

## 4.2

# サーバサイジング

MetaFrameの導入を検討するとき、「1台のサーバを何クライアントが同時に利用できるか」という質問がよくあります。これがサーバサイジングの課題です。

### 4.2.1 コンパック社のベンチマークテスト結果

参考までに、コンパック社でのベンチマークテストの結果を表2に示します。

このベンチマークテストは、下記の環境で実行されました。

サーバOSには、Windows2000Server、MetaFrame XPをインストールしてあります。搭載したソフトウェアは、英語版です。ヘビーユーザがアクセスしたと想定しています。ヘビーユーザは1分間に300文字（60単語）の入力をする想定します。

同時接続のユーザ数を少しずつ増やし、プロセッサの利用率が80%になったとき、または、アプリケーションの応答時間が大幅に低下する時点のユーザ数が表2の数字です。1人あたり8MBのメモリを使うアプリケーションを利用しました。ただし、使用するアプリケーションやユーザの利用条件によって結果は変化しますので、この数値をそのまま採用するには注意してください。

### 4.2.2 MetaFrameサーバの台数決定要因

MetaFrameサーバを同時に使えるクライアントユーザ数は、次の要因で決定されます。

表2 MetaFrameサーバのベンチマークテストの結果

| サーバモデル         | プロセッサ           |       |          |       | メモリ | 同時接続のユーザ数 |
|----------------|-----------------|-------|----------|-------|-----|-----------|
|                | CPUの種類          | CPU台数 | 周波数      | キャッシュ |     |           |
| ProLiant DL360 | PentiumIII      | 2CPU  | 933MHz   | 256KB | 1GB | 53        |
|                | PentiumIII      | 2CPU  | 1.0GHz   | 256KB | 2GB | 60        |
|                | PentiumIII      | 2CPU  | 1.266GHz | 512KB | 2GB | 91        |
| ProLiant DL380 | PentiumIII      | 1CPU  | 1.000GHz | 256KB | 4GB | 32        |
|                | PentiumIII      | 2CPU  | 1.000GHz | 256KB | 2GB | 60        |
|                | PentiumIII      | 1CPU  | 1.133GHz | 512KB | 4GB | 42        |
|                | PentiumIII      | 2CPU  | 1.133GHz | 512KB | 4GB | 90        |
|                | PentiumIII      | 2CPU  | 1.266GHz | 512KB | 2GB | 91        |
| ProLiant DL580 | PentiumIII XEON | 2CPU  | 900MHz   | 2MB   | 4GB | 65        |
|                | PentiumIII XEON | 4CPU  | 900MHz   | 2MB   | 4GB | 120       |
| ProLiant DL760 | PentiumIII XEON | 4CPU  | 900MHz   | 2MB   | 4GB | 125       |
|                | PentiumIII XEON | 8CPU  | 900MHz   | 2MB   | 4GB | 170       |

クライアントユーザの入力速度  
利用するアプリケーションのCPU負荷やメモリ使用容量  
サーバハードウェアの仕様

これら3つの各要因の変化値は多様にあります。したがって1つの方程式のようなもので簡単に、「同時に使えるクライアントユーザ数」を算出することはできません。では、画面をみているだけのクライアントユーザもいるし、ログオンはしているが離席しているクライアントユーザも勘定しなければなりません。そこで、使う予定のアプリケーションを実際に1台のサーバにインストールしてみて、多くのオペレータに同時に使ってもらい、事前にテストするのが現実に近い算定方法です。その結果をもとにサーバ1台で同時に使えるクライアントユーザ数を算出します。

この方法は、オペレータとクライアントマシンを大量に準備できなければテストが行えません。たとえオペレータを集められたとしても、そのオペレータ全員がアプリケーションの操作に未熟であった場合には、操作が滞りがちになり、サーバにあたえる負荷は小さくなるかもしれませんが。また、「複数のセッション」でのオペレーションを検証する場合に、人が操作するためにはキーボード・マウス・ディスプレイが必ず必要になり、1台のクライアントマシンでは1セッションしかテストできません。

