



筑波大学  
*University of Tsukuba*



# 大学の教育研究用端末上での ベアメタルハイパバイザの運用

大山 恵弘

筑波大学 学術情報メディアセンター

# 全学計算機システム

## Campus-wide Computer System

- 筑波大学全学の教育研究等のために提供される計算機システム
- 2017年3月に更新，以後5年間運用予定
- 複数キャンパス内の16拠点（サテライト）に1000台以上の端末を配置
  - 一部のサテライトでは24時間利用可能
- どの端末でも，ほぼ同一の環境が利用可能
  - vThrii ベース（BitVisor ベース）のネットワークブートシステムを利用

# ネットワークブートのために 用いている基盤ソフトウェア



## • vThrii

- ネットワークブート機能，マルチブート機能を提供
  - Windows, macOS, Linux をサポート
  - ディスクイメージをローカルディスク or サーバから取得してブート
- BitVisor を拡張した商用ソフトウェア
- ブート後も OS の下で（こっそり）稼働し続ける
  - ユーザが存在を意識することはない
- 東京大学の教育用計算機システムで運用実績あり
  - Windows 10 と Mac OS X El Capitan のデュアルブート
  - 端末は1000台以上

# 端末室



# 端末



製品名	Dell Optiplex 7040 SFF
CPU	Intel Core i5-6600 (4 cores, 4 threads, 3.3GHz, 6 MB cache)
メモリ	8GB (2 × 4 GB), 2133 MHz, DDR4
ストレージ	SSD Micron 1100 (2.5 inch, 256 GB, SATA)
ネットワーク	Gigabit Ethernet
ディスプレイ	Dell P2417H (23.8 inch) Dell P2217H (21.5 inch)

