

# SFS による UML-KNOPPIX サービスの性能評価

<http://unit.aist.go.jp/it/knoppix>

須崎有康<sup>1)</sup>, 飯島賢吾<sup>1)</sup>, 丹英之<sup>2)</sup>, 後藤和弘<sup>3)</sup>

k.suzaki@aist.go.jp, k-ijima@aist.go.jp, tanh@alpha.co.jp, kazugoto@oita-ri.go.jp

産業技術総合研究所<sup>1)</sup>, (株)アルファシステムズ<sup>2)</sup>, 大分県産業科学技術センター<sup>3)</sup>

**概要** KNOPPIX のルートファイルシステムが収められている圧縮ループバックデバイス(cloop)ファイルセキュアなファイルシステム Self-certifying File System で WAN 公開することにより、KNOPPIX 対応 UserModeLinux のみあれば不特定多数のユーザが KNOPPIX を利用可能な環境を構築した。この環境でのネットワークトラフィック、起動時間を測定した。CD では判りにくい KNOPPIX 起動時のデータの読み出し量、必要バンド幅などを計測でき、圧縮の必要性、キャッシュの効果が確認できた。

## Evaluation of UML-KNOPPIX Service on SFS

Kuniyasu Suzaki<sup>1)</sup>, Kengo Iijima<sup>1)</sup>, Hideyuki Tan<sup>2)</sup>, Kazuhiro Goto<sup>3)</sup>

National Institute of Advanced Industrial Science and Technology<sup>1)</sup>, Alpha Systems Inc.<sup>2)</sup>, Oita Industrial Research Institute<sup>3)</sup>

**Abstract** : We customized UserModeLinux to boot KNOPPIX with compressed loop-back device "cloop" file. We open the cloop files with Self-certifying File System on WAN. This environment enables anonymous-users to run a customized UserModeLinux on WAN. We measured the network traffic and boot time which are difficult to measure with CD-boot. The results show mass of read-data and required bandwidth and confirm the effect of compress and cache.

### 1. はじめに

KNOPPIX<sup>[1,2]</sup>はハードディスクを使わない CD ブータブル Linux であるため、Windows プレインストールマシンでも手軽に試せる。KNOPPIX では統合デスクトップ環境 KDE、オフィスソフトウェア OpenOffice.org、Web ブラウザ Mozilla、メイラソフト Sylpheedなどをまとめ、1枚のCDのみでLinux環境を実行できる。既存のLinuxユーザばかりでなく、Linuxに躊躇していたユーザにも利用されるようになったが、幾つかの問題も明らかになった。現状では、isoファイルのCD焼付けやCDブートするためのBIOS設定が出来ないユーザも多い。真にノービスのユーザがフリーソフトを体感するには、環境整備が欠けている。この問題を解決するためにLinuxエミュレータのUserModeLinux<sup>[3]</sup>を使って起動できる環境を作成した<sup>[4]</sup>。この環境によりユーザは最新のKNOPPIXをCDに焼くことなく利用できるようになった。

我々は更にユーザの利便性を上げるために、不特定多数のユーザが Web ブラウザからクリックのみで WAN から UserModeLinux 上の KNOPPIX

を利用できるようにした。この環境構築のために WAN 対応のセキュアなファイルシステム (SFS: Self-certifying File System<sup>[5]</sup>)を利用した。本論文では環境構築の詳細とともに性能の報告を行なう。ここでは CD ブートでは測定が出来なかった読み出しデータ量や必要バンド幅などが測定できた。また、KNOPPIX で使っている圧縮ループバックデバイスの必要性やキャッシュの効果なども明らかになった。

### 2. KNOPPIX とは

KNOPPIX とはドイツの Klaus Knopper 氏が開発を進めている CD ブータブル Linux である。今までも CD ブータブル Linux は何種類か提案されてきたが、KNOPPIX は AutoConfig 機能によるデバイスの自動認識/設定が優れている点と独自の圧縮ループバックデバイス cloop を用いて 700M CD-ROM に 1.8G 程度のコンテンツを収録し、且つ、使いやすいデスクトップ環境にまとめた点が評価を得ている。

## 2.1 AutoConfig

KNOPPIX のデバイス自動認識・設定機能である AutoConfig(/etc/init.d/knoppix-autoconfig スクリプト)は、ハードディスク、ビデオデバイス、ネットワークデバイス、サウンドカード、USB デバイス、PCMCIA カード等を自動認識し、適切なドライバを組み込む。ハードディスク上にファイルシステムがあれば、その種別を認識して読み書き可能にする。ネットワークデバイスがあれば自動的に DHCP の設定まで行ない、即座に WWW のブラウザが利用可能となる。この knoppix-autoconfig スクリプトの中身は RedHat 系のデバイス自動認識ソフト kudzu からヒントを得て作成された hwsetup と mkxf86config などの設定スクリプトの組み合わせで実現されている。

## 2.2 cloop

cloop は圧縮機能が付いたループバックデバイスである。ループバックデバイスを用いることでファイルをファイルシステムとしてマウントできる。また、zlib を使って各ファイルの読み出し時に解凍を行なっている。解凍後はメモリにキャッシュされるので2度目以降のアクセスは高速である。

KNOPPIX では cloop ファイルにルートファイルシステムのイメージを格納している。cloop を使えばハードディスクと比べて読み出しの遅い CD ドライブを使っても、読み出しデータは約半分で済み、圧縮データの解凍は CPU に任せられるため、読み出し速度低下が緩和される。

