

UNIXでTVを見よう

法林 浩之

(株)インターネット総合研究所
hourin@suplex.gr.jp

何の話をするのか

- UNIXマシンでTVを見る方法の紹介
 - 一応、FreeBSDとLinuxを対象とする
- もうちょっと具体的に
 - TVチューナーカード/ビデオキャプチャカードの選択
 - カーネルやデバイスドライバの設定方法
 - TV視聴アプリケーションの使い方
 - 他のマシンから(外出先を含む)リモートで視聴する方法

今日の守備範囲

- 「TVを『リアルタイムで』見る」ことに焦点を絞って紹介
- もうちょっと具体的に
 - 場所は自宅/外出先を問わない
 - とにかくオンタイムで見ることには執着
 - 画質は問わない(特に外出先では期待できない)
 - 録画については触れない(一応できるけど今回は割愛)

(個人的な)経緯

- 動機
 - 家具を減らしたい
 - UNIXマシンをいろんな用途に使いたい
- UNIXマシンでTVを見られるようになった
 - たぶん7年前ぐらい
 - そのうちテレビは壊れたので捨てて、それ以来買っていない
- 外出先でもTVを見たくなくなった
 - 会社、旅行先など
 - 特に、リアルタイムで見たいときがある
 - ▷阪神戦、プロレス生中継など
- 外出先でも見られるようにした
 - 1年前ぐらい
 - 昨年の阪神戦で大活躍

必要なもの

- 最低限必要なもの
 - BSDもしくはLinuxマシン
 - デイスブレイ(これがないとTV見られません)
 - TVチューナーカード/ビデオキャプチャカード
 - TV視聴アプリケーション
- 外出先で見るときに使うもの
 - 自宅のインターネット回線(常時接続)
 - ストリーミングシステム
 - ▷ 自宅: サバ、エンコーダ
 - ▷ 端末: プレーヤー
 - ▷ 今回はRealVideoを使用

PC本体の選定

- 基本的には、普通のデスクトップPCでよい
 - CPU: 遅くてもよい
 - メモリ: できれば多く
 - ハードディスク: 小さくて十分
- 要件
 - ディスブレイ: 800x600ぐらい出れば十分
 - ビデオカード: XFree86が動くもの、ビデオメモリが多い方がいい?
 - PCIスロット: TVチューナーカード/ビデオキャプチャカードを挿す
 - サウンドカード(オンボード可): TVチューナーの音声出力を取り込む
- 私の自宅マシンの例
 - 基本性能: PentiumII 400MHz、メモリ128MB、ディスク100GB×2
 - ディスブレイ: シヤープ LL-T1610W (1280x1024投影可)
 - ビデオカード: ATI 3D RagePro
 - サウンドカード: Sound Blaster 16

OSの選定

- BSD系
 - FreeBSD: ○
 - NetBSD, OpenBSD: たぶん○
 - Mac OS X: 製品を買う方が確実:-)
- Linux
 - カーネルは対応している
 - アプリケーション: 主なディストリビューションにはバイナリパッケージあり
- 今回使用するOS
 - FreeBSD 4.9-RELEASE
 - ▷ FreeBSD 5.xでも動くはず
 - Vine Linux 2.6
 - ▷ カーネルは2.4.22を使用
 - ▷ カーネル2.6でも動くはずだが試せなかったので省略

カード選び

- TVチューナーカード
 - 地上波TVの視聴ができる
 - 入力端子: TV映像(RF)、ビデオ映像(S-VideoやRCA(黄色))
 - ▷ 衛星放送チューナーから得た映像はビデオ入力へ
 - 出力端子: TV音声(ステレオ)→オーディオ装置/サウンドカードへ
 - ▷ 取り込んだ映像はPCIバス経由でTV視聴アプリケーションへ
- ビデオキャプチャカード
 - TVチューナーカードからTVチューナーを除いたもの
 - この手のカードには2種類ある
 - MPEGエンコードチップが載っているもの/載っていないもの
 - 今回はMPEGチップを使わないので、載っていないものを使用
- FreeBSDやLinuxで使えるカードは限られる
 - 対応しているチップの種類が少ない

