

Open School Platform プロジェクト ～大分市・豊後大野市地域プロジェクト

－ HTTP-FUSE KNOPPIX におけるキャッシュを利用した起動に関する検討 －

後藤和弘[†] 北川健司[‡] 江原裕幸[§] 中川俊哉^{††} 下村昭洋^{‡‡}

[†]大分県産業科学技術センター [‡](株)アルファシステムズ [§](財)ハイパーネットワーク社会研究所
^{††}(株)大分県自治体共同アウトソーシングセンター ^{‡‡}大分シーイーシー(株)

Open School Platform Project. - Oita, Bungo-Ohno Project -

Kazuhiro GOTO[†], Kenji KITAGAWA[‡], Hiroyuki EHARA[§], Toshiya NAKAGAWA^{††}, Akihiro SHIMOMURA^{‡‡}

[†]Electronics and Informatics Gr., [‡]Alpha Systems Inc., [§]Institute for HyperNetwork Society, ^{††}OLGO, ^{‡‡}OITA CEC Co.,LTD

要 旨

財団法人コンピュータ教育開発センターが実施した平成 18 年度 Open School Platform プロジェクト（大分市・豊後大野市地域プロジェクト）において、県内の高校を対象として OSS デスクトップ環境を実際の授業で実証した。また、ネットワーク起動方式においてサーバへのトラフィックの集中を防ぐ起動手法を提案し、有効性を確認した。

1. はじめに

学校教育現場ではオープンソースソフトウェア（以下、OSS）の利活用が期待されているが、広く普及する段階には至っていない。教育向けにカスタマイズされた KNOPPIX-Edu¹⁾ のダウンロードは大学が大半であり、高等学校はこれからという状況にある²⁾。業務での OSS 利活用に関する実証³⁾では非 OSS 環境への慣れによる影響が指摘されており、教育における早い段階で OSS を利用する機会に触れることが重要と考えられる。

OSS の普及に向けて、独立行政法人 情報処理推進機構は平成 16 年度に全国 17 校、約 3800 名を対象に Linux パソコンによる実証を実施した⁴⁾。また、財団法人コンピュータ教育開発センター（以下、CEC）は教育現場への OSS ベースの IT 環境の普及を目指して平成 17 年度から実証を行ない、平成 18 年度は導入のためのパッケージモデルや文教市場における OSS デスクトップに関するサポートとビジネスのあり方に関するモデルを提案して、実験後の自立的な運用を目指している。

本報告では CEC が実施した平成 18 年度 Open School Platform プロジェクト⁵⁾において、県内の高校を対象とした実証の内容を紹介する。また、実証で採用したネットワーク起動方式について、サーバへのトラフィックの集中を防ぐ手法を提案し、その有効性を検証した。

2. 実証の概要

2.1 Open School Platform プロジェクト

財団法人コンピュータ教育開発センターは教育現場への OSS の普及促進を目指し、平成 17 年度には全国 4 地域の小中学校 15 校を対象として、OSS デスクトップ端

末 700 台規模、約 6700 名の児童生徒が参加した実証実験を行なった⁶⁾。平成 18 年度はこれまでの成果を「Open School Platform パッケージ」として公開して全国の学校への普及促進を目指し、全国で 6 つのプロジェクトが採択され、小中高校 16 校が実証に参加した。

2.2 大分市・豊後大野市地域プロジェクト

大分市・豊後大野市地域プロジェクトは、大分県内の工業系高校 3 校を対象に、教育現場での利用実績が多い KNOPPIX を OSS デスクトップ環境とした。KNOPPIX は手軽に Linux を利用できる点やカスタマイズが容易な点がメリットで、大分を含む 3 テーマが採用した。また、非 OSS 環境を残したままでの OSS デスクトップの利用や、生徒の自宅学習にも容易に対応できる。参加校 3 校のうち、2 校が KNOPPIX を CD から起動する方式を、1 校がネットワークから起動する方式を実証し、各校の対象授業に応じてメニューやアプリ等をカスタマイズした。導入システムの概要を Fig.1 に示す。

