

Console System Using ThinClient for the J-PARC Accelerator

Susumu Yoshida^{A)}, Makoto Takagi^{A)}, Shigenobu Motohashi^{A)}, Takao Iituka^{A)},
Norihiko Kamikubota^{B)}, Noboru Yamamoto^{B)}, Tadahiko Kato^{B)},
Hidetoshi Nakagawa^{B)}, Jun-ichi Odagiri^{B)}, Takahiro Matumoto^{B)}

^{A)} Kanto Information Service (KIS)

8-21 Bunkyocho Tsuchiura, Ibaraki, 300-0045

^{B)} High Energy Accelerator Research Organization (KEK)

1-1 Oho, Tsukuba, Ibaraki, 305-0801

Abstract

The console terminal can be said to be the one of the essential devices in accelerator control.

As a method to do the increased many console management that simply, I thought that simplified management with the terminal, which was diskless and was network setup possibility.

I report the result that I tested at two sides of management and use. I argue about a problem in the ThinClient use that met with by an examination and the solution.

J-PARC制御へのThinClient端末の導入

1. はじめに

加速器制御においてコンソール端末は必要不可欠な装置の一つといえます。

昨年より運転が開始されているJ-PARCのLinac運転においては一般的なPCにLinuxをインストールしコンソール端末として使用しています。運転用と開発用を合わせると十数台の数になります。

PCをコンソールとして使用する上でネックになるのがOSや環境設定などの管理が台数に比例して増えることにあります。今後、RCS、MRの運転開始に伴いコンソールの増設は必須となっており最終的には、運転用のコンソールのみでも数十台規模の端末が中央制御室等で利用されることが予想され、端末増加に伴う管理の煩雑化が想定されます。

端末増加に伴う管理の簡素化を実現する方法の一つとして、ディスクレスでネットワーク立ち上げ可能な端末を用い、管理を一元化する方法が考えられます。

そこでコンソール端末の選定機種の一つとしてThinClient端末を導入しテストを行いました。

管理及び運用の2つの側面でテストを行った結果を報告するとともに試験運転での利用で遭遇したThinClient運用上の問題とその解決方法についても報告します。

2. 機種選定

今回の機種選定ではThinClient端末であることを前提条件としました。

ThinClient端末とは端末自身にはハードディスクを持たずに画面表示とキーボードやマウスなどの入力操作のみに特化し、アプリケーションやファイル

管理をすべてサーバ側で実施するシステムの端末です。

ThinClient端末のメリットとしてはハードディスクやファンを無くすことで故障率を減らせることと、OSやアプリケーションなどの資源をサーバに置き一括管理が行える点にあります。また、端末側にデータを記憶する領域を持たないことによりデータ漏洩などの危険が無くなることも注目すべき点です。

具体的な機種選定の条件としては拡張性があることが絶対条件とされました。具体的にはPCIカードの増設が可能であり、2~4Displayのビデオカードが動作することが必須条件とされました。

これは運転時においては複数の操作画面を表示する関係上、1つの端末で最低限2つ以上のディスプレイを接続できることが運用上望ましく、またスペースの有効活用という点においても必要とされる条件でした。

機種選定当時この条件を満たす機種はH社のThinClientのみだったこともありおのずと機種は限定されることとなりました。

H社 Thin Client	
Processor Type	AMD Geode NX-1500
Processor Speed	1000MHz
Amount of flash memory	512MB
Memory size	512MB : 1G

表1 ThinClientスペック

3. ThinClientの実現方法の選択

3.1 実装方式

ThinClientの実装方式としては大きく分けて2つの方式が用いられます。

1つはサーバーベース型、ブレードPC型に見られる画面転送方式、もう一方はネットワークブート方式と呼ばれています。

画面転送方式の特徴はアプリケーションの実行動作をサーバ側で行い画面のイメージのみ転送を行うことにあります。端末は遠隔操作のみに徹することになります。

もう一方のネットワークブート方式は端末上にネットワーク経由でOSをダウンロードしアプリケーションの動作は端末上で行われることとなります。

今回のThinClientの実装方式としてはLinux系OS (Scientific Linux 4.4) の使用を前提としていることもありLinux系OS実装方式としてメジャーであるネットワークブート方式を採用することにしました。