## 2.3 CD 版 KNOPPIX のブート

CD のブートに必要なコンテンツは図1のようになる。ブートに関して必要なファイルはフロッピーイメージファイル"boot.img"と cloop ファイルの"KNOPPIX"である。まず、"boot.img"を用いてブートシーケンス"linuxrc"が始まる。ここでは RAM Disk をルート(/)としてブートし、cloop ファイル"KNOPPIX"をループバックでKNOPPIX にマ

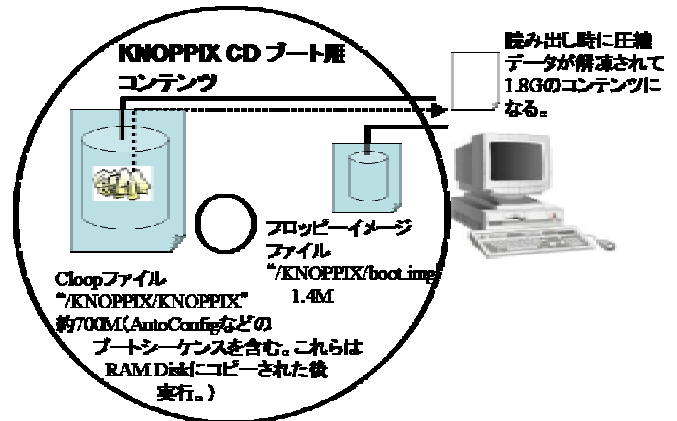


図1 CD 版 KNOPPIX のブート用ファイル

ウントする。その後 /KNOPPIX より必要なディレクトリ(usr, lib)がルート(/)に対してシンボリックリンクとして張られる。/etc 以下の書換えが必要な設定ファイルは/KNOPPIX/etc より RAM Disk の/etc にコピーを行なう。/home,/var など本質的に書換えが必要なディレクトリのためには tmpfs ファイルシステムをメモリ上に作成する。

上記の"linuxrc"によるブートシーケンスが終了後、制御を init に移し、/etc/init.d にある knoppix-autoconfig 等のブートシーケンスを走らせ、デバイス認識やドライバの組込みを行なう。

## 3. UserModeLinux-KNOPPIX

UserModeLinux を使って GUI を含めた KNOPPIX を仮想計算機のように利用する方法を説明する。

### 3.1 UserModeLinux とは

User Mode Linux (以下 UML と略記)は Jeff Dike 氏によって開発が進められている「ユーザプロセスとして実行可能な linux カーネル」である。UML 上で動作する Linux はゲスト OS と呼び、実ハードウェア上で動作する Linux をホスト OS と呼ぶ。UML はユーザプロセスとして Linux カーネルを利用できることにより、安全にカーネルのテストやデバッグを行なうことができる。残念ながら UML では仮想計算機のようにすべてのデバイスを用意していないこと、カーネルが UML 専用になるため通常の Linux ディストリビューションをそのままインストールすることはできないことなどの欠点がある。

### 3.2 cloop 対応 UML

UML では専用カーネルとなるため、KNOPPIX CD が提供するカーネルと置き換える必要がある。KNOPPIX では、cloop 以外は特殊なデバイスを利用しない。このため UML のカーネルで cloop デバイスのみ対応可能にすればよい。cloop のドライバは標準 UML kernel に入っていないので、ドライバを含んだ UML カーネルを作り直した。

### 3.3 ブートシーケンス

我々はカスタマイズした UML カーネルから KNOPPIX の cloop ファイルを利用してブート可能にした。これを UML-KNOPPIX と呼ぶ。

CD 版の KNOPPIX でもブートシーケンスは CD 内のフロッピーイメージ boot.img から始まる。boot.img のブートシーケンスの部分を UML 対応用 initrd として作成した(図 2)。この initrd を用いて既存の cloop ファイルをマウントし、ブート可能となる。ちなみに UML でも CD 版と同じにブートシーケンスが終了後、制御を init に移し、/etc/init.d にある knoppix-autoconfig 等を走らせ、デバイス認識やドライバの組み込みを行なう。

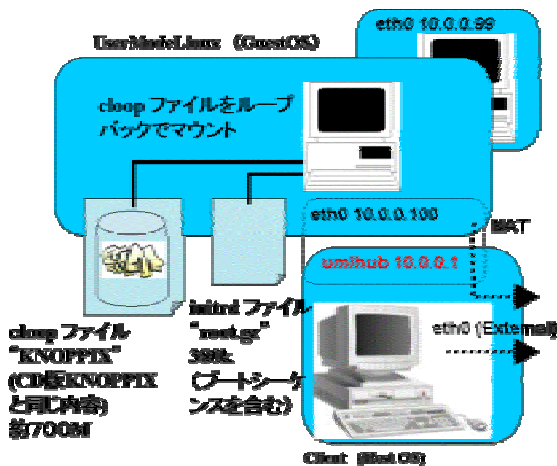


図 2 UML による KNOPPIX 起動概要

### 3.4 ネットワーク環境

UML では幾つかのネットワーク環境が利用できる。代表的なのが TUP/TAP だが、TUP/TAP では UML とホスト OS 間のみのネットワークで、複数

の UML-KNOPPIX を立ち上げた場合に相互の通信ができない。このため umlhub を利用した。このソフトウェアは UML とホスト OS の間に LAN 環境を提供する。dhcp の機能も入れたので、UML を立ち上げる度に IP アドレスは自動的に設定される。これにより複数の UML-KNOPPIX を立ち上げて相互に通信が可能になった。また、ホスト OS 側がルータになり、IP マスカレードによってゲスト OS は外部のネットワーク(ホスト OS の eth0)と接続可能になっている。