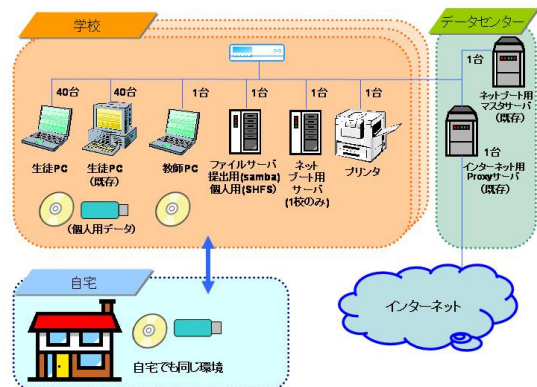


Fig.1 導入システムの概要

実証では 120 台のデスクトップ端末を新規導入し、教員と生徒合わせて約 700 人規模、10 教科（約 600 時間）の授業において OSS デスクトップ環境を利用した学習環境の検証をおこなった。Table 1 に OSS デスクトップ端末やサーバなど主要な機器の一覧を示す。また、各校における対象授業ならびに授業で活用した OSS アプリケーションを Table 2~4 にまとめた。

Table 1 実証における主要な機器の一覧

学校名	OSS 端末	ファイルサーバ	ネット起動サーバ	備考
情報科学高等学校	40 台	2 台	なし	CD 起動
鶴崎工業高等学校	40 台	1 台	なし	CD 起動
三重総合高等学校	40 台	1 台	2 台	ネット起動
合計	120 台	4 台	2 台	

Table 2 OSS アプリを活用した授業（情報科学高校）

科目名	OSS アプリケーション
ソフトウェア技術	OpenOffice.org Base (データベース) ※)以下、OpenOffice.org は OOo と略記
マルチメディア応用	Kwrite (エディタ), Firefox (ブラウザ)
プログラミング技術	Eclipse (統合開発環境)
課題研究	Firefox (ブラウザ), Nqc (Mindstorm の開発環境), OOo Impress (プレゼンテーション), Blender (3DCG 作成)

Table 3 OSS アプリを活用した授業（鶴崎工業高校）

科目名	OSS アプリケーション
工業基礎	Kwrite (エディタ), 十進 Basic (開発環境)
実習	Kwrite (エディタ), PICC (PIC マイコン用コンパイラ), PonyProg (PIC マイコン用ライター), gcc (コンパイラ), h8write (H8 マイコン用ライター)
計測制御	Kwrite (エディタ), gcc (コンパイラ), h8write (H8 マイコン用ライター)
課題研究	Kwrite (エディタ), PICC (PIC マイコン用コンパイラ), PonyProg (PIC マイコン用ライター), Firefox (ブラウザ), OOo Writer (ワープロ), OOo Impress (プレゼンテーション)

Table 4 OSS アプリを活用した授業（三重総合高校）

科目名	OSS アプリケーション
情報 A	OOo Writer (ワープロ), OOo Calc (表計算), Ooo Impress (プレゼンテーション), Firefox (ブラウザ)
情報処理	OOo Writer (ワープロ), OOo Calc (表計算)

2.3 実施内容の評価

授業における OSS デスクトップ環境や OSS アプリケーションの利用について、実証実験の前後で生徒と教員にアンケートやインタビューを行ない、システムの使いやすさや課題等について検証した。また、端末の利用状況をサーバ上に記録する仕組みを導入し、起動時間などを定量的に計測した。

アンケートでは、利用するうえでの課題や改善の要望はあるものの、約 7 割の生徒が「KNOPPIX を利用した授業は楽しかった」と回答している。また、教員の多くが「実証実験は教育的に意味があるものである」と回答されており、教員や生徒双方から寄せられた課題を解決することで、今後も普及が期待できると考えられる。以降では、アンケートにおいて課題とされた起動時間について改善策を検討する。

3. ネットワーク起動方式に関する検討

3.1 KNOPPIX の起動時間に関する課題

KNOPPIX は、CD や DVD から起動して手軽に Linux を利用できる点がメリットであるが、物理ドライブの性能や CD メディアに記録されるデータの配置の関係で起動時間の遅さが指摘されている。これを解決する仕組みとして LCAT(Live CD Acceleration Tool kit)⁷⁾ が提案されており、本実証の起動 CD でも採用されている。