1 学情サテライト  
①A203



2 1Cサテライト  
⑤1C206



6 3Dサテライト  
⑬3D207



10 春日サテライト  
⑱7C102, ⑲7C103, ㉑7C202



13 体芸図書館サテライト  
㉗視聴覚室



1 学情サテライト  
②A207



3 1Dサテライト  
⑥1D301



7 3Kサテライト  
⑭3K203



11 東京サテライト  
㉒文京校舎 4F 454



14 医学図書館サテライト  
㉘端末室



1 学情サテライト  
③B205



4 2Aサテライト  
⑦2A303, ⑧2A304



8 文修サテライト  
⑮8B201



12 中央図書館サテライト  
㉓2F閲覧室～㉔5F閲覧室



15 図書館情報学図書館サテライト  
㉙1F春日ラーニング・commons



1 学情サテライト  
④B206



5 2Dサテライト  
⑩2D201, ⑪2D202～⑫2D204



9 医学サテライト  
⑯4A402, ⑰4B212



12 中央図書館サテライト  
㉕2Fコミュニケーションルーム



16 大塚図書館サテライト  
㉚文京校舎 地下1F 閲覧室



# OSのデュアルブート



Windows 10 Enterprise LTSP



Ubuntu 16.04 LTS

# なぜ vThrii を導入することになったか？

所属組織による見解ではなく、個人の感想です

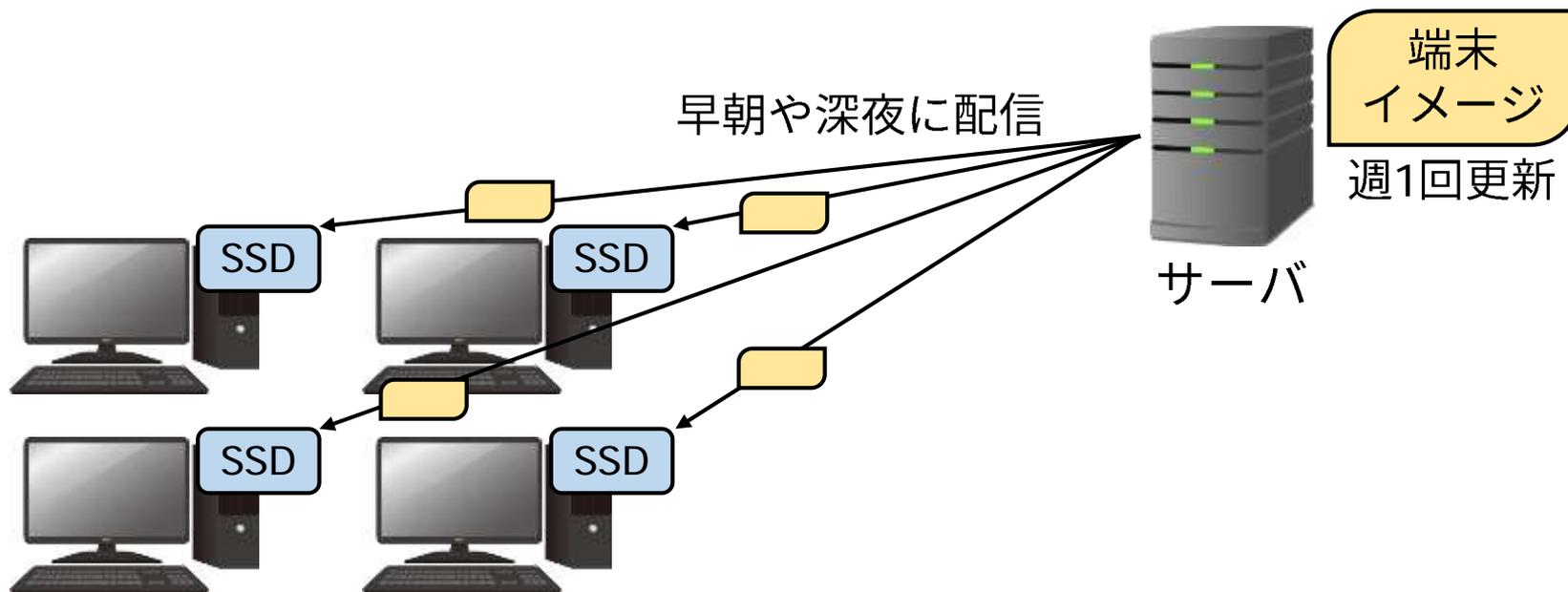
- 導入する決定的な理由はなかった
  - Windows と Linux のデュアルブートができて、
  - 性能もそこそこで、
  - 価格も適度で、
  - 東大での運用実績もある
- しかし、導入しない強い理由もなかった
  - 致命的問題はない
  - ネガティブな噂をあまり聞かない
    - 運用実績が少ないだけかもしれないが...
  - BitVisor に対する拒否反応もない
    - 「端末でベアメタル」「ゲストOSは1個だけ」には慣れた

# 起動時間（2017年12月4日時点）

	Windows	Linux
OS 選択画面の表示までの時間	18秒	
サインイン画面の表示までの累積時間	33秒	41秒
アプリケーションが起動できるまでの累積時間	90秒	60秒

※ 初回サインインではもっと長い時間がかかる

# 端末イメージの配信



端末

- 通常，端末はほぼ SSD だけで起動できる
  - 端末の電源オフ後も SSD がイメージを保持する
  - イメージが端末にキャッシュされていれば，起動時にイメージの取得処理が実行されることはまずない
- ユーザのプロファイル情報は，起動時にサーバから読み込まれる

# 端末のOS起動まわりの障害や不調

所属組織による見解ではなく、個人の感想です

- 時々発生している
  - 端末は1000台以上あるので不可避という気もする
- 具体的症状
  - 黒画面のまま起動せず
  - Windows の画面が出たまま停止
  - 起動が異常に遅い（特にイメージ配信直後）
  - OS が突然シャットダウン処理を始める（原因は既知）
- 業者が都度対応
  - ネットワンシステムズ
  - キヤノンITソリューションズ
  - イーゲル

# 障害対応

- 「黒画面のまま起動せず」「Windows の画面で停止」



vPro による再起動



現地に出向いて電源ボタンを押して再起動，とか，  
現地に出向いてケーブル類のチェック&差し直し

- これで大体直る
  - 直らないものの多くは SSD 異常などのハードウェア故障

症状をイーゲル社やデル社にエスカレーション

# 障害について思うこと

- ハイパバイザがあると，原因の候補が1つ増える
  - アプリが原因？
  - Windows が原因？
  - ハードウェアが原因？
  - ハイパバイザが原因？
- 特に起動とシャットダウン周りの障害では，私はずいぶん vThrii を疑ってしまう
  - vThrii のせいではないことも多いと思うが
  - OS やアプリが遅すぎるときも，つい vThrii を疑ってしまう
    - アンチウィルスなどの常駐系ソフトウェアと共通の悲しさ
- 逆に，99%の一般ユーザはまず Windows を疑う？
  - vThrii の知名度の低さと存在感のなさが幸いしている可能性

# ここからはちょっと趣味の話



# 導入に関する考慮事項1：性能

- ハイバイザの導入による性能低下の懸念
  - OS やアプリケーションが遅くならないか？
  - vThrii のオーバヘッドは小さいことが良く知られている
  - しかし，オーバヘッドを増大させる要因は多数考えられる
    - OS の種類やバージョンや設定
    - メモリ量
    - CPU の性能
    - ...