### 3.5 GUI 環境

UML では仮想ビデオカードを提供しないため、グラフィカルインターフェースがそのままでは利用できない。幸いにもネットワークが使える環境を提供するので、Xnest を利用してゲスト OS の GUI をホスト OS の X に写像する。Xnest は X client でありながら一方で X server として働くプログラムで、一つのウィンドウの中に X を立ち上げるような Window システムの入れ子ができる。これを利用して見た目は通常の仮想計算機と同じ機能を提供できる。複数の cloop ファイルを使った UML の起動を図 3 に示す。



図 3 UML による KNOPPIX の複数起動

### 3.6 ポーティング

UML-KNOPPIX は KNOPPIX 用の Debian パッケージのみではなく、TurboLinux、FedoraCore、Gentoo などのディストリビューション用のパッ

ページも作成した。それぞれのディストリビューションで UML-KNOPPIX をインストールすれば、1アプリケーションとして KNOPPIX が利用可能となる。

#### 4. WAN での UML-KNOPPIX 利用

UML を利用することで CD を作成する手間を省くことができた。しかし、cloop ファイルを 700M 一括ダウンロードして利用することに変わりはなく、ネットワークの負荷は低減されない。cloop ファイルを NFS で共有し、ランダムアクセス可能にすることで、各ユーザが必要ときに必要なデータをダウンロードできるようにすればネットワークのトラフィックが低減すると共に利便性が向上する。この方式は論文<sup>[4]</sup>で提案したが、NFS ではセキュリティの関係で LAN に限定されてしまう。WAN 環境に対応したファイルシステムを利用して、不特定多数のユーザが自由に使うことができる方式を提案する。

WAN 環境で利用可能なファイルシステムは Coda<sup>[6]</sup>, InterMezzo<sup>[7]</sup>, NFS4 など多く提案されている。しかし、残念ながらこれらのファイルシステムを利用するためのポートがセキュリティの問題から閉じられていることが多い。これに対し、既存のプロトコルで利用できる擬似ファイルシステムもある。プロトコルに ssh, ssh2 を利用するものとしては SHFS<sup>[8]</sup> (Shell File System)、SNFS<sup>[9]</sup> (Secure versions of the Network File System)、SFS<sup>[3]</sup> (Self-Certifying File System) などがある。このうち、利便性と性能に優れている SFS を採用した。

SFS は ssh2 プロトコル上に NFS プロトコルを乗せるファイルシステムである。SFS サーバへのアクセスは下記のようにマシン名と公開鍵をディレクトリパス名に含み、アクセスと同時に認証が行なえる。

```
>cd /sfs/@ks1.aist.go.jp,3rwmjfkvgsnbutptu9xyzf548zngq442/
      マシン名      公開鍵
```

クライアント側では sfsd をデーモンとして起動し

ておけば、ルート権限による明示的な mount 操作が必要ない。サーバでは厳密に読み出し専用設定ができるので、安全な運用もできる。SFS は Web ブラウザからも下記のような URL で利用可能であり、HP からのリンクでシームレスに参照できる。

file://sfs/@ks1.aist.go.jp,3rwmjfkvgsnbutptu9xyzf548zngq442/

シームレスな参照ができることを利用して、Web ブラウザから UML スクリプトをクリックするのみで SFS サーバの cloop を使って KNOPPIX を起動可能にした(図 4)。この実験サービスは下記 URL より利用することができる。

<http://unit.aist.go.jp/it/knoppix/uml/sfs/>

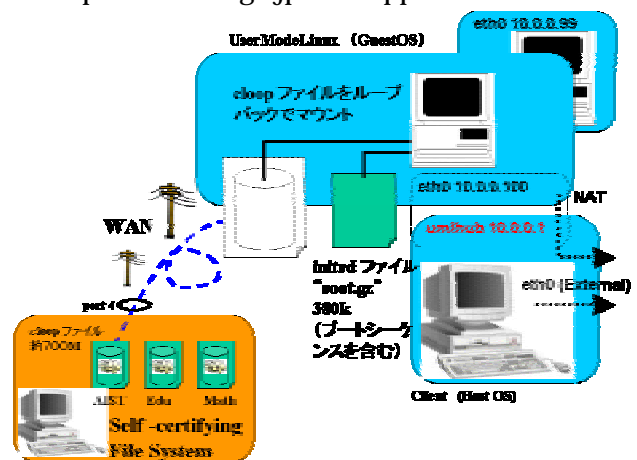


図 4 SFS からの UML-KNOPPIX の起動

SFS サーバのスクリプトからは KNOB(KNOPPIX for Bioinformatics<sup>[10]</sup>), KNOPPIX-EduTG<sup>[11]</sup>, yak 版<sup>[12]</sup>, DVD 版 KNOPPIX などが利用可能である。この環境では必要ときに必要なデータのみ転送されるのでネットワーク負荷も少ない。また、cloop による圧縮ループバックデバイスを介しているため、データ転送量も通常の約半分である。SFS サー