この問題に対する1つの答えとして、ベンチマークテストツールによって自動的にアプリケーションの操作を行うという手法があります。これにより、多くのオペレータを用意できなくても、かわりにプログラムがアプリケーションの操作を行うことが可能になります。また、1つのクライアントで複数のセッションを起動しても、自動実行が各セッションに対して行われるようにすることで、実際のクライアント台数以上のユーザ数でテストを行うことが可能です。実際に大規模なシステムの事前検証を実現するにはこの手法以外に選択肢がないといえるでしょう。

このベンチマークテストの方法については、4.2.5項「ベンチマークテストの実施」(p.226)以降で説明します。

### 4.2.3 搭載CPU本数への考察

MetaFrameのサーバとして2CPU機4台の構成と4CPU機2台の構成のどちらがコスト・性能が良いかという質問があります。どちらも、CPUの総本数は同じ8本です。

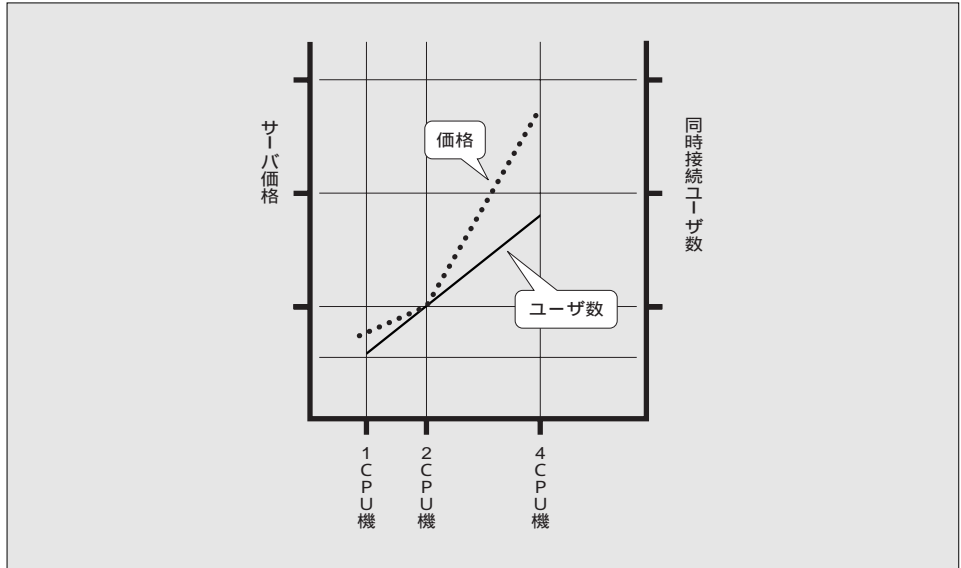
4.2.1項「コンパック社のベンチマークテスト結果」を参考にすると、4CPU機を同時利用できるユーザ数は、2CPU機の2倍よりやや少なくなります。一方、メーカーにより、やや異なりますが、4CPU機の価格は2CPU機の2.5倍以上になります。したがって、この設問に対する答えは、「2CPU機4台のほうがコスト性能比がよい」となります。

一方、ラックマウント型の2CPU機の価格は、1CPU機の1.2倍程度ですから、1CPU機4台より2CPU機2台のほうがコスト性能比がよいといえます。

MetaFrame XPのサーバライセンスは、無制限のサーバ台数へのインストールが許可されています。サーバ台数が2台でも4台でもMetaFrame XPの価格は変わりません。マイクロソフトのWindows2000のサーバライセンスだけが価格面で影響します。マイクロソフトのCALやTSCALの数はサーバが2台でも4台でも変わらないので、それほどインパクトはありません。

注) MetaFrame1.8では、サーバ台数毎にサーバライセンスを購入する必要があります。MetaFrame1.8ではサーバ

図12 2CPU機と4CPU機の比較



台数を少なくするとMetaFrameの購入費を抑えることができました。

データベースサーバでは、CPU本数を多く、キャッシュメモリも大きい仕様のサーバが向いています。CPU本数が多く、キャッシュメモリも多いと、レスポンスも速くなり、多くのトランザクションを処理できます。逆に、サーバ台数を増やしてもレスポンスが速くなることはありません。また、データベースサーバが複数だと、データベースの整合性を維持するのが困難です。ただし、耐障害性を考えて、データベースサーバでも、1台でなく2台の体制が望ましいでしょう。