FreeBSDで対応しているチップ

- Brooktree Bt848/849/878/879 (Bt8x8系)
 - 現在Brooktreeは買収されてConexantになっている
 - 最近はブランド名がFusionになっている
 - 実際に手に入るのはBt878あたりを使ったカード
 - かつてはよく使われていたが、現在は消えつつある
 - しかしこれしか選択肢がないので、使用
- その他の対応チップ一覧
<http://www.jp.freebsd.org/www.FreeBSD.org/ja/releases/5.2R/hardware-386.html>
 - 5.2用だが4.9もたいがい同じ

Linuxで対応しているチップ

- Brooktree Bt8x8系チップ
 - bttvドライバを使用
 - Linuxカーネル2.4に含まれている(たぶんもっと前からある)
- Philips SAA7130/7133/7134/7135 (SAA713x系)
 - saa7134ドライバを使用
 - Linuxカーネル2.4に含まれていない(パッチが必要)
 - Linuxカーネル2.6.2で入ったが、まだ開発途上
- Conexant CX2388x
 - cx88ドライバを使用
 - 同じくLinuxカーネル2.6.2で入ったが、まだ開発途上

結局どのカードを使えばいいの？

- 無難なのはBt8x8系チップ搭載カード
 - FreeBSDでもLinuxでも動く
 - 徐々に入手が難しくなっている
- Linuxの人はSAA713x系チップ搭載カードも利用可
 - カードによっていろいろいる問題があるらしいので要注意

現在販売されているカードで使えそうなもの

- Bt8x8系
 - GV-BCTV5/PCI(I/O DATA)
 - BT878A-STVPCI2(玄人志向) → 今回使用
 - AD-PCITV(アルファデータ)
 - CBP-AV(BUFFALO): TVなし
- SAA713x系
 - SAA7130-STVLP(玄人志向) → 今回使用
 - EX-VISION300TV(ELSA)
 - PrimeTV 7133LP(NOVA)
 - QSTV15(Canopus)

FreeBSD, LinuxでTVチューナーカードを使う

- やること
 - カーネルの再構築(デバイスドライバのインストール)
 - TV視聴アプリケーションのインストール
- 紹介順序
 - FreeBSDでのカーネル再構築
 - Linuxでのカーネル再構築(Bt8x8編)
 - Linuxでのカーネル再構築(SAA713x編)
 - TV視聴アプリケーションのインストールと利用

FreeBSDでのカーネル再構築手順

- 一般的な手順と同じでよい
 - カーネル設定ファイルに必要事項を記述

```
# cd /usr/src/sys/i386/conf/
# cp GENERIC tv
# vi tv
```
 - カーネルのコンパイルとインストール

```
# config tv
# cd .././compile/tv
# make depend; make; make install
```
 - デバイスファイルの作成

```
# cd /dev
# ./MAKEDEV bktr0 ←キャプチャ用
# ./MAKEDEV snd0 ←サウンド用
```
 - 再起動

```
# reboot
```

カーネル設定ファイルに追加する項目

- TVチューナー/ビデオキャプチャ関係

```
device smbus
device iicbus
device iicbb
device bktr
options OVERRIDE_CARD=0
options OVERRIDE_TUNER=4 ←TVチューナーの指定
```
- サウンド関係

```
device pcm
```

 - サウンドカードにより異なるので注意

カード/チューナーの指定

- OVERRIDE_CARD: /usr/src/sys/dev/bktr/bktr_card.h 参照

```
#define CARD_UNKNOWN 0
#define CARD_IO_BCTV3 17
```
- OVERRIDE_TUNER: /usr/src/sys/dev/bktr/bktr_tuner.h 参照

```
#define NO_TUNER 0
#define TEMIC_NTSC 1
#define PHILIPS_NTSC 4
#define PHILIPS_FR1236_NTSC 9 (FMチューナー付き)
```
- 該当するデバイスの番号を指定
 - わからないときは、設定しないか0を指定
- **sysctl**でも設定できる

```
# sysctl -w hw.bt848.card=17
# sysctl -w hw.bt848.tuner=4
```

 - 一度カーネルを再構築してから**sysctl**で調べ、値を定めたら再度カーネルを再構築するとよい

FreeBSDでのデバイス認識例

```
% cat /var/run/dmesg.boot  
(抜粋)  
bktr0: <BrookTree 878> mem 0xe2000000-0xe2000fff irq 10 at device 18.0 on pci0  
iicbb0: <I2C bit-banging driver> on bti2c0  
iicbus0: <Philips I2C bus> on iicbb0 master-only  
iicbus1: <Philips I2C bus> on iicbb0 master-only  
smbus0: <System Management Bus> on bti2c0  
bktr0: Card has no configuration EEPROM. Cannot determine card make.  
bktr0: Pinnacle/Miro TV, Philips NTSC tuner.
```