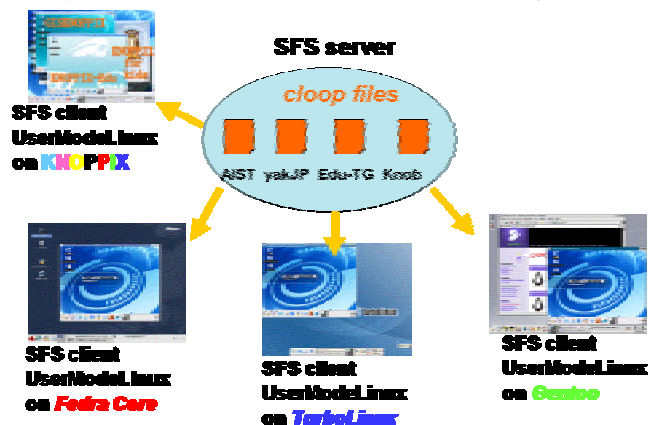


図 5 多くのディストリビューションからの UML-KNOPPIX の起動

バで DVD 版を公開しているのはこのことを確認可能にする意味も込められている。

UML-KNOPPIX は FedraCore、TurboLinux、Gentoo などのパッケージも作成したので SFS と組み合わせて同一のサービスを利用できる(図 5)。

## 5. 性能評価

SFS からの UML-KNOPPIX 起動の性能をいくつかの側面から調べた。実験装置としては SFS サーバ、クライアントそれぞれに Pentium4 2.4GHz、メモリ 1GB 搭載のマシンを使用し、これらをギガビット対応ハブへ接続した。SFS サーバには KNOPPIX 日本語版をハードディスクへインストールし、SFS サーバ関連のパッケージを apt-get で追加した。クライアントは KNOPPIX 日本語版 v3.3 を CD から起動した。

はじめに回線の伝送能力を確認するため、netperf を用いてクライアント・サーバ間で通信を行ない、このときの packets を tcpdump でキャプチャした。そして、netperf の出力結果と tcpdump による解析結果から求めたスループットを比較した。netperf によるギガビット対応ハブで接続された実験環境における測定結果は約 570Mbps であり、netperf の結果と tcpdump から求めた結果はほぼ一致していた。また、tcpdump で packets をドロップした割合は 0.01% 程度であり、実験環境において約 600Mbps ほどの通信を行なった場合でも、tcpdump による解析が可能であることを確認した。

UML 起動におけるバンド幅を評価するには、クライアントから SFS サーバ上の cloop を指定して UML 起動する際に、サーバ側で tcpdump で packets をキャプチャする。tcpdump では SSH2 に関する packets のみをキャプチャし、X が起動完了した後にキャプチャを停止させ、スループット等について解析した。

### 5.1 KNOPPIX 起動

KNOPPIX の起動時間を測定した。KNOPPIX は

比較のために日本語版 3.3(AIST)、KNOPPIX 日本語版 3.2 ベースの Edu 版、オリジナルの KNOPPIX3.1 ベースの GIS 版<sup>[13]</sup>を使った。それぞれデフォルトで KDE の起動が終了し、データ転送がなくなるまでを測定した。この結果を図 6 に示し、総転送量、起動時間を表 1 にまとめる。

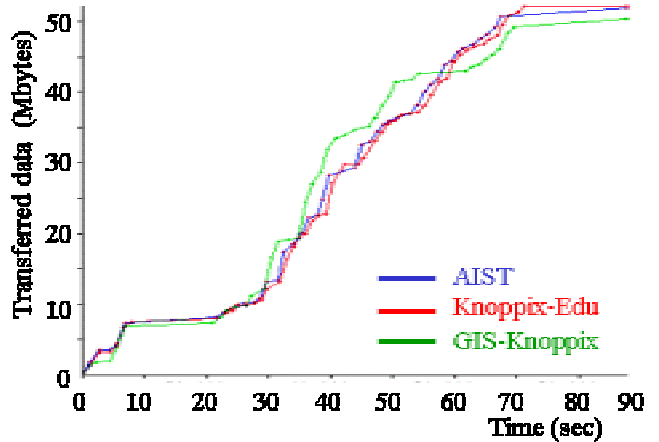


図 6 KNOPPIX 起動時のデータ転送量

表 1 起動時間とデータ転送量

	AIST 版	Edu 版	GIS 版
起動時間(sec)	90	90	90
総転送量 (MB)	51.6	52.3	50.3

この結果より、3 種類の KNOPPIX どれも起動時間が同じであった。総データ転送量についても 50MB 程度であった。また、図 6 より起動のデータ転送の振る舞いもほぼ同じであることが判った。