- オーバヘッドを正確に把握する実験が必要

# 導入に関する考慮事項2： 互換性

- 実機上で生じなかった不具合が生じないか？
  - ベアメタルハイパバイザは、ほぼ実機と同じ仮想ハードウェアを提供
  - しかし、実機と異なる性能・動作がある可能性
  - 差異に敏感な OS やアプリケーションの存在の可能性



- 生じる差異を正確に把握するには実験が必要

# 実験

- OS の基本処理の性能の測定
- 実機上での動作との相異の調査

# OSの基本処理の性能

- Ubuntu 上で UnixBench を実行
  - <https://github.com/kdlucas/byte-unixbench>
- 端末室の端末とほぼ同じスペックの実験用端末（比較端末）を使用
  - とともに Dell Optiplex 7040SFF
  - CPU と SSD の性能がわずかに違うのみ
    - CPU: クロックが 0.1GHz だけ違う
    - SSD: 速度がわずかに違う
  - 比較端末ではハイパバイザなしで Ubuntu を実行

# わずかに違う仕様

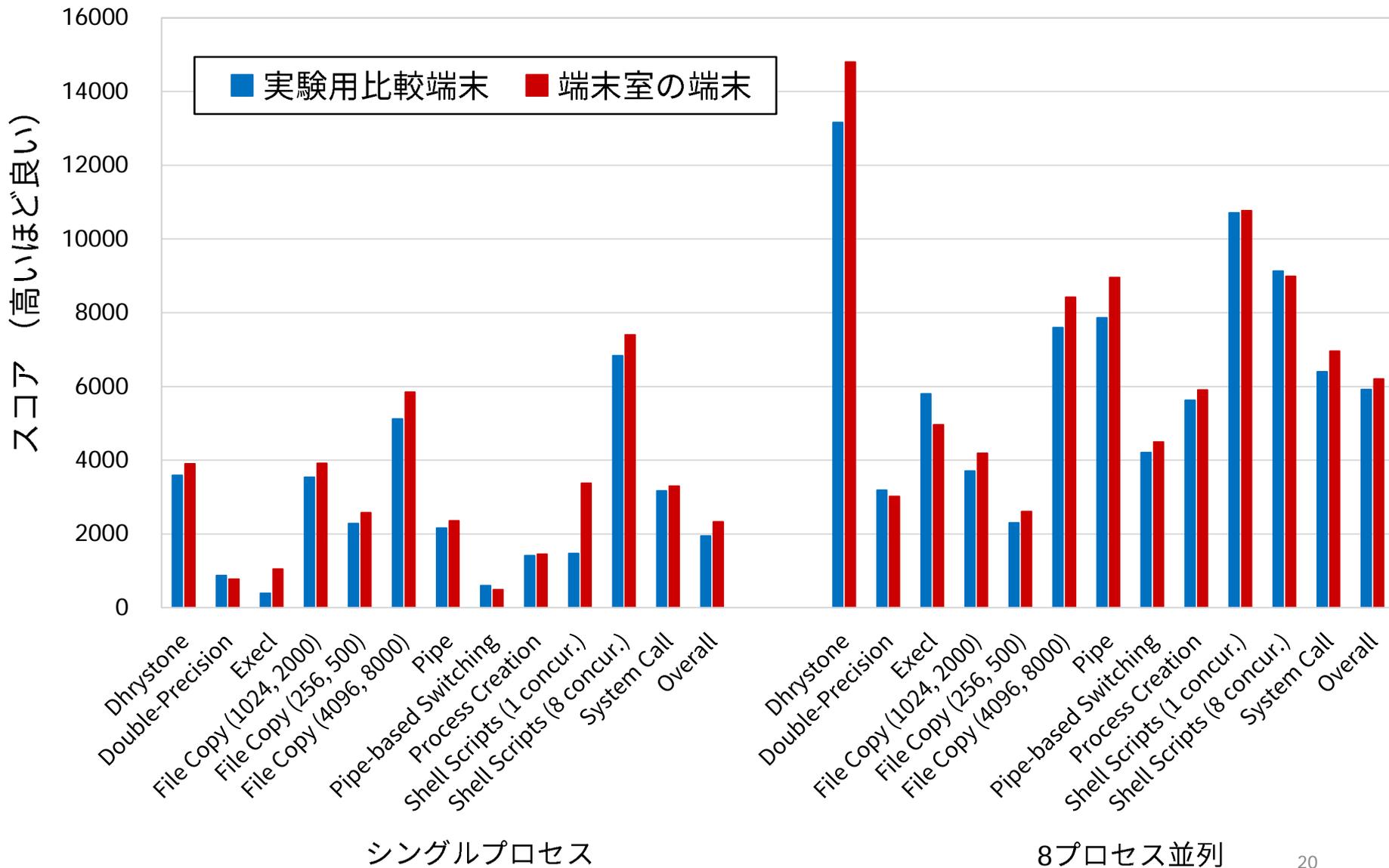
オリジナル

製品名	Dell Optiplex 7040 SFF
CPU	Intel Core i5-6600 (4 cores, 4 threads, 3.3 GHz, 6 MB cache)
メモリ	8GB (2 × 4 GB), 2133 MHz, DDR4
ストレージ	SSD Micron 1100 (2.5 inch, 256 GB, SATA)

実験用

製品名	Dell Optiplex 7040 SFF
CPU	Intel Core i5-6500 (4 cores, 4 threads, 3.2 GHz, 6 MB cache)
メモリ	8GB (2 × 4 GB), 2133 MHz, DDR4
ストレージ	SSD Intel Pro 2500 (M.2, 180GB, SATA)

# 測定結果



# 仮想マシンと実機での動作の相異の調査

- 仮想化機構の存在を検出するプログラムを使用
  - Paranoid Fish (Pafish) と AI-Khaser
  - 実機と仮想マシンで動作が変わりうる多数の処理を実行して，結果を出力
  - 処理の例：
    - 一部の CPU 命令の実行時間や挙動の測定
    - CPUID 命令が返す値の取得
    - プロセススケジューリング挙動の測定
    - 特徴的なデバイスの名称や属性の調査
    - 特徴的なレジストリキーやファイルの存在の調査
    - ディスクサイズの測定
    - MAC アドレスの取得

# Pafish の出力例

```
Pafish (Paranoid fish) *
Some anti(debugger/VM/sandbox) tricks used by malware for the general
public.
[*] Windows version: 10.0 build 14393