この方式の特徴としてはサーバ側にあるOSイメージをネットワーク経由で参照しブートするところにあります。

また、Windows ThinClientに採用されているサーバーベース方式のようにサーバ側でアプリケーションが動作するのではなく、端末側でアプリケーションが動作するなどの特徴があります。

ネットワークブート方式

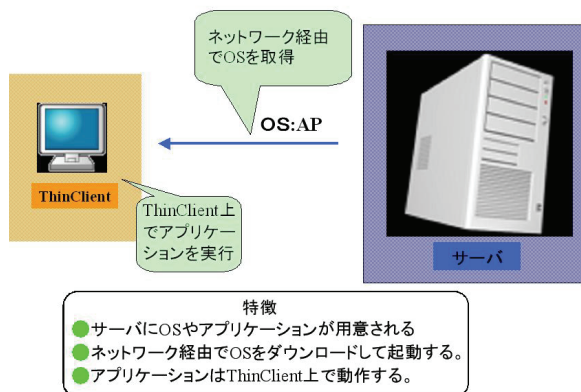


図1 ネットワークブート方式

この方式の利点としてPXE (Preboot eXecution Environment) 規格に対応したネットワークインターフェイスを持つ端末であればサーバ側でDHCP, TFTP等の一般的なサービスを提供するだけで運用がはじめられるため初期コストがかからないというメリットが上げられます。

3.2 OSイメージ

ネットワークブート方式のOSイメージにはNFS-rootとRAMdisk-rootがあります。

NFS-rootでは、サーバ側に用意されたrootファイル領域 (/etc, /usr, /binなど) をNFSマウントする方

式、一方RAMdisk-rootはrootファイル領域のイメージファイルを端末側のメモリ上に展開する方式になります。

今回はNFS-root方式を使用して実装を行いました。

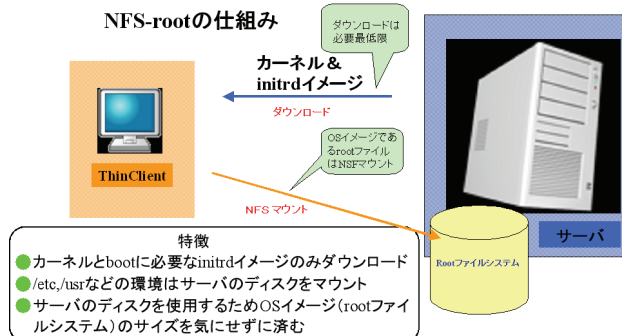


図2 NFS-rootの仕組み

選択の理由としては端末の使用用途において、GUIを動作させるためのX環境などを実装したり、後のアプリケーション追加などを考慮し、イメージサイズを制限されるRAMdisk-root方式よりサーバのディスクを使用できるNFS-root方式の方が拡張性に優れている点を評価しました。

4. boot環境の実装

4.1 サーバ側サービス

サーバ側の準備としてはDHCP, TFTP, NFSの3つのサービスを立ち上げる必要があります。これらのサービスはLinuxOSのサーバであればインストール時に選択可能な一般的なサービスとなっています。

既存サーバにインストールされていない状況であればrpmやyumを使用してインストールが必要になります。

4.2 OSイメージの作成

いくつかあるrootファイル領域作成方法のうち今回はLiveCDを利用した方法を用いました。LiveCDはCD上から起動できるOSの総称でLinux系OSだとKNOPPIXなどが有名です。

今回はJ-PARC用計算機のOSとして採用されているScientific LinuxのLiveCDを元にイメージの作成を行いました。

LiveCDからのイメージ作成^[1]については特別な作業は必要なくKernel及びinitial RAM disk, その他のイメージをサーバ側の所定のディレクトリにコピーすることで完了します。ただし、標準配布のLiveCDに組み込まれていないサービスやアプリケーションの導入が必要な場合にのみオリジナルLiveCDの作成^[2]が必要となります。