次に、1 秒単位で転送したデータをスループットと定義し、起動の必要なバンド幅を推定する。図 6 の測定結果をスループットに直したグラフを図 7 に示す。

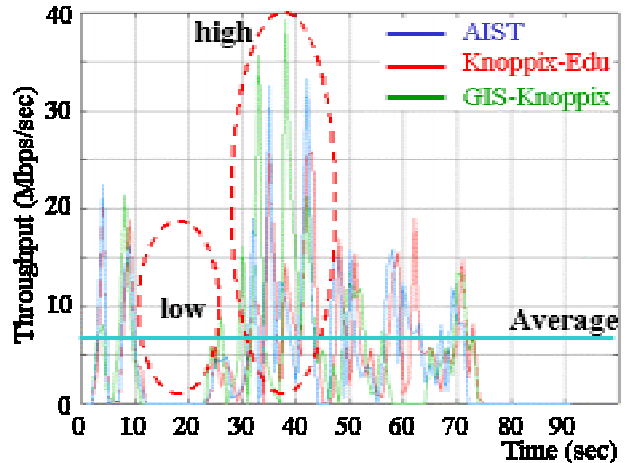


図 7 KNOPPIX 起動時のスループット

この結果より、データ転送にはかなり波があることがわかる。特に起動 10 秒から 20 秒にはデータ転送が無くなり、30 秒から 40 秒の間に大量のデータ転送が起こる。この部分を詳細に調べるとデータ転送が無くなった時点には KNOPPIX の hwsetup が実行されているためであった。hwsetup でデバイスを認識している間にはデータの読み出しは行なわれていないことが判った。UML 内部では仮想デバイスが固定しているため、hwsetup を行なう必要が無く、起動時間の短縮が計れる。また、30 秒から 40 秒の間の大量データ転送は X ウィンドシステムの起動であることが判った。この詳細を求めるために起動スクリプトとその実行時間の関係を次章で調べた。

また、平均スループットは 5.8Mbps であった。平均だけを見ると ADSL のみで十分に思えるが、データ転送には波があるので十分ではない。必要なバンド幅は最大スループットで決まる。この結果は表 2 のようになった。

表 2 起動時の最大スループット

	AIST 版	Edu 版	GIS 版
最大スループット(Mbps)	38.7	31.0	36.9

いずれも最大バンド幅が 40Mbps 以下であり、1GEther のバンド幅(netperf 測定で 570Mbps)をまったく使い切っていないことが判った。この原因は cloop を使うことで CPU の圧縮展開によって律速されていると思われる。今回利用した CPU は高速な P4 2.4Ghz なので、40Mbps 程度あればバンド幅は充分である。

## 5.2 起動手順とスループット

個々の起動スクリプトによるスループットの変化を測定するために、KNOPPIX 日本語版 3.3 の起動スクリプトで実行時刻を取れるように改良した。変更したスクリプトは/etc/init.d/内のスクリプトと/etc/X11/内の起動スクリプトである。この変更後のスループットを図 8 に示す。

この計測によりスループットが大きくなる点が

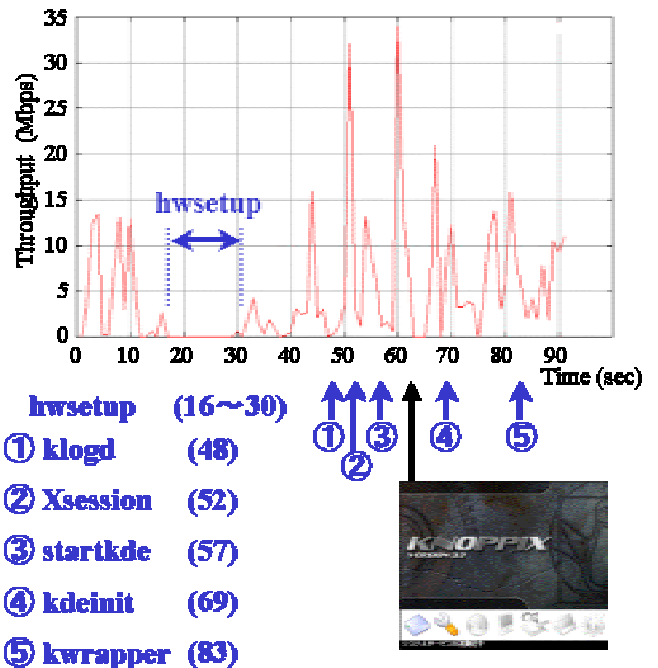


図 8 起動手順とスループット

二ヶ所あった。それは Xsession と startkde の直後であった。それぞれディスプレイの表示が切り替わる部分と連動している。特に startkde 直後では背景の図が現れるので、このデータ転送の負荷と思われる。この部分を除けば、20Mbps 程度でもバンド幅は充分であることが判った。

## 5.3 デスクトップの比較

デスクトップの違いによる起動時間の違いを測定した。デスクトップとしては KNOPPIX の起動時に選択できる KDE (デフォルト), xfce, twm,

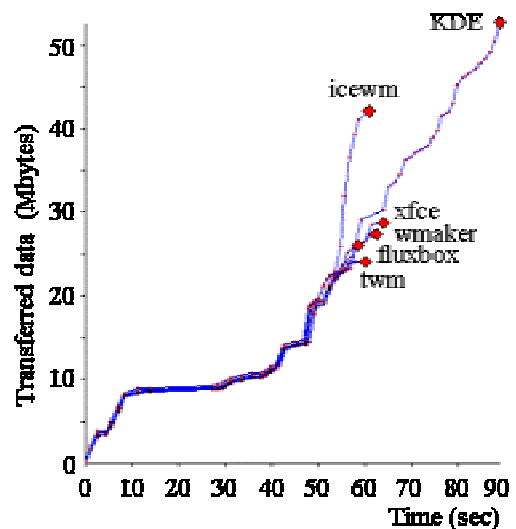


図 9 デスクトップ比較(データ転送量)