Linuxでのカーネル再構築手順(Bt8x8編)

- 一般的な手順と同じでよい
- カーネルの設定変更
 - # cd /usr/src/linux
 - # make menuconfig (あるいはmake xconfig)
- カーネルのコンパイルとインストール
 - # make dep clean bzImage modules modules_install
 - # cp arch/i386/boot/bzImage System.map /boot/
- デバイスファイルの作成
 - # cd /dev
 - # ./MAKEDEV video ← キャプチャャー用
 - # ./MAKEDEV audio ← サウンド用
- etc/modules.confの設定
 - # vi /etc/modules.conf
- 再起動
 - # reboot

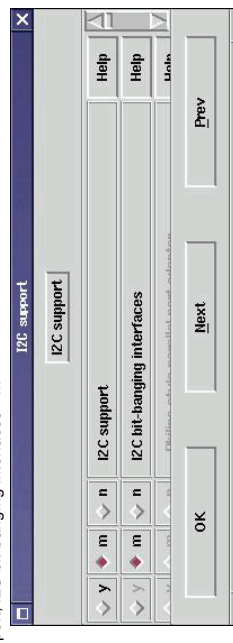
カーネルの設定変更(Bt8x8編)

- Multimedia devices → Video 4 Linuxをmに
 - その下のVideo 4 Linux → BT848 Video 4 Linuxをmに



カーネルの設定変更(Bt8x8編)(つづき)

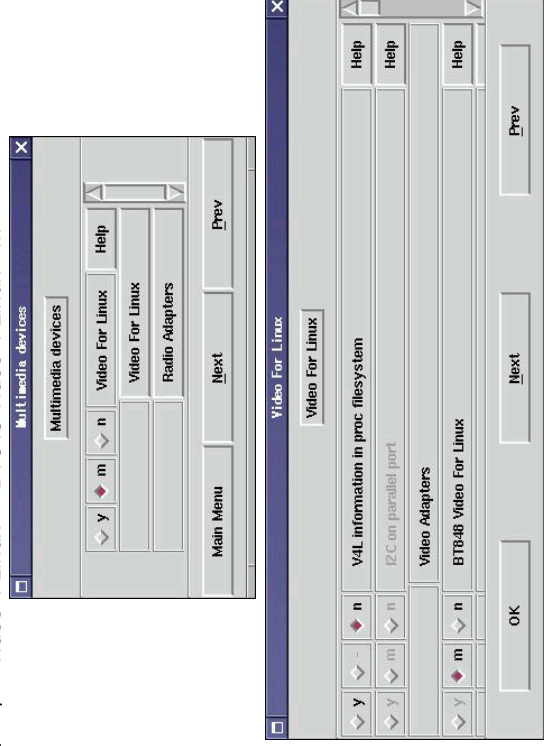
- Video 4 Linuxを使うために必要な設定
 - Character devices → I2C support
 - ▷ I2C support, I2C bit-banging interfacesをmに



- サウンド関係の設定
 - Sound Card Supportをyに
 - 持っているサウンドカードに対応した設定をする
 - デバイスドライバはOSSを使用
 - ▷ ALSAだと後で困るので

カーネルの設定変更(Bt8x8編)

- Multimedia devices → Video 4 Linuxをmに
 - その下のVideo 4 Linux → BT848 Video 4 Linuxをmに



/etc/modules.confの設定(Bt8x8編)

- BT878A-STVPCI2使用時の追加項目
 - alias char-major-81 bttv ←デバイス番号
 - options bttv card=78 ←キヤプチャカード番号
 - options tuner type=2 ←TVチューナー番号
- カード/チューナーの番号は
/usr/src/linux/Documentation/video4linux/bttv/CARDLISTを参照
- キヤプチャカード番号の例
 - card=0 - ** UNKNOWN/GENERIC ***
 - card=78 - Jetway TV/Capture JW-TV878-FBK, Kworld KW-TV878RF
 - card=81 - GV-BCTV5/PCI (IO/DATA)
 - card=99 - AD-TVK503 (アルファデータ)
- TVチューナー番号の例
 - type=2 - Philips NTSC (F11236,FM1236 and compatibles)
 - type=4 - NoTuner
 - type=31 - SHARP NTSC_JP (2U5JF5540)
 - type=40 - HITACHI V7-J180AT