[*] CPU: GenuineIntel
Hypervisor: <
CPU brand: Intel(R) Core(TM) i5-6600 CPU @ 3.30GHz
[-] Debuggers detection
[*] Using IsDebuggerPresent() ... OK
[-] CPU information based detections
[*] Checking the difference between CPU timestamp counters (rdtsc) ...
OK
[*] Checking the difference between CPU timestamp counters (rdtsc)
forcing VM exit ... traced!
[*] Checking hypervisor bit in cpuid feature bits ... OK
[*] Checking cpuid hypervisor vendor for known VM vendors ... OK
[-] Generic sandbox detection
[*] Using mouse activity ... OK
[*] Checking username ... OK
[*] Checking file path ... OK
[*] Checking common sample names in drives root ... OK
[*] Checking if disk size <= 60GB via DeviceIoControl() ... OK
[*] Checking if disk size <= 60GB via GetDiskFreeSpaceExA() ... OK
```

# 調査結果

- vThrii を挟んでも、各処理の挙動は実機とほぼ同じ
  - 挙動の変化が検出された処理はわずか
    - ハイパバイザに制御を移動させる命令（CPUID 命令）のサイクル数
    - プロセススケジューリングの挙動が影響を与える関数（NtYieldExecution 関数）の返回值
- vThrii はハイパバイザであるにもかかわらず実機に極めて近い環境を OS に提供している

# おわりに

- 筑波大学の vThrii ベースのネットワークブートシステムは約8か月運用されたが、大体うまく動作している
- vThrii が原因と特定された障害や vThrii のオーバヘッドに対する不満は報告されていない
  - 「特定」や「報告」がされていないだけかもしれないが
- 起動時間に関しては考えることが多い
  - ローカル SSD へのイメージのキャッシュはほぼ必須という印象
    - キャッシュされていないと起動がすごく遅い
    - よって、実は現在は、それほどネットワークブートをしていない
  - 起動時間をさらに短縮するには詳細な性能分析が必要
- 何もしていないのに SSD のビジー率が99%になることがある...
- 最短でも2022年3月まで vThrii のサポートが必要です