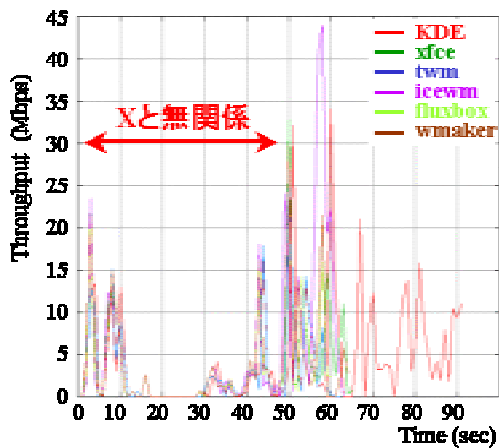


図 10 デスクトップ比較(スループット)

fluxbox, icewm, wmaker それぞれについて測定した。データ転送量およびスループットについて図 9 と図 10 に示す。

この結果より軽量デスクトップとして twm がデータ転送量としては最小であるが、起動時間としては fluxbox の方が若干早いことが判った。icewm も起動は早いですがデータ転送が大きく、最大スループットも最大 45Mbps となっていた。xfce, wmaker など KDE 以外の起動時間は 60 秒前後であり、大きく変わらないことが判った。KNOPPIX のデフォルトデスクトップである KDE は起動時間 90 秒、データ転送量 50MB と最大であり、KDE の使いやすさのためにデータ転送量で 20MB、起動時間で 30 秒ほど負荷が大きいことが判った。

#### 5.4 バンド幅の比較

バンド幅の影響を調べるために物理的ネットワークハブを変えることによって制限した実験を行なった。ハブは 1000M, 100M, 10M の 3 種類とした。データ転送量とスループットの測定結果を図 11 と図 12 に示す。

この結果より、1000M, 100M では違いが無いことが判る。これは 5.1 で求めた最大スループットが 40Mbps をカバーしていること一致する。しかし、10M ハブでは 40Mbps を下回るため、その影響がでていることがわかる。起動時間も 90 秒だったもの

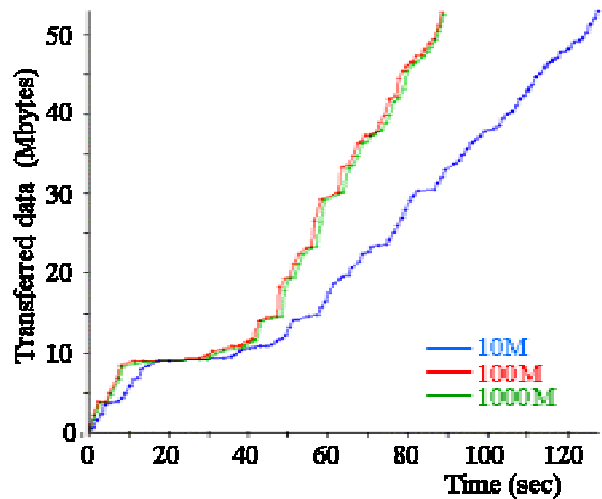


図 11 バンド幅比較(データ転送量)

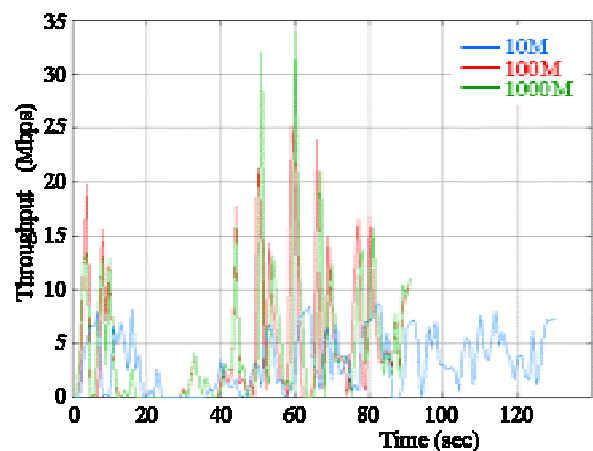


図 12 バンド幅比較(スループット)

が 130 秒と遅くなっている。

WAN からの利用を想定した場合、現在一般家庭に入っている ADSL は 40Mbps 以下のため影響があると思われる。100M の BFLET'S を利用した場合でも実測のバンド幅は低いので影響は避けられない。この対処は今後の課題である。

#### 5.5 圧縮の効果

KNOPPIX で使っている圧縮の効果を知るため、圧縮を行なわなかった場合と比較した。圧縮ありは zlib の cloop を使い、圧縮なしは通常のループバックデバイス loop を使った。データ転送量とスループットを調べた結果を図 13 と図 14 に示す。

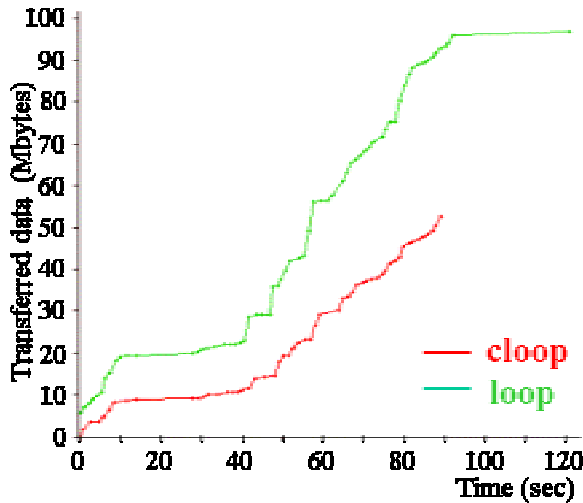


図 13 圧縮の効果(データ転送量)