bttvドライバ使用時のデバイス認識

- モジュールとして組み込んだときは、デバイス使用時に認識される
 - % dmesg
(抜粋)
 - Linux video capture interface: v1.00
 - bttv: driver version 0.7.104 loaded
 - bttv: Bt8xx card found (0).
 - bttv0: using: BT878(Jetway TV/Capture JW-TV) [card=78,insmod option]
 - i2c-core.o: adapter bt848 #0 registered as adapter 0.
 - bttv0: using tuner=5
 - i2c-core.o: driver i2c TV tuner driver registered.
 - tuner: probing bt848 #0 i2c adapter [id=0x10005]
 - tuner: chip found @ 0xc0
 - tuner(bttv): type forced to 2 (Philips NTSC (F11236,FM1236 and ...)) [insmod]
 - bttv0: registered device video0
 - bttv0: registered device vbi0
 - bttv0: registered device radio0

Linuxでのカーネル再構築手順(SAA713x編)

- 普通の再構築に比べて手間が増える(※の部分)
 - ※カーネル2.4のソースコードにパッチを当てる
 - カーネルの設定変更
 - カーネルのコンパイルとインストール
 - デバイスファイルの作成
 - (慎重にやるならここで再起動)
 - ※saa7134ドライバのインストール
 - /etc/modules.confの設定
 - 再起動

カーネル2.4のソースコードにパッチを当てる

- カーネル2.4.22以上のソースコードを用意
 - # cd /usr/src
 - # tar xjf linux-2.4.22.tar.bz2
 - # ln -s linux-2.4.22 linux
- Video for Linux 2を使うためのパッチを入手
 - 説明: <http://bytesex.org/v4l/>
 - パッチ: <http://bytesex.org/patches/2.4.22-3/patch-2.4.22-kraxel.gz>
- パッチをあてる
 - # bzip2 -dc patch-2.4.22-kraxel.gz | patch -p1

saa7134 ドライバのインストール

- ソースコードを入手
 - 説明: <http://bytesex.org/saa7134/>
 - ソースコード: <http://bytesex.org/saa7134/saa7134-0.2.10.tar.gz>
- コンパイルしてインストール
 - % tar xvfz saa7134-0.2.10.tar.gz
 - % cd saa7134-0.2.10
 - % make KDIR=/usr/src/linux
 - # make install
 - モジュールとして組み込まれる

/etc/modules.confの設定(SAA713x編)

- SAA7130-STVLP使用時の追加項目
 - alias char-major-81 saa7134 ←デバイス番号
 - options saa7134 card=10 ←キャプチャカード番号
 - options tuner type=2 ←TVチューナー番号
- キャプチャカードの番号は
saa7134-0.2.10/doc/CARDLIST.saa7134を参照
 - 0 -> UNKNOWN/GENERIC
 - 5 -> SKNet Monster TV
 - 10 -> Kworld/KuroutoShikou SAA7130-TVPCI
 - 13 -> ELSA EX-VISION 300TV
 - 14 -> ELSA EX-VISION 500TV
 - 16 -> AOPEN VA1000 POWER
- TVチューナーの番号はsaa7134-0.2.10/doc/CARDLIST.tunerを参照
 - 内容はBt8x8編で解説したものと同じ

saa7134 ドライバ使用時のデバイス認識

- モジュールとして組み込んだときは、デバイス使用時に認識される
 - % dmesg
(抜粋)
 - i2c-core.o: i2c core module
 - Linux video capture interface: v1.00
 - saa7130/34: v4l2 driver version 0.2.9 loaded
 - saa7130[0]: subsystem: 1131:0000, board: Kworld/KuroutoShikou (→つづく)
 - SAA7130-TVPCI [card=10,insmod option]
 - i2c-core.o: adapter saa7130[0] registered as adapter 0.
 - i2c-core.o: driver i2c TV tuner driver registered.
 - tuner: chip found @ 0xc0
 - tuner(btv): type forced to 2 (Philips NTSC (F11236,FM1236 and ...)) [insmod]
 - tuner: type already set (2)
 - saa7130[0]: registered device video0 [v4l2]
 - saa7130[0]: registered device vbi0