一方、KNOPPIX のネットワーク起動方式は複数提案されており、実証では HTTP-FUSE KNOPPIX⁸⁾を採用した。この方式は、通常は CD や DVD に保存される 700MB 近い OS イメージファイルを分割・圧縮し、数百バイトから数百キロバイトのサイズにしたファイル（以下、ブロックファイル）をサーバ上に配置する。クライアントは、必要に応じてブロックファイルをダウンロードしながらデスクトップ環境を起動する。教室など 40 台のクライアントを一斉に起動する場合には短時間にサーバにアクセスが集中するため、サーバの NIC やスイッチングハブがボトルネックとなって、クライアントの起動時間に影響を与えてしまう。実証ではダウンロードするファイルのトータルサイズを抑えることで起動時間を短縮することができているが、本稿では異なるアプローチについて検討する。

3.2 キャッシュを利用した起動手法の提案

HTTP-FUSE KNOPPIX では、クライアントはダウンロードしたブロックファイルを RAM や USB メモリ等へキャッシュする。アプリケーションの起動などブロックファイルが必要となったときには、まずローカルのキャッシュ内を検索し、すでにブロックファイルをダウンロード済であればそれを利用する。また、見つからな

ければサーバへブロックファイルを要求することになる。そこで、クライアントにキャッシュされたブロックファイルを利用して、他のクライアントを起動する手法について提案する。

クライアントでは、キャッシュしたブロックファイルを他のクライアントから利用できるよう、Webサーバを起動する。そして、Webサーバのコンテンツとして、キャッシュしたブロックファイルを設定する。以降、このクライアントを「代替サーバ」と呼ぶ。一方、サーバにはブロックファイルを要求してきたクライアントのアドレスをデータベースに登録する処理を導入する。クライアントから要求があった場合にはデータベースを検索し、クライアントへブロックファイルを送信するか、代替サーバのアドレスを通知するかを決定する。1つのブロックファイルに対して複数の代替サーバを対応づけるとともに、クライアントからの要求に対してランダムに代替サーバを選択することで、特定の代替サーバへトラフィックが集中することを防ぐ。提案手法でブロックファイルを取得する仕組みを Fig.2 に示す。

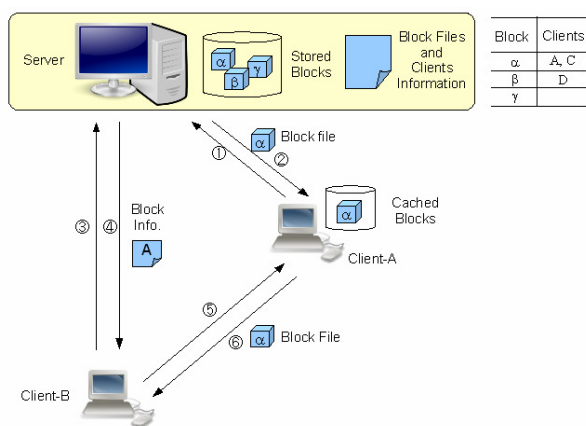


Fig.2 提案手法によるブロックファイルの取得

2 台のクライアントを起動する場合について、ブロックファイルを取得する手順を示す。ブロックファイルのファイル名は通常 32 文字からなる英数字であるが、以下では説明上、ファイル名を α と略記する。

- ① どのクライアントもブロックファイルをダウンロードしていない状態で、クライアント A が初めてサーバへブロックファイル α を要求する。
- ② サーバはクライアント A のアドレスとファイル名 α を対応づけてデータベースへ記憶するとともに、クライアント A にブロックファイルを送信する。クライアント A はこれ以降に代替サーバとなる。
- ③ つづいてクライアント B がサーバへブロックファイル α を要求する。

- ④ サーバはデータベースを検索し、ファイル α は代替サーバ A がダウンロードしていることから、クライアント B にはファイルではなく、代替サーバ A のアドレスを通知する。
- ⑤ サーバからの情報をもとに、クライアント B は代替サーバ A へブロックファイル α を要求する。
- ⑥ 代替サーバ A はブロックファイル α をクライアント B に送信する。