#### 4.2.4 サーバOSの考察

Windows2000系サーバOSには、下記の3種類があります。

- Windows2000 Server
- Windows2000 Advanced Server
- Windows2000 Data Center Server

数百台のサーバを必要とするような、スケーラビリティを要求されるシステムでも、MetaFrameのサーバのOSは、Windows2000 Serverで充分です。なぜなら、CPU本数が2本程度のサーバハードウェアを使うからです。ただし、Windows2000の次期サーバOSである.NET Serverの時代になると、また別途検討が必要になるかもしれません。

Windows2000 Advanced Server以上には、NLB(Network Load Balancing)の機能がありますが、MetaFrameのサーバOSとして使う場合は、NLBの機能は使いません。MetaFrameのロードバランシング機能は500台の規模で実証済みですが、NLBは32台が最高値です。

MetaFrameは、アプリケーションサーバとしてのロードバランシング機能の要請に応え、多くのパラメータで負荷値を算定します(2.2.B1項「アプリケーション・サーバとしてのロードバランシング機能」(p.39)を参照ください)。

## 4.2.5 ベンチマークテストの実施

ベンチマークテストツールは、多くのメーカーから色々な種類の製品が提供されています。シトリックスからはCitrix Server Test Kitが、マイクロソフトからはRoboClient・Server/SimulatedClient<sup>注1)</sup>が提供されています。

これらのツールでは主に2つの機能が提供されています。複数のクライアント上でICA/ターミナルサービス・クライアントをリモート起動しログインさせるためのツールと、起動されたアプリケーションの操作を行う機能です<sup>注2)</sup>。これらの機能により、「複数のセッションから」、「同時にアプリケーションを操作する」ことが可能になります。それでも人によって、入力速度は色々あるので、完全に現実に近い数値を算出できません。

注1) Windows2000 Server 日本語版において、筆者はマイクロソフトから提供されているこれらのツールの動作確認ができていません。

注2) Simulated Clientは作成されたスクリプトに応じてRDPセッションのアプリケーションを操作することが可能ですが、Citrix Server Test KitではWinBatchによって作成された自動操作を行うための.EXEファイルが提供されているのみになります。

### 4.2.5.1 テスト環境

実際にテストを行うには図13のような環境が必要になります。

テストにおいては、MetaFrameXP/ターミナルサーバ以外のサーバでボトルネックが発生しないように注意してください。とくに注意する必要があるのはデータベースサーバを使用した

