

# ブートサーバによるクライアント PC の集中管理システム

## Centralized Management system of Client PCs Using Bootserver

武田 利浩†  
Toshihiro Taketa

阿部 康一‡  
Kouichi Abe

平中 幸雄†  
Yukio Hiranaka

### 1. まえがき

学生用の実習システムでは、その目的に応じて、たくさんの PC に同様の設定を施して、準備する必要がある。初期インストールやアップデートなど、煩雑な作業を必要とする。それらの作業を簡単にするために、古くからある方法として、NFS(Network File System)を使用したディスクレスシステムがある。しかし、この方法では、ネットワークの実習で、パケットのキャプチャを行うような場合には、不要なパケット (NFS のパケット) を観測してしまうため、不向きである。最近では、CD-ROM から起動可能な OS を用いて、そのような目的に使用する事がある。しかしながら、CD-ROM からの起動は遅く、実行中にも、CD-ROM へのアクセスを行うため、実習時の効率は良くない。さらに、たくさんの CD-ROM を作成するのは面倒であるため、古いシステムを使い続ける事になりがちである。

本研究では、ブートサーバによる、クライアント PC の集中管理を行うシステムを構築する。本システムでは、利用目的に応じて、3つの動作モードが利用可能で、一時的にブートサーバの管理から離れて、独立して起動する動作モードも用意した。

### 2. システムの概要

#### 2.1 システムの構成

本システムでは、ネットワーク関連の実習での利用を考慮し、対象とする OS として、FreeBSD6.1R[1]を採用し、PXE (Preboot eXecution Environment) [2]によるネットワークブートを利用する。

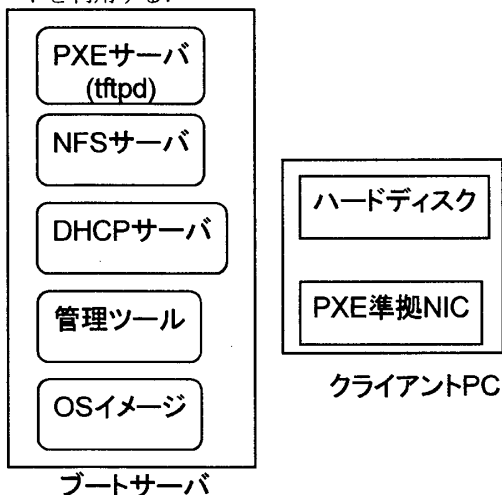


図1 システムの構成

システムの構成を、図1に示す。ブートサーバは、3つのサーバ、PXEサーバ (TFTPサーバ)、DHCPサーバ、NFSサーバとOSのイメージファイル、管理用のツール群を提供する。

クライアントPCは、PXE準拠のNIC (Network Interface Card) と BIOS を持つ。ローカルブートを行いたい場合は、ハードディスクを搭載している。

#### 2.2 動作モード

本システムでは、様々な利用目的に対応できるように、3つの動作モードを提供する。

##### (1) ディスクレス

PXEを利用してネットワークブートし、NFS上のファイルシステムをルートファイルシステムとしてマウントする。PCはブートサーバに管理されている。

新しいPCの接続した場合は、最初にこのモードで起動され、必要があれば、OSイメージファイルのコピーを行い、他の動作モードへ移行する。

##### (2) 管理付きローカルブート

PXEを利用してネットワークブートし、ファイルシステムは、ローカルに接続されたハードディスクを用いる。どのような構成や設定で起動されるかは、ブートサーバ側で管理されている。

##### (3) 管理なしローカルブート

ローカル接続されたハードディスクから起動し、ハードディスク内のファイルだけで動作する。ブートサーバ無しで使用できる事ができ、管理下には無い。ブートサーバの無い環境に持ち出して使用する場合に、この動作モードを利用する。OSの更新やpackageの追加の必要が生じた場合は、上記の2つの動作モードに戻り、ブートサーバの管理下でそれらを行う。

#### 2.3 OSのイメージファイル

本研究では、vnode形式のファイルを使用しており、これをOSのイメージファイルと呼ぶ事にする。用意したファイルは、Base.img, X.imgの2つである。Base.imgは、最小限のシステムをインストールしたもので、X.imgは、X Window System関連のファイルを格納したものである。

この他に、ユーザの指定したpackageを格納したExt.imgを追加する事ができ、ユーザの行った変更を保存するUser.imgを作成可能である。

本文中では、省略するが、各OSイメージファイルには、ファイル名に作成した日付と時刻からなる識別子を含む。この識別子によって、設定や構成の違うOSイメージを複数保持する事ができ、システムのアップデートや使用目的の変更にも容易に対応できる。