なお、ここでは 1 つのブロックファイルにつき、1 台の代替サーバを登録する場合について説明しているが、実際の教室などで運用する場合にはクライアントの台数やネットワーク機器など利用環境に応じて代替サーバの設定数を検討する必要がある。

4. 提案手法の評価

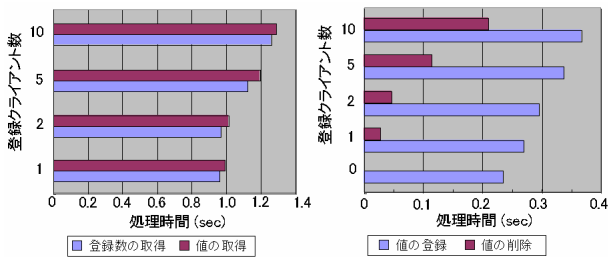
4.1 評価項目について

提案手法では、サーバにおいてデータベースの検索やクライアントへの送信内容を判断する処理を組み込んでおり、これらの処理がクライアントからのアクセスに与える影響について検証する必要がある。そこで、導入したデータベースの検索処理性能を計測するとともに、各処理を組み込んだ状態でクライアントからサーバへアクセスした場合のアクセス性能を計測・評価する。さらに、複数台のクライアント端末を一斉に起動した場合におけるサーバへのトラフィックを計測し、本手法の効果を検証する。サーバは CPU Opteron 3.0GHz, RAM 2GB のマシンに、OpenSUSE 10.1, Apache 2.2.0 をインストールしている。また、クライアントは Pentium4 2.66GHz, RAM 512MB のマシンを利用した。

4.2 データベースの処理性能

データベースには、キーと値を対応づけて管理する QDBM を採用した。このようなデータベースは、他にも gdbm や ndbm, Berkley DB などが提案されており、Sendmail や OpenLDAP, Postfix といったソフトウェアでも活用されている。

データベースに登録されたクライアント（代替サーバ）の台数による影響を検証するため、登録台数が 1, 2, 5, 10 台の場合についてキーの検索や、値の取得、登録、削除等の操作を行ない、それぞれの処理時間を計測した。HTTP-FUSE KNOPPIX におけるクライアントは、起動時に約 2000 個のブロックファイルをダウンロードするため、検証では登録 1 台あたり 2000 個の情報をデータベースに格納した。例えば、クライアントを 10 台登録した条件では、データベースには 20000 個のキー（および値）が登録された状態で検索などの各種操作を実行することになる。

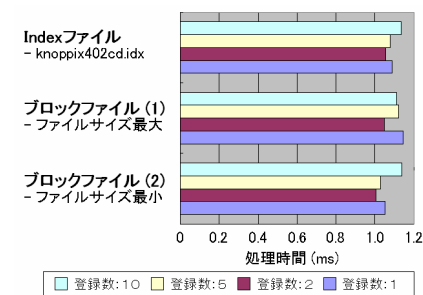


(a) キーの検索および値の取得 (b) 値の登録および削除
Fig.3 データベースの操作における処理時間

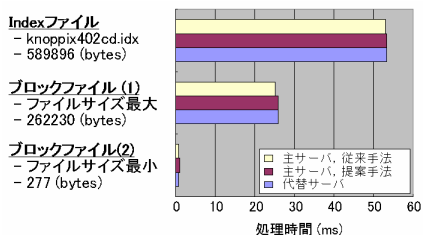
各処理における計測結果を Fig.3 に示す. Fig.3(a)はキーの検索, およびキーに対応する値の取得を 10000 回実行した場合の処理時間である. 登録台数が増えるにしたがって処理時間も増加しているが, 各操作 1 回あたりの処理時間は 100 マイクロ秒程度となり, 今回の検証の範囲では登録台数による差は小さいと考えられる. 一方, Fig.3(b)はデータベースへの情報の登録, そして削除における処理時間である. 1 台のクライアントをデータベースに登録 (または削除) する場面を想定し, それぞれ 2000 個のキーと値を登録 (または削除) している. 検索と比べて処理時間は長い, これらの操作は代替サーバのみが該当するため, 教室など 40 台規模で考えると処理時間全体に占める割合は少ない.