図13 ベンチマークテストの環境

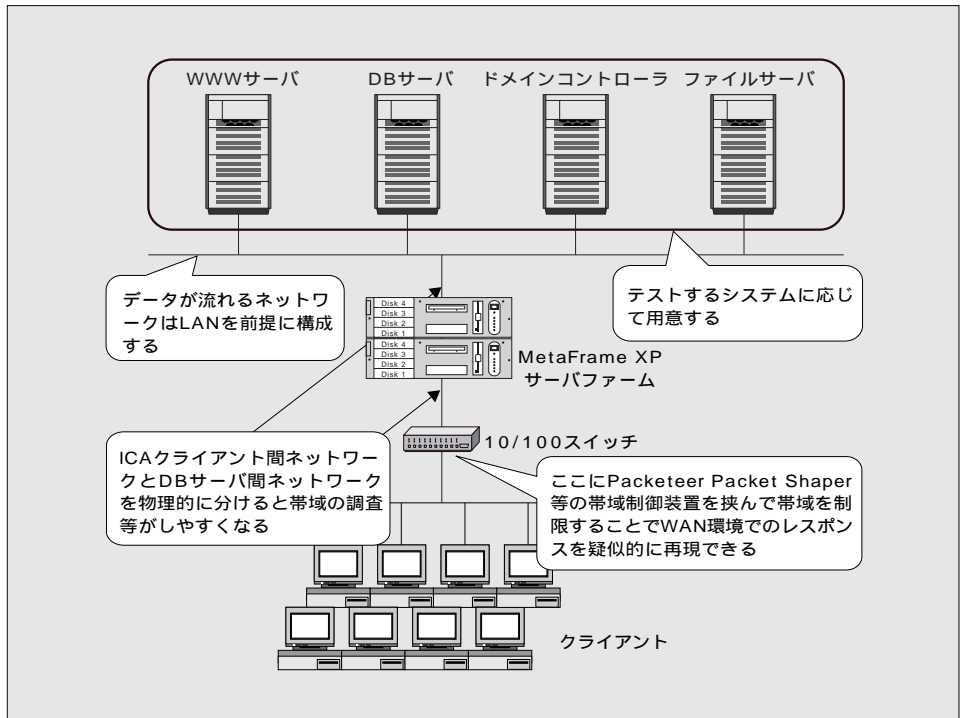


表3 ミディアムユーザとヘビーユーザの定義

| 種 類                        | 内 容  |
|----------------------------|--|
| ミディアムユーザ<br>(熟練した従業員)      | <p>熟練した従業員は、意思決定処理中に、情報を収集、付加価値を盛り込み、情報のやり取りを行います。熟練した従業員は、何を処理し、どのように業務を達成するか、自分自身で決定します。</p> <p>業務内容は柔軟性に富み、計画外で進行する傾向があります。この例として、マーケティング、プロジェクト管理、営業、デスクトップパブリッシング、意思決定、データマイニング、経済分析、経営管理、設計、承認などが挙げられます。</p> <p>通常、ミディアムユーザの平均的な入力速度は、1分間に35ワード（語）です。コンパックは、ミディアムユーザごとに12MBのメモリを割り当てました。システムダウンによる損害は、多大です。</p>        |
| ヘビーユーザ<br>(構造化された作業を行う従業員) | <p>構造化された作業を行う従業員は、一般的に業務の流れや処理過程に沿って作業を進め、同じ処理を繰り返し行うことが多くなります。</p> <p>業務内容は、計画から外れることは稀で、通常計画どおりに進行します。この例として、クレーム処理、支払い業務、請求業務、カスタマーサービス、ハイエンドな製造、ハイエンドな保守、修理などが挙げられます。通常、ヘビーユーザの平均的な入力速度は、1分間に60ワードです。</p> <p>コンパックは、ヘビーユーザごとに9MBのメモリを割り当てました。</p> <p>システムダウンによる損害は、状況により変わります。多くの場合、この種のユーザは、コンピュータの可用性をそれほど重視しません。</p> |

クライアント/サーバ・システムの場合です。データベースのサーバにはある程度余裕をもった機器を準備すべきです。少なくとも試験時に予想される最大アクセス人数に対して十分な性能を発揮できるサーバでないと、データベース・サーバが先にボトルネックになってしまい、ターミナル・サーバのサイジングデータを取得することができなくなってしまいます。

#### 4.2.5.2 テストシナリオの計画

ユーザの操作をベンチマークツールで行わせるために、どのような操作を行うかを決めなければなりません。ただし、これには標準的なユーザ像を設定する必要があります。

標準的なユーザが、ログオンしてからアプリケーションに入るまで、あるいはキー操作などがどれくらいの速度で行われるかを定義します。表3はユーザ層をミディアムとヘビーに分類して定義した例です（これはGartner Groupによる定義です）。