TV視聴アプリケーション

- あれこれ紹介せずに、代表的なものを紹介
 - FreeBSD: fxtv
 - Linux: xawtv

fxtv

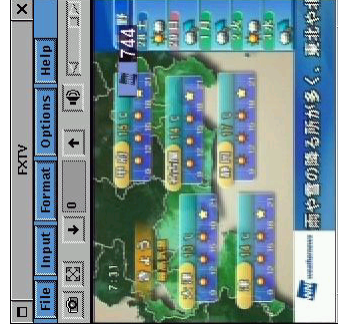
- http://people.FreeBSD.org/~rhh/fx_tv/
- Bt8x8用のTV視聴アプリケーション
- 最新版は1.03だが、現在は開発が止まっている模様
- FreeBSD, NetBSD, OpenBSDともバイナリパッケージあり
 - というわけでインストーラ方法は省略(⤵)

fxtvの設定方法

- fxtvには設定ファイル(.fx_tvなど)がないらしい
 - 膨大かつ長いコマンドオプションで設定
 - Xのリソースファイルで設定
 - ▷ オリジナルは/usr/X11R6/lib/X11/app-defaults/Fx_tv
 - 今回はXのリソースファイルで設定
 - Fx_tv*inputFormat: nts_cj
 - Fx_tv*defaultInput: tuner
 - Fx_tv*defaultAudioInput: tuner
 - Fx_tv*tunerMode: cable (アンテナの場合はantenna)
 - Fx_tv*cableFreqSet: jpn_cable
 - Fx_tv*cableStationList: 1-12 (CATVのチャンネル範囲に合わせる)
 - Fx_tv*antennaFreqSet: jpn_bcst
 - Fx_tv*antennaStationList: NHK(2) MBS(4) ABC(6) KTV(8) YTV(10)

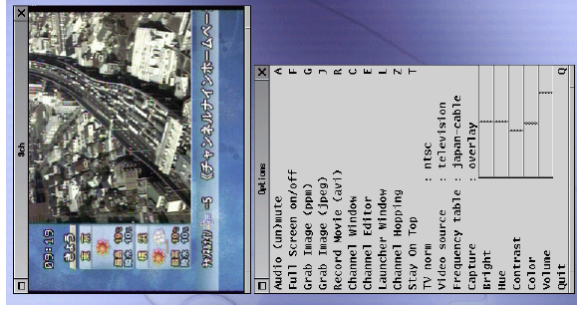
fx_tvの使い方

- チャンネルの切り替え: ↓ ↑ ボタン
- 入力系統の切り替え: Input → Tuner/S-Video
- 画面の拡大
 - 起動時はオプションで指定
 - % fxtv -geometry 640x480
 - 起動後は×ボタンで拡大/縮小
- フルスクリーンで表示する方法は不明(ないかも)
- 大画面で見たいときは
 - あらかじめXFree86を低解像度で表示できるように設定
 - xvidtune -nextで解像度を切り替える
- 画像の取り込み、録音、録画もできる



xawtv

- <http://bytestex.org/xawtv/>
- saa7134ドライバと同じ作者によるもの
- LinuxでもFreeBSDでも動く
- 最新版は3.91
 - rpm, dpkg, FreeBSDのportsにもあり
 - VinePlusで提供されているrpmはバージョンが古い
- ソースコードからのインストーラ手順
 - % configure
 - % make
 - # make install



xawtvの設定

- xawtvにて設定
 - [global]
 - ← チャンネルに関係ない設定
 - freqtab = japan-cable ← アンテナ受信ならjapan-bcast
 - [defaults]
 - ← 各チャンネルに対するデフォルト設定
 - norm = ntsc
 - input = television
 - [1ch]
 - channel = 1
 - key = 1
 - [12ch]
 - channel = 12
 - key = plus
 - [skape]
 - input = svideo
 - key = 0

.xawtv設定上の注意

- 一部の設定項目において、大文字が有効なときと小文字が有効なときがある
 - 小文字: ntsc,television,svideo,compisite1
 - 大文字: NTSC,Television,S-Video,Composite1
 - 最近のバージョンでは小文字表記らしいのだが...
- どちらか片方しか有効にならない(もう片方だとエラーになる)
- xawtv起動時に怒られたら修正する

xawtvの使い方

- チャンネル/入力系統の切り替え: マウス左クリック → 選択
- 画面サイズは起動時のオプションで
 - % xawtv -geometry 640x480
- フルスクリーン表示: 画面内でF ofxvの所で紹介した方法でXを低解像度にしておくと大画面になる
- 画像の取り込み、録音、録画もでき
 - XFree86起動時にXvideoを有効にしている
と取り込まない → xawtv -noxv で起動すれば取り込める



xawtv使用上の注意

- japan-cableに設定すると13ch以降しか映りかからない
 - ソースコードに1-12chの周波数データが載っていないため
- 対策: xawtv-3.91/frequencies/intsc-cable-ip.listを修正してコンパイル
- 修正箇所: ファイルの先頭に以下を追加