†山形大学, Yamagata University

‡マジックワークス, MagickWorX

### 3. PCの起動方法

#### 3.1 pxeboot

本研究で採用した FreeBSD では、PXE に対応したブートローダとして、pxeboot がある。pxeboot は、指定された NFS サーバ上のファイルシステムをルートファイルシステムとしてマウントして起動することができる。さらに、pxeboot は、/boot/loader.rc を読み込んで起動することができる。loader.rc によって指定された mfsroot を読み込み、これをルートファイルシステムとしてマウントして起動することも可能である。

以下では、本システムで提供する3つの動作モードについて、その起動方法を説明する。

#### 3.2 ディスクレス

ディスクレスの場合は、PXE を利用してネットワークブートし、NFS 上のファイルシステムをルートファイルシステムとしてマウントして起動する。具体的な手順は次のようになる。

- (1)クライアント PC は、DHCP (BOOTP) サーバから IP アドレスとロードするファイル (pxeboot) の情報をもらい、PXE サーバから pxeboot をロードする。
- (2)pxeboot は、指定された NFS サーバ上の /boot/ kernel をロードする。
- (3)NFS 上のファイルシステムをルートファイルシステムとしてマウントし起動する。

#### 3.3 管理付きローカルブート

管理付きローカルブートの場合は、PXE を利用してネットワークブートし、ファイルシステムは、ローカルに接続されたハードディスク中の OS イメージファイルを用いる。

- (1)クライアント PC は、PXE サーバから IP アドレスをもらい pxeboot をロードする。
- (2)pxeboot は、指定された NFS サーバ上の /boot/loader.rc を読み込み、指定された kernel と mfsroot をロードする。
- (3)読み込んだ mfsroot をルートファイルシステムとしてマウントし起動する。ここで、mfsroot の作り方によっては、mfsroot のみでの利用も可能である。その場合は、次の(4)の処理はスキップされる。
- (4)起動処理中に、mfsroot 中の /etc/rc が実行されるが、この /etc/rc の先頭で、ハードディスクをマウントし、さらにその中にある Base.img を /u にマウントする。最後に、/u/usr を /usr へ unionfs 形式でマウントする。必要に応じて、他の OS イメージファイルも同様にマウントする。
- (5)以後は、通常の起動処理と同じである。

#### 3.4 管理なしローカルブート

ローカル接続されたハードディスクから起動し、ハードディスク内のファイルだけで動作する。ブートサーバ無しで使用できるが、管理下には無い。具体的な手順は次のようになる。

- (1)クライアント PC をブートサーバのある環境で起動する場合は、最初の起動デバイスがハードディスクであるか、DHCP サーバが、PXE に関する情報を返さないようにしておき、PXE による起動が失敗して、ハードディスクから起動するようにしておく。

(2)ハードディスクから起動する。ローダとして /boot/loader が起動し、/boot/loader.rc を読み込み、指定された kernel と mfsroot をロードする。

(3)読み込んだ mfsroot をルートファイルシステムとしてマウントし起動する。ここで、mfsroot の作り方によっては、mfsroot のみでの利用も可能である。その場合は、次の(4)の処理はスキップされる。

(4)起動処理中に、mfsroot 中の /etc/rc が実行されるが、この /etc/rc の先頭で、ハードディスクをマウントし、さらにその中にある Base.img を /u にマウントする。最後に、/u/usr を /usr へ unionfs 形式でマウントする。必要に応じて、他の OS イメージファイルも同様にマウントする。

(5)以後は、通常の起動処理と同じである。

### 4. まとめ

本研究では、ブートサーバによる、クライアント PC の集中管理を行うシステムを構築した。本システムでは、利用目的に応じて、(1) ディスクレス、(2) 管理付きローカルブート、(3) 管理なしローカルブートの3つの動作モードが利用可能である。OS は、ハードディスクに直にインストールされるのではなく、OS イメージファイルの形で準備される。そのため、システムの更新や、利用目的の変化などに応じて、利用する OS イメージファイルを切り替えれば良く、すばやく、柔軟に対応可能である。

### 5. 今後の課題

現在の管理ツールは、コマンドベースであるが、管理者の負担の軽減や、前提知識の乏しいユーザでも利用可能なように、Web ベースのインターフェースをつける必要がある。また、多様な目的に対応するため、様々な package を組み込んだ OS イメージファイルを準備する必要がある。それらを手動で作っているのは、管理者の負担が大きいため OS イメージファイルの自動生成ツールの充実も必要である。

本研究では、FreeBSD を対象としたが、実習の内容によっては、他の BSD 系システムや Linux などへの対応も必要と思われる。さらに、VMware[3]などの仮想マシンの利用も検討したい。

### 参考文献

- [1]FreeBSD, <http://www.freebsd.org/>
- [2]Preboot Execution Environment (PXE) Specification, <http://www.intel.com/design/archives/wfm/downloads/pxespec.htm>
- [3]VMware, <http://www.vmware.com/>