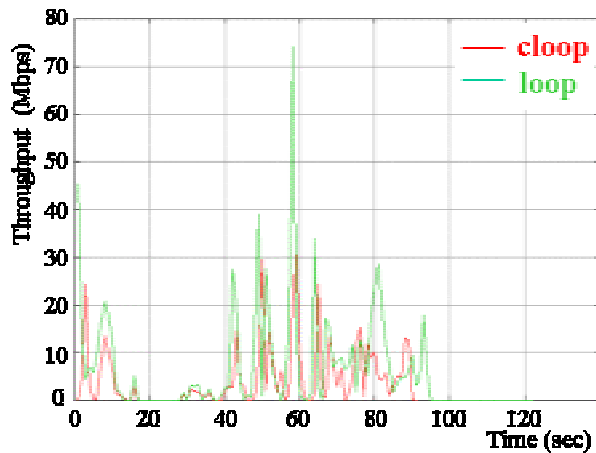


図 14 圧縮の効果(スループット)

測定の結果よりデータ転送量、起動時間、スループットともに圧縮がある cloop の方が良い性能であることが判った。この性能と性能比を表 3 にまとめた。起動時間は 90 秒であったものが 118 秒になり性能比は 1.3 倍である。総転送データ量は 50MB から 97MB になり 1.9 倍となった。最大スループットは 35Mbps から 80Mbps になり、2.3 倍のバンド幅を必要とすることが判った。この結果より圧縮を行なわなかった場合はネットワークの負荷が大きくなり、WAN では適さないことを示している。また、圧縮は CD においても容量が少なくなるばかりで無く、読み出しデータ量が少なくなり起動速度が向上することも推測できる。これは DVD においても圧縮は必要であることを示唆している。

表 3 圧縮の効果比較

	cloop	loop		比
起動時間(sec)	90	118		1.3
総転送量 (MB)	52	97		1.9
最大スループット(Mbps)	35	80		2.3

## 5.6 キャッシュの効果

UML は 1 アプリケーションなので、複数回利用する場合、クライアントマシン上のホスト OS である Linux のキャッシュ効果が期待できる。この効果を測定するために複数回起動のデータ転送量、起動時間を比較した。起動時間はいままでの測定はネットワーク上のデータ転送が無くなることで終了判定していたが、キャッシュの効果があると正確に判定できない。このため、2 回目の測定はクライアントの起動画面で KDE の設定が停止する時点を終了と判定した。測定方法が異なるので 2 回目の起動時間は参考値とする。この結果を表 4 に示す。表より 2 回目の起動ではキャッシュの効果は大きく、ネットワークでのデータ転送がほとんどなくなったことが判る。複数回目に起こったデータ転送が何の要求であるか解析しなければならないが、この結果はキャッシュが大きければネットワークの接続が切れても利用可能であることを示唆している。これは今後、coda や Intermezzo などの非常時接続を想定したファイルシステムの導入の参考データとなる。

起動時間は 90 秒から 80 秒になったが、期待していたほどに効果が無かった。データ転送はほぼなくなっているのに、律速しているものは別の要因である。圧縮の展開あるいは、Linux および KDE が起動時に行なう各種のデバイス認識のための同期があるので、これらが律速していると思われる。

表 4 キャッシュの効果

UML 起動	パケット数	バイト数	起動時間
1 回目	58647	52 MB	90sec
2 回目	7	899 B	(参考値)80sec



## 5.7 クライアントマシン性能による比較

クライアントマシンを変えて性能を測定した。測定に用いたマシンの仕様は表5になる。

表5 クライアントマシン仕様

	Dell PwerEdge 600SC	IBM ThinkPAD X30	Cacio FIVA 216	Toshiba Dynabook ss3380
CPU	P4 2.4G	P3 1.2G	Cruose 600M	P2 400M
Mem	1024M	768M	256M	128M
CDdrive	48 倍速	24 倍速	24 倍速	24 倍速
NIC	1000M	100M	100M	100M

各クライアントマシンでのデータ転送量と起動時間の関係を図15に示す。

この結果より、PowerEdge 600SC と ThinkPAD X30 では大きな違いが無いことが判った。マシンの仕様としては PowerEdge の方がすべての面において勝るがその影響が出ていない。キャッシュの測定と同様に起動時のデバイス認識による同期などが律速していると推定できる。つまり、SFS から UML-KNOPPIX を起動した場合、90 秒がほぼ最速であることが予想できる。

また、FIVA や Dynabook では起動時間の低下があることが判る。この原因一つは CPU 性能と推定される。また、Dynabook ではデータ転送量が増加しているが、これはメモリが少ないためにキャッシュデータを保持しきれずに、再読み込みが起きていると推定できる。いずれにしろ、FIVA や Dynabook 程度の仕様ではその性能が起動時間に現れ、使い勝手の低下が起こる。これを一つの指標に

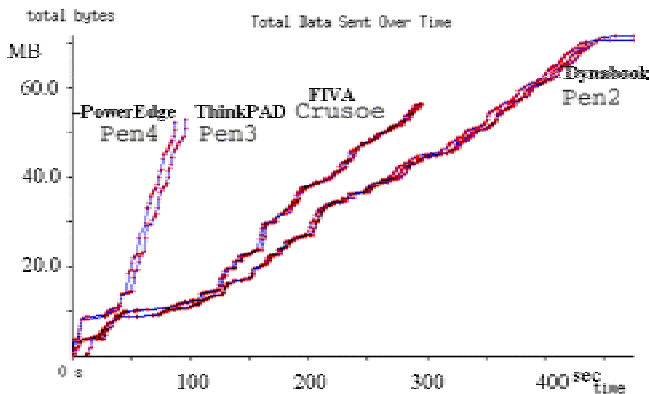


図15 クライアントマシンの違いによる性能

して、利用者に提示していきたい。

## 6.今後の課題

SFSを使ってWAN上のcloopファイルを公開し、UML-KNOPPIXが起動できることを確認した。本論文では、その性能評価を中心に述べたが、まだ評価しなければならない課題がある。この課題と今後の開発方向について述べる。