4.3 HTTP サーバへのアクセス性能

サーバへのアクセス性能を計測するため, Apache HTTP server benchmarking tool を利用して, 3 つの異なるファイルを対象としてデータベースの検索, およびブロックファイルのダウンロードにかかる処理時間を計測した. Fig.4 に計測結果を示す.



(a) データベースの検索



(b) ファイルのダウンロード

Fig.4 クライアントの要求に対する処理時間

対象とするファイルは, クライアントの起動時に初めに要求するインデックスファイル, そして, ダウンロードする約 2000 個のブロックファイルのうちでファイルサイズが最大のファイルと, 最小のファイルとした.

Fig.4(a)はクライアントからの要求に対して, 指定されたブロックファイルをデータベースから検索するときの処理時間である. データベースへ登録されたクライアント (代替サーバ) の台数が 1, 2, 5, 10 台の場合について処理時間を計測・評価した. 図から, 登録台数によって処理時間に若干の差があるものの, 台数との明確な相関関係は見られない. また, 検索処理ではファイルサイズは関係しないため, 各ファイルにおける処理時間はほぼ同じ数値となっていることがわかる. これに対して, ファイルをダウンロードする場合には Fig.4(b)のようにファイルサイズによって処理時間に明確な差が見られる. ブロックファイルを提供するサーバ (図中「主サーバ」) において, 従来手法では要求されたファイルを送信するのみであるのに対して, 提案手法ではデータベースの検索などの処理を行なうにもかかわらず処理時間はほぼ同じであった. このことは, データベースの検索等の処理はアクセス性能に影響を与えていないと考えられる. さらに, 主サーバと代替サーバではハードウェア性能が大きく異なるものの, 代替サーバからファイルをダウンロードする場合でも, 主サーバと同等のダウンロード性能であることを確認した.

4.4 複数クライアントの起動

提案手法の有効性を検証するため, Fig.5 に示す実験環境を構成した. 代替サーバを 2 台と設定して端末を一斉に起動し, サーバからクライアントへ転送されるパケットを tcpdump コマンドでキャプチャして解析した. また, クライアントは PXE ブートに対応させているために起動 CD 等を使わずに電源投入後にサーバから TFTP で minirt.gz をダウンロードする. クライアントが起動する KNOPPIX のバージョンは v4.0.2 である.

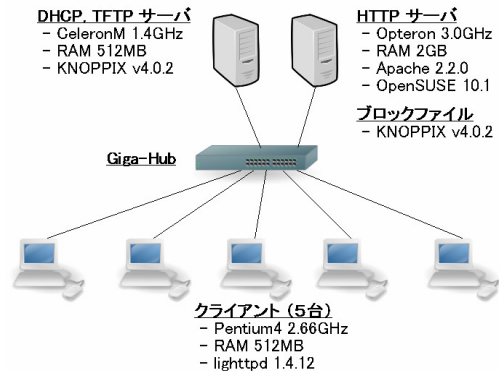
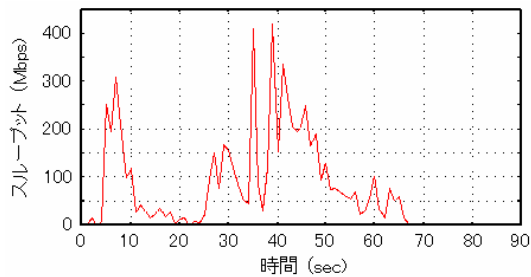
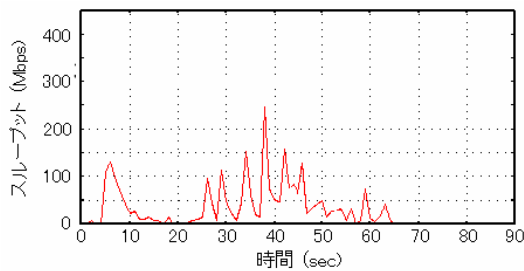