[1]	freq = 91250	[7]	freq = 189250
[2]	freq = 97250	[8]	freq = 193250
[3]	freq = 103250	[9]	freq = 199250
[4]	freq = 171250	[10]	freq = 206250
[5]	freq = 177250	[11]	freq = 211250
[6]	freq = 183250	[12]	freq = 217250

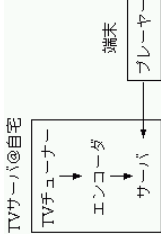
(右上に続く)
- 参考: 地上波とCATVの周波数一覧
 - <http://www2.fctv.ne.jp/~yodegura/catvband.htm>

v4lctl (V4LCTL)

- Video 4 Linuxデバイス操作コマンド
- xawtvの付属品
- チャンネルの操作
 - % v4lctl setchannel 1ch
- 入力系統の操作
 - % v4lctl setinput svideo
- チャンネル名や入力系統は.xawtvを参照して動作する
- ところが、v4lctlとxawtvで.xawtvの記法が異なることがある
 - ntsc/NTSC,svideo/S-Videoなど
 - 対策: 設定ファイルを2種類用意し、切り替えて使用
- リモートで操作するときに役立つ

リモートでTVを見る方法

- 基本アイデア
 - TVチューナーカードから得た映像/音声をストリーミングで流し、端末で受信する
 - 今回はReaVideoを使う方法を紹介
- システム構成



- この方法の特徴
 - さままな接続形態に対応できる
 - ▶ 会社、旅先、移動中など
 - 映像だけでなく音声も届き、同期も取れる
 - 画質は期待できない(特に低速回線で見るとき)

リモートでTVを見るために必要なもの

- Helix Producer: エンコーダ
 - RealVideo9コーデックを使用
 - 必ずTVチューナーと同一マシンで動かす
- Helix Universal Server: サーバ
 - 今回はTVチューナーと同一マシンで動かすが、別マシンでも可
- RealPlayer: プレーヤー
 - TVチューナーと別マシンで動かす
 - 同一マシンにもインストールしておくと同動作確認時に役立つ
- 常時接続回線とグローバルIPアドレス
 - 外出先からTVサーバに接続するのに必要
- TVサーバはLinuxで構築
 - Helix ProducerにFreeBSD版がないため
- 端末はRealPlayerさえ動けばどんなOSでもよい
 - 今回はFreeBSDを使用

RealPlayerのインストール

- Windows, Macの人: RealOnePlayerを使用
 - RealVideo9コーデックに対応済み
- Linux, BSDの人: RealPlayer8(Linux版)を使用
 - BSDの人はLinuxエミュレータ使用
 - RealVideo9コーデックに未対応 → 後からライブラリを追加して対応

RealPlayer8のRealVideo9コーデック対応方法

- RealVideo9対応パッケージを導入
 - http://docs.real.com/docs/playerpatch/unix/rv9_libc6_i386_cs2.tgz
- 入手したファイルを展開
- RealVideo9用ライブラリをRealPlayer8用ライブラリのディレクトリにコピー
 - % cd rv9/codecs
 - % cp drv4.so.6.0 rv40.so.6.0 /usr/local/lib/RealPlayer8/Codecs
- RealPlayerを起動して動作確認
 - <http://www.realnetworks.com/solutions/leadership/realvideo.html>に行き、ビデオが見えたら成功
- FreeBSDのportsでRealPlayerをインストールすると、RealVideo9対応ライブラリが自動的に追加される

Helix Producerのインストール

- Helix Producer Basic(無償版)を入手
 - <http://www.jp.realnetworks.com/products/producer/basic.html>
 - Helix Producer Plus(有償版)に誘導されないように注意
- 入手したファイルを実行してインストール
 - expressインストールでよい
- 注意: サウンドドライバはOSSを使うこと
 - Helix ProducerはALSAに対応していないらしい