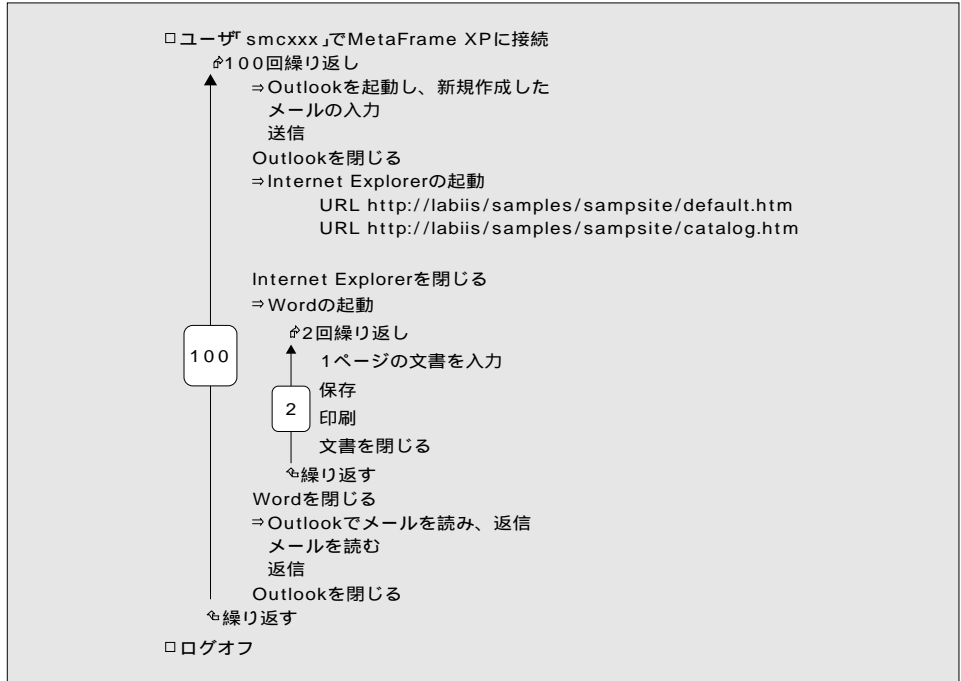
ここで定義した内容にしたがい、実際にアプリケーションを操作する流れを決定します。図14にヘビーユーザの操作の流れ図を示します。

#### 4.2.5.3 テストスクリプトの作成

次に実際にテストシナリオに沿ってユーザの具体的なオペレーションを決定します。オペレーションが決まったら、プログラムで自動実行できるようにスクリプトを作成します。スクリプトの作成は自動実行ツールによりさまざまです。

どのようなツールを使うにせよ、この際に重要なのは人間は機械とは違ってある程度操作に時間がかかるということです。このことを忘れてしまうと、ベンチマークテストの目的を履き違えたデータを取得することになりかねません。Officeアプリケーションのマクロを人間の操作を超える速度で動作させて、人のオペレーションの代用とするのは間違ったアプローチとい

図14 ヘビーユーザの操作



えます。MetaFrameのベンチマークテストを行っているつもりが、いつの間にかたんなるプロセスのベンチマークを行っていたなどということのないよう注意してください。

スクリプト化にはさまざまなツールを使うことが可能です。前述したシトリックスとマイクロソフトが提供しているツール以外にもサードパーティ製品のテストツールやバッチ化ツールを行うことが可能です。これらのツールのライセンスに注意してください。開発ライセンスを購入すれば実行形式での実行時にはライセンスが必要ないものや、セッション数だけライセンスを必要とするものまでさまざまです。非常に高価なものもあれば無料で使えるものもあります。

筆者はRational社Visual Testをよく利用しています。実際のスクリプトはシナリオレコーダによりユーザの操作を自動記録し、それをヒナ型にしてテストスクリプトを開発します。その際にはログをファイルに残すようにすることで、処理の実行にかかった時間を記録したりなども、あわせて行うようにします。

下記にVisual Testによるスクリプトの例を示します。

```

'$INCLUDE'RECORDER.INC'
SetDefaultWaitTimeout(Timeout)
Scenario"Wordの起動から終了まで"

'
'Microsoft Wordの起動
'
'スタートをクリック
WTaskbarStartClk(VK_LBUTTON)

```

```

'プログラムメニューのMicrosoft Wordを起動
WToolBarButtonClk("@2","プログラム(&P)")
WToolBarButtonClk("@1","MicrosoftWord")
'
'Windowが出てくるのを待つ
'
CurrentWindow=WFndWndC("文書1-Microsoft Word","OpusApp",
FINDWINDOWFLAGS,Timeout)
'起動後1秒待ってから操作開始
Sleep(1.0)
'
'テキストの入力
'
Play"COMPAQProLiantServer:MetaFrame%
({KANJI})jissentekisuto{ENTER}"
Play"{ENTER}%({KANJI})"
:

```

スクリプトができれば、実際に動かしてみても人間に無理な挙動や、途中で止まってしまうことのないように調整し完成させます。

実際のスクリプト作成作業は、ここに書いたこと以上に複雑で難解なものです。Windowsのメッセージ処理などに関して知らなければ作成できない可能性もあります。実際の詳しい作成方法は自動実行ツールのマニュアルを参照してください。