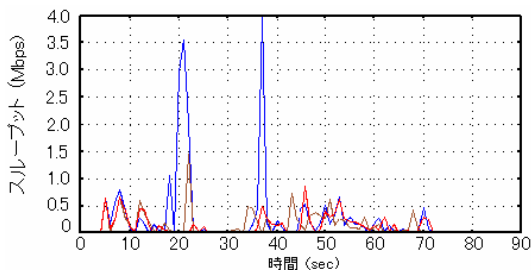
Fig.5 実験システムの構成



(a) 従来手法 (ファイルのダウンロード)



(b) 提案手法 (ファイルのダウンロード)



(c) 提案手法 (クライアント情報の取得)

Fig.6 スループットの計測結果

サーバからクライアントへ転送されたパケットについてスループットを解析した結果を Fig.6 に示す。従来手法では 5 台のクライアントすべてがサーバからブロックファイルをダウンロードするため、Fig.6(a)のように最大値が 400Mbps 以上になっている。これに対して、提案手法ではサーバからダウンロードするクライアントは 2 台のみであるため、スループットは最大で 250Mbps 程度 (Fig.6(b)) であった。また、代替サーバの情報を取得するクライアントでは、トラフィックは Fig.6(c)のように最大でも 4Mbps 程度であり、最も低いクライアントでは 1Mbps 以下であった。これらのクライアントは、サーバから取得した情報をもとに代替サーバへブロックファイルを要求しており、提案手法によってサーバへのトラフィックを軽減できることを確認した。

5. まとめ

学校教育現場における OSS の活用促進を図るため、県内の工業系高校 3 校を対象として実際の授業において OSS デスクトップ環境を利用した実証実験を行なった。アンケートやヒアリング、定量的なデータ計測など

を通じて普及に向けての課題の抽出ならびに、解決策について検討した。今後の普及に向けてはこれらの課題を解決するとともに、OSS の技術面やコスト面での優位性だけでなく、実際の授業でどのように活用できるか、どのように生徒に役立つか、といった視点を現場に示すことも重要であると感じる。実証を通じて、県高教研工業部会工業クラブ主催の「第 13 回生徒研究成果発表大会」において、KNOPPIX を利用したテーマが優秀賞に選ばれたことは有効性を示した実例と言える。

また、OSS デスクトップ環境をネットワークから起動する方式については、サーバへトラフィックが集中する課題を解決するため、クライアントにキャッシュされたファイルを利用する起動手法を提案し、その有効性を確認した。今後は 40 台規模のクライアントで検証が必要であるとともに、利用環境に応じて代替サーバの登録数を設定する基準や、地域 IX や WAN 環境等への展開についても検討していきたい。

謝辞

本報告は、経済産業省の委託を受け (財) コンピュータ教育開発センターが実施している平成 18 年度 Open School Platform プロジェクト (大分市・豊後大野市地域プロジェクト) における実施内容について紹介した。

参考文献

- 1) KNOPPIX-Edu, <http://www.knoppix-edu.org/>.
- 2) 千葉大作, 中川俊哉, 下村昭洋, 江原裕幸, 後藤和弘, 志子田有光, 早川栄一, “中川高等学校における OSS 活用実践とコミュニティ連携モデルの提案—Open School Platform Project 大分プロジェクトの取り組み—”, 電子情報通信学会技術研究報告, Vol.106, No.507, pp.31-36, 2007.
- 3) 2005 年度「自治体におけるオープンソースソフトウェア活用に向けての導入実証」成果, <http://www.ipa.go.jp/software/open/2005/stc/report/index.html>
- 4) 2004 年度「学校教育現場におけるオープンソースソフトウェア活用に向けての実証実験」成果, <http://www.ipa.go.jp/software/open/2004/stc/eduseika.html>
- 5) OSP ポータルサイト, <http://e2e.cec.or.jp/osp/>
- 6) 「Open School Platform」プロジェクト, <http://www.cec.or.jp/e2e/osp/index.html>
- 7) LCAT (Live CD Acceleration Tool kit), <http://sourceforge.jp/projects/lcat/>
- 8) HTTP-FUSE KNOPPIX, <http://unit.aist.go.jp/itri/knoppix/http-fuse/index.html>