4.3 ThinClientの起動

サーバ側のサービスに対してThinClient端末のbootに関する設定^[1]を行ったあと端末をネットワークに接続し電源を入れることでネットワークブート

が始まり端末が起動します。

起動時にはrootファイルシステム以外のユーザ作業領域のNFSマウント、LDAPサーバによる統一アカウント管理により全端末共通の作業環境を実現しました。

これにより起動後はPC端末同様の運用が可能となりました。

5. 運用試験における問題点及び改善

グループ内での使用においては得に問題もなく通常のPCコンソールと同様に使用できることが確認されました。

次のステップとしてよりハードで実践的な運用試験を行うべく、運転中のLinacの現場で本格的な運用試験を行い実働に耐えられうかどうかの現地試験を行いました。

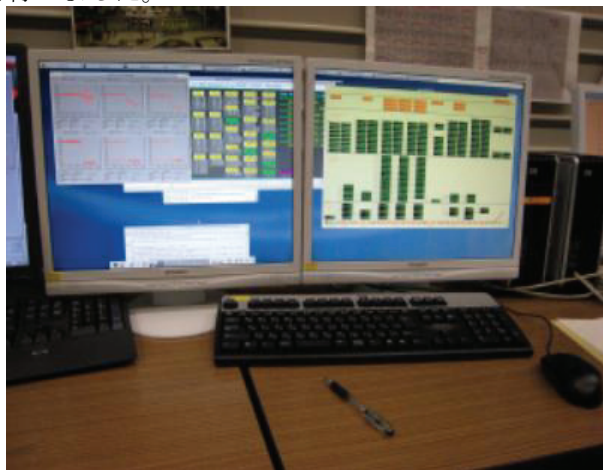


図3 Linacコミッション風景

この現地試験においてはいくつかの問題点が発生し一番深刻なものとしては“端末の停止”が上げられます。

頻度としては1日一回の割合で発生するという深刻なものでした。また、この現象は同じ利用方法のPC端末では発生せずThinClient端末独自の現象でもありました。

地道な調査の結果分かった原因はアプリケーションのメモリリークによるものと判明しました。LinuxOSではある一定のメモリ消費率を超えメモリが足りなくなるとoomkillerという不要なプロセスを停止するサービスが存在します。この機能によりThinClient端末上で動作していたXのプロセスが強制停止し端末としての機能が損なわれる状況が発生していました。

PC端末でこの現象が発生しなかったことについてはハードディスクに実メモリーの2倍に相当するSWAP領域を保持していることでメモリ消費をサポートしていることが有効に働いていた模様です。

現在はメモリリークを起こしていたアプリケーションのバージョンUPを行うことで端末が停止する状況は発生しなくなりましたが、今後の運用においても同様のケースが発生することが予想されます。

これについては他のサーバ計算機上でアプリケーションを実行し端末側では画面表示のみを行うX端末のような運用方法をとることで回避することが可能だと考えています。

このように運用試験においてはメモリリーク問題によってハードディスクを持たずSWAP領域を取ることができないThinClient端末の欠点が浮き彫りとなる結果となりましたが、それと同時に実際の運転環境においても運用次第で十分実用可能であることが検証されました。

6. 結論

初期導入時にサーバ側でのサービス設定やOSイメージの構築など若干敷居の高い作業が必要になりますが、追加購入したThinClient端末の初期セットアップについてはDHCPサーバに登録を行うなどの少ない処理で迅速にユーザに提供できるようになりました。

OSのバージョンUPや設定変更についても一元管理が可能となり管理面での利便性はかなりの向上を見ることができました。

運用試験の結果としては実メモリの枯渇による問題が発生しました。これは前述のメモリリークのみでなく波形データを大量に表示するなどメモリ消費量が多くなるとアプリケーションが停止する状況も確認されています。

このようにThinClient端末を使う上ではメモリ使用量への若干の配慮が必要となりますが、通常使用においてはPCコンソールと分け隔てなく使用することが可能であり、他に目立った問題点は今のところ発見されていません

実用上若干の問題点があるものの運用上カバーできる範囲であり、多数のコンソール導入時の選択肢の一つとしては十分検討に値する結果となりました。

参考文献

- [1] “Scientific Linux Live CD and DVD-Diskless Client-SL/RHEL Diskless Client”,
URL: <http://linux.web.psi.ch/livecd/diskless.html>
- [2] “Scientific Linux Live CD and DVD-Diskless Client-SL/RHEL Building your own LiveCD”,
URL: <http://linux.web.psi.ch/livecd/build.html>