ] [boot ] [ 768.72 MB] [ 8] [ 1]
* [/dev/sda2 ] [reiserfs ] [gentoo ] [ 12.00 GB] [ 8] [ 2]
* [/dev/sda3 ] [ext3 ] [data ] [ 350.00 GB] [ 8] [ 3]
* [/dev/sda4 ] [ext3 ] [backup ] [ 300.00 GB] [ 8] [ 4]
* [/dev/sda5 ] [lvm2pv ] [ ] [ 134.38 GB] [ 8] [ 5]
* [/dev/sda6 ] [lvm2pv ] [ ] [ 106.24 GB] [ 8] [ 6]
* [/dev/sdb1 ] [reiserfs ] [usb8gb ] [ 7.46 GB] [ 8] [ 17]

### コマンドラインとその引数

* ====> fsarchiver version 0.8.5 (2018-07-10) - http://www.fsarchiver.org <====
* Distributed under the GPL v2 license (GNU General Public License v2).
* \* usage: fsarchiver [<options>] <command> <archive> [<dev1> [<dev2> [...]]]
* <commands>
* \* savefs: save filesystems to an archive file (backup a device to a file)
* \* restfs: restore filesystems from an archive (overwrites the existing data)
* \* savedir: save directories to the archive (similar to a compressed tarball)
* \* restdir: restore data from an archive which is not based on a filesystem
* \* archinfo: show information about an existing archive file and its contents
* \* probe [detailed]: show list of filesystems detected on the disks
* <options>
* -o: overwrite the archive if it already exists instead of failing
* -v: verbose mode (can be used several times to increase the level of details)
* -d: debug mode (can be used several times to increase the level of details)
* **-A**: allow to save a filesystem which is mounted in read-write (live backup)
* -a: allow to save a filesystem when acls and xattrs are not supported
* -x: enable support for experimental features (they are disabled by default)
* -e <pattern>: exclude files and directories that match that pattern
* -L <label>: set the label of the archive (comment about the contents)
* -z <level>: legacy compression level from 0 (very fast) to 9 (very good)
* -Z <level>: zstd compression level from 1 (very fast) to 22 (very good)
* -s <mbsize>: split the archive into several files of <mbsize> megabytes each
* **-j <count>**: create more than one (de)compression thread. useful on multi-core cpu
* -c <password>: encrypt/decrypt data in archive, "-c -" for interactive password
* -h: show help and information about how to use fsarchiver with examples
* -V: show program version and exit
* <information>
* \* Support included for: lzo=yes, lzma=yes, lz4=yes
* \* Support for ntfs filesystems is unstable: don't use it for production.
* <examples>
* \* save only one filesystem (/dev/sda1) to an archive:
* fsarchiver savefs /data/myarchive1.fsa /dev/sda1
* \* save two filesystems (/dev/sda1 and /dev/sdb1) to an archive:
* fsarchiver savefs /data/myarchive2.fsa /dev/sda1 /dev/sdb1
* \* restore the first filesystem from an archive (first = number 0):
* fsarchiver restfs /data/myarchive2.fsa id=0,dest=/dev/sda1
* \* restore the second filesystem from an archive (second = number 1):
* fsarchiver restfs /data/myarchive2.fsa id=1,dest=/dev/sdb1
* \* restore two filesystems from an archive (number 0 and 1):
* fsarchiver restfs /data/arch2.fsa id=0,dest=/dev/sda1 id=1,dest=/dev/sdb1
* \* restore a filesystem from an archive and convert it to reiserfs:
* fsarchiver restfs /data/myarchive1.fsa id=0,dest=/dev/sda1,mkfs=reiserfs
* \* restore a filesystem from an archive and specify extra mkfs options:
* fsarchiver restfs /data/myarchive1.fsa id=0,dest=/dev/sda1,mkfs=ext4,mkfsopt="-I 256"
* \* restore a filesystem from an archive and specify a new label and a new UUID:
* fsarchiver restfs /data/myarchive1.fsa id=0,dest=/dev/sda1,label=root,uuid=5f6e5f4f-dc2a-4dbd-a6ea-9ca997cde75e
* \* save the contents of /usr/src/linux to an archive (similar to tar):
* fsarchiver savedir /data/linux-sources.fsa /usr/src/linux
* \* save a filesystem (/dev/sda1) to an archive split into volumes of 680MB:
* fsarchiver savefs -s 680 /data/myarchive1.fsa /dev/sda1
* \* save a filesystem and exclude all files/dirs called 'pagefile.\*':
* fsarchiver savefs /data/myarchive.fsa /dev/sda1 --exclude='pagefile.\*'
* \* generic exclude for 'share' such as '/usr/share' and '/usr/local/share':
* fsarchiver savefs /data/myarchive.fsa --exclude=share
* \* absolute exclude valid for '/usr/share' but not for '/usr/local/share':
* fsarchiver savefs /data/myarchive.fsa --exclude=/usr/share
* \* save a filesystem (/dev/sda1) to an encrypted archive:
* fsarchiver savefs -c mypassword /data/myarchive1.fsa /dev/sda1
* \* same as before but prompt for password in the terminal:
* fsarchiver savefs -c - /data/myarchive1.fsa /dev/sda1
* \* extract an archive made of simple files to /tmp/extract:
* fsarchiver restdir /data/linux-sources.fsa /tmp/extract
* \* show information about an archive and its filesystems:
* fsarchiver archinfo /data/myarchive2.fsa

# Linux Mint 20.3インストール

## USBドライブ(128GB)にインストール

* USBドライブ(32GB)にLiveイメージを作成する。
* Liveイメージを起動する。
* もう一つのUSBドライブ(128GB)を接続してそこにLinuxMintをインストールする。
* （インストーラのなかでext4のファイルシステムを作成、初期化する）
* インストールが終わったらUSBドライブ(32GB)を抜いて再起動する。
* インストール済のUSBドライブ(128GB)から起動する。
* OSの更新や各種設定、アプリケーションのインストール（Chrome, GitHubDesktop, JDK17, eclipse, NOTOフォント, qtorrent, Vim, fsarchiver）

## USBドライブ(128GB)の起動イメージをバックアップ

* USBドライブ(32GB)で起動する。
* USBドライブ(128GB)の中のマイクロSDを抜いてSDカードアダプタに差したものをセットする。
* HDDドライブ(2TB)をセットする。これはバックアップ用ドライブ。
* この時点でデバイスは以下のようになっている。
* /dev/sda WindowsのCドライブ
* /dev/sdb LinuxMint ライブ(USBドライブ(32GB))
* /dev/sdc バックアップ用ドライブ(HDDドライブ(2TB))
* /dev/mmcblk0 LinuxMint(USBドライブ(128GB))
* /dev/mmcblk0p0 LinuxMintのOSイメージ(ext4)

ここで以下を実行する。

sudo mkdir /mnt/backup バックアップ用ドライブのマウントポイント作成

sudo mount /dev/sdc2 /mnt/backup バックアップ用ドライブのマウント

sudo mkdir /mnt/backup/LinuxImageBackup バックアップ用フォルダ作成

sudo fsarchiver savefs /mnt/backup/LinuxImageBackup/linuxMint20-3.fsa /dev/mmcblk0p0

（15分程度待つ）

sudo umount /mnt/backup バックアップ用ドライブのアンマウント

## バックアップファイルでCドライブにインストール

Cドライブインストール後のパーティション