Helix Producerが使うデバイスの検出

```
% producer -pd
VIDEO
Device 00: Kworld/KuroutoShikou SAA7130-TV /dev/video0
Port 00: S-Video ← スカパー
Port 01: Composite1
Port 02: Television ← TVチューナー
AUDIO
Device 00: Sound Blaster /dev/dsp
Port 00: synth
Port 01: line ← スカパー
Port 02: mic ← TVチューナー
Port 03: cd
```

□ v4lctlでTVを起動
% v4lctl setstation 1ch

□ Helix Producerでファイルに出力(つまり録画)
% producer -ac 00 -ap 02 -vc 00 -vp 02 -cs 240x180 -ad '150k' -o test.rm

- -ac: オーディオデバイス番号
- -ap: オーディオポート番号
- -vc: ビデオデバイス番号
- -vp: ビデオポート番号
- -cs: 画面サイズ
- -ad: 帯域
- -o: 出力ファイル名

□ 適当な所でCtrl-Cを打って録画終了

□ RealPlayerでtest.rmを再生できればOK

Helix Producerの動作確認

Helix Universal Serverのインストール

- Helix Universal Server Basic(無償版)を入手
http://licensekey.realnetworks.com/rmforms/products/servers/eval/index.html?ulf=b
- 入手したファイルを実行してインストール
- インストール時にいろいろと指定する
 - ライセンスキー: 入手するとXML形式のファイルがメールで届く
 - インストール先のディレクトリ
 - ユーザー名とパスワード: エンコーダからのストリーム受付や管理画面での認証に使用
 - 各種ポート番号: PNM(デフォルト: 7070)、RTSP(デフォルト: 554)、HTTP(デフォルト: 80)、MMS(デフォルト: 1755)、管理画面(デフォルト: 乱数)
- 起動
cd (インストール先)
Bin/rmserver rmserver.cfg
- RealPlayerで動作確認
○ rtsp://serverhost.domain.jp/real9video.rm が再生できればOK

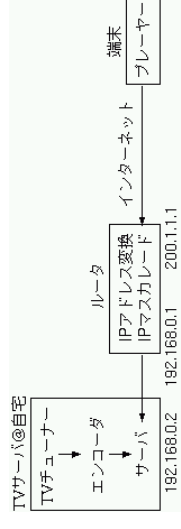
Helix ProducerからHelix Serverへの送出

- 同一マシンでProducerとServerを動かすとする
% producer -ac 00 -ap 02 -vc 00 -vp 02 -cs 240x180 -ad '28k,56k,256k'
-sp username:password@localhost:80/stream.rm
- -ad: 帯域
 - 複数指定すると、回線状況に応じて帯域が動的に変化(SureStream)
 - 指定できる数字は、\$INSTALLDIR/audiences/ を参照(最大3つ)
- -sp: Helix Serverとの接続方法、ストリーム名など
 - 書式: username:password@serverhost.domain.jp:port/stream.rm
 - username,password: Helix Serverインストール時に指定したもの
 - serverhost: Helix Serverのホスト名かIPアドレス
 - port: Helix Serverインストール時に指定したHTTPポート番号
 - stream.rm: ストリーム名(自由に指定)
- RealPlayerで動作確認
○ rtsp://serverhost.domain.jp/broadcast/stream.rm が再生できればOK

外出先からの接続

- 必要なもの
 - 常時接続回線
 - ▷ ストリームは上り回線を流れることに注意
 - グローバルIPアドレス
 - ▷ 最低1個、固定でも動的でもよい
- システム構成
 - TVサーバがグローバルIPアドレスを持っているとき
 - ▷ 端末からは、そのIPアドレスのHelix Serverにアクセスする
 - TVサーバがプライベートIPアドレスを持っているとき
 - ▷ 例えば、自宅にLANがあるとき
 - ▷ たいいていルータが入っているので、ここで細工する

自宅にLANがあるときの構成

- システム構成


```
graph LR
    TV[TVサーバ@自宅] --- R[ルータ]
    R --- I[インターネット]
    I --- T[端末]
    R --- S[エンコーダ]
    S --- SA[サーバ]
    subgraph Router
        R --- IP1[IPアドレス変換]
        R --- IP2[IPアドレス変換]
    end
    subgraph TVServer
        TV --- IP3[192.168.0.2]
    end
    subgraph Internet
        I --- IP4[192.168.0.1]
    end
    subgraph Terminal
        T --- IP5[200.1.1.1]
    end
```
- ルータでの設定
 - IPマスカレードの設定により特定ポートへのアクセスをTVサーバに向ける
 - IPマスカレードを設定するポート
 - RTSPのポート(デフォルト: 554/tcp)
 - HTTPのポート(デフォルト: 80/tcp)
 - 管理画面のポート(デフォルト: 乱数/tcp)
 - モニター画面のポート(デフォルト: 9090/tcp)
 - SSHのポート(デフォルト: 22/tcp)
 - ▷ TVマシンにリモートログインして操作