### 6.1 複数台起動の性能評価

今回の性能評価ではクライアントマシンが1台の場合のみを対象とした。この結果より必要なバンド幅が判り、複数台起動の推定できるようになった。今後は実際に複数台についての計測を行ない、推定が正しいこと、サーバが負荷に耐えられるかを確認する。

### 6.2 ファイルシステムの差分更新

cloopを使ったループバックデバイスは読み出し専用であるためファイルの更新ができない。この制約はCD配布を基本としているため仕方がない面があるが、パッケージの更新が一切出来ないなど利便性が悪い。読み出し専用ファイルシステムに対し追記更新を許す方法があるので、それらの技術を適用したい。例えば、仮想計算機のVMwareの仮想ディスクはundoableと言うモードを持ち、ゲストOSが起動してからシャットダウンするまでのファイルの更新部分のみを記録し、元の仮想ディスク自体を変えずファイルの更新が可能である。

UMLにも同様の機能(Copy On Write)があるためこれを利用する予定である。また、追記更新機能のあるファイルシステムもフリーソフトウェアとして開発されているので、その利用も検討する。候補としてはUnionFS<sup>[14]</sup>やOverlayFS<sup>[15]</sup>である。

### 6.3 coLinux

Windows上のLinuxエミュレータcoLinux<sup>[16]</sup>がイスラエルのDan Aloni氏を中心に開発されており、2004年1月25日にアナウンスされた。coLinuxは

まだ開発途中で不安定であるが、KNOPPIX の起動までは成功している。残念ながら SFS が Widows 対応していないため、すぐにクリックのみの KNOPPIX 利用はできないが、別の WAN 対応ファイルシステムと組み合わせて同様の環境を構築する予定である。

## 7.おわりに

CD 版 KNOPPIX がルートファイルシステムとして使う圧縮ループバックデバイスファイル(cloop)を WAN 対応のファイルシステム Self-certifying File System で公開した。これにより、不特定多数のユーザが UserModeLinux を用いて WAN から KNOPPIX を起動できるようになった。本論文では、この起動に関して詳細な性能評価を行なった。

性能評価の結果、最大スループットが 40Mbps であり、この程度のバンド幅があれば起動に影響がないことが判った。cloop 圧縮の効果は大きく、圧縮しない場合と比較して読み出しバンド幅が半分以下になることが判った。この結果は CD 版の KNOPPIX にも当てはまり、DVD のように容量が大きなメディアでも圧縮が有効であることを示している。メモリが大きい場合、2 回目の起動ではキャッシュが有効となり、ほとんどネットワークアクセスが無いことも確認できた。ただし、キャッシュは起動時間に大きな影響を与えなかった。これは圧縮の展開時間による律速、あるいは各種のデバイス認識のための同期によると思われる。この詳細解析は今後の課題である。

今後は、この性能評価をもとに改良を加える。性能的には現在一般家庭に普及している ADSL でストレスない起動ができるように改良する。また、付加的な機能としては、CD 版では出来なかったファイル更新を UserModeLinux の CopyOnWrite などを使って実現する。パッケージ管理ツールである apt-get を利用できるようにして、個々のユーザがカスタマイズまでネットワークからできるようにしたい。

## 謝辞

本研究の一部は情報処理振興事業協会(IPA)の平成 15 年度「未踏ソフトウェア創造事業」鵜飼 PM の「KNOPPIX ホスティング環境」の成果である。

## 参考文献&URL

- [1] KNOPPIX, <http://www.knopper.net/knoppix>
- [2] KNOPPIX 日本語版, <http://unit.aist.go.jp/it/knoppix/>
- [3] UML, "[http:// user-mode-linux.sourceforge.net/](http://user-mode-linux.sourceforge.net/)"
- [4] 須崎, 飯島, 丹, "KNOPPIX UserModeLinux を使った KNOPPIX マイグレーション", Linux Conference 2003. <http://lc.linux.or.jp/paper/lc2003/CP-16.pdf>
- [5] SFS, "<http://www.fs.net/sfswwww>"
- [6] coda, "<http://www.coda.cs.cmu.edu/>"
- [7] Intermezzo, "<http://www.inter-mezzo.org/>"
- [8] SHFS, "<http://shfs.sourceforge.net/>"
- [9] SNFS, "<http://www.math.ualberta.ca/imaging/snfs/>"
- [10] KNOB, "<http://knob.sourceforge.jp/>"
- [11] YakJP, "<http://sourceforge.jp/projects/ya-knoppix-jp>"
- [12] KNOPPIX-EduTG, "<http://www.eng.tohoku-gakuin.ac.jp/knoppix/>"
- [13] GISKNOPPIX, <http://www.sourcepole.com/sources/software/gis-knoppix/>
- [14] UnionFS, "<http://www.fsl.cs.sunysb.edu/project-unionfs.html>"
- [15] Overlay FS, "<http://ovlfs.sourceforge.net/>"
- [16] coLinux, "<http://www.colinux.org/>"