自宅LAN内TVの視聴方法

- sshでルーターのアドレスを指定(アドレス変換されてTVサーバへ)
% ssh 200.1.1.1
- TVを起動
% v4lctl setstation 1ch
- Helix Producerを起動しストリーム送出開始
% producer -ac 00 -ap 02 -vc 00 -vp 02 -cs 240x180 -ad '28k,56k,256k'
-sp username:password@localhost:80/stream.rm
- RealPlayerで動作確認
○ rtp://200.1.1.1/broadcast/stream.rm
- 実際にはTV視聴用シユルスクリプトを作成し、TVサーバにて実行
% ./stream.sh 1ch 28k,56k,150k
○ cronやatコマンドで指定時刻に起動することも可能

TV視聴用シユルスクリプト

```
#!/bin/sh
# $1: チャネル名、$2: 帯域
channel=$1
bitrate=$2
#すでにxawtvが動いているらkillする
killall xawtv
#v4lctlでTVを起動
cd $HOME
cp xawtv-big .xawtv
/usr/local/bin/v4lctl setstation $channel

#入力系統に応じた設定
case $channel in
skapa) #スカパー
# audio from line
audioport="01"
/usr/bin/aumix -l 60
/usr/bin/aumix -m 0
# video from S-video
videoport="00"
;;
*) #地上波
# audio from mic
audioport="02"
/usr/bin/aumix -l 0
/usr/bin/aumix -m 50
# video from Television
videoport="02"
;;
esac

#producerを起動
/usr/local/bin/producer -ac 00
-ap $audioport -vc 00 -vp $videoport
-cs 240x180 -ad $bitrate -sp
username:password@localhost:80/stream.rm
```

回線状況に応じた設定

- サーバへ端末の回線状況を推測し、エンコーダの帯域を設定する
- 低速回線で接続しているとき(PHSなど): 28k固定で設定
- 高速回線で接続しているとき(ホテルのLANなど): 150kぐらいの固定で設定
○ 回線状況により調整
- 回線状況が大きく変化するとき(会社のLANなど): 幅広く3種類選んで設定
○ 28k,64k,150kとか

映像素材に応じた設定

- 動きの多い素材(スポーツ中継など): フレーム数を重視する
 - 画面サイズを小さくする
 - producerコマンドで-vm smoothを追加する
 - ただし低速回線の場合は、フレーム数を稼ぐのはあきらめる
 - ▷ 画質が粗くなり鑑賞に耐えなくなる
- 動きの少ない素材(ニュース、ドラマなど): 各コマの画質を重視する
 - producerコマンドで-vm sharpを追加する

まとめ

UNIXマシンでTVを見る方法を紹介

□UNIXマシンでTVチューナーカードを動かす

○FreeBSD
▷Bt8x8系チップとbktvドライバ

○Linux

▷Bt8x8系チップとbtvドライバ

▷SAA713x系チップとsaa7134ドライバ

□TV視聴アプリケーションを動かす

○FreeBSD: fxtv

○Linux: xawtv

□リモートでTVを見る

○RealVideoによるライブ配信(見るのは自分だけ)

参考文献

- BSD magazine 第9号(2001年): bktvドライバ、fxtvの紹介
- UNIX magazine 2001年8月号「UNIX便利帖」: fxtvの紹介
- Software Design 2003年3月号「Linux, ビデオデジキ化計画」
- 「Linuxカーネル設定ガイド」翔泳社
- 「Helix Universal Server Advanced Book」BNN新社
- Linuxでのハードディスクレコーダ構築の記事
 - 日経Linux 2003年12月号
 - Software Design 2004年2,3月号

□今日の資料

○<http://www.suplex.gr.jp/~hourin/jus20040227/>

質問受け付けます

□難しいことはたぶんわかりません(^_^;)

○特にデバイスドライバの実装の詳細とか

□懇親会もありますのでぜひご参加ください