#### 4.2.5.4 テストの実施と負荷測定

スクリプトを実行するためのプログラムをMetaFrame XPサーバ上にコピーし、公開アプリケーションに登録します。クライアントから公開アプリケーションを起動すると自動的にアプリケーションが操作されるようになると思います。ここまでくれば後は実際にシナリオを走らせて負荷を計測するだけです。

ICAクライアントをクライアントマシン上で起動する操作はリモート実行ツールがあれば簡単に実現できます。Windows2000であればWSHとWMIでやるのもひとつの方法です。リモートでの起動にこだわらずクライアントマシン上で定刻になったら接続するようにatなどを設定しておくのでもかまいません。

実際に負荷の計測にはWindows2000に付属のパフォーマンスモニタを利用します。このとき基本的に負荷は平均で見えるようにしてください(瞬間的なピークとしてはWindowsのスケジューラは100%を示すようにスケジューリングしてしまいます)。主に測定すべきデータは以下に示すカウンタです。

##### メモリ関係のカウンタ

Memory - AvailableBytes

仮想メモリの空き領域の合計です。瞬間の値を示します。

Memory - CommittedBytes

仮想メモリのコミット(使用されているという意味で捉えてかまわないでしょう)された容

量の合計です。

Memory - Page Faults/Sec ページフォールトが1秒間に何回発生しているかの値です。これにはハードページフォールトとソフトページフォールトの両方が含まれます。

Memory - Page Input/Sec

ハードページフォールト（実際にハードディスクとやりとりしなければならないページフォールト）が発生し、ハードディスクからメモリに読み込まれたページ数です。

Memory - Page Output/Sec

1秒間にハードディスクに出力されたページ数です。

Memory - Pages/Sec

仮想メモリを利用する上で、ディスクに対してI/Oが発生したページ数です。メモリ不足に陥っているかどうかの判断はこの値を利用します。

Terminal Services Session - Working Set Peak（抜き出しで1つのセッション）

測定している対象セッションで必要になったWorking SetのPeakが保持されます。ユーザ1人あたりの最大メモリ使用量の目安になります。

#### プロセッサ関係のカウンタ

Processor - %Processor Time(\_Total)

プロセッサを実際に使用している時間の割合です。これが長い時間80%を超えるような状態ではプロセッサがボトルネックになっていると判断することができます。

System - Processor Queue Length

プロセッサキューにあるスレッド数です。これが大きい値を示す場合、プロセッサがボトルネックになっていると判断することができます。

Processor - % Privileged Time

Processor - % User Time

それぞれ、特権モードで実行された時間の割合と、ユーザモードで実行された時間の割合を示します。メモリ不足によりスラッシングが発生している場合には特権モードでの時間の割合が高くなります。

#### ディスク関係のカウンタ

Physical Disk - % Disk Time

ディスクがビジー状態にある経過時間の割合です。

Physical Disk - Avg. Disk sec/Transfer

平均ディスク転送時間です。

#### ネットワーク関係のカウンタ

Network Interface - Bytes Received/sec

指定したインターフェースで受信したバイト数です。

Network Interface - Bytes Sent/sec

指定したインターフェースで送信したバイト数です。

#### ターミナルサービス関係のカウンタ

Terminal Services - Active Sessions

アクティブなセッションの数です。セッション数の変化が読み取れます。

パフォーマンスモニタを使用してデータを得る際には、データの形式をCSVとします。これにより、データをExcelで可視化してグラフ化するのが容易になります。

ほとんどの場合、必要なメモリ量は簡単に求められるため、サーバのサイジングという目的



ではCPUがどれくらい使用されるかが主な調査項目になると思います。

ネットワーク関係の値には注意が必要です。Officeなどのアプリケーションであればほとんどの場合考慮は必要ないと思いますが、クライアント/サーバ・システムのプログラムを利用する場合に、サーバとクライアント間の通信が結果に混ざってしまわないようにする必要があります。この場合には、NICを複数用意してICAのトラフィックとクライアントサーバのトラフィックを分離したり、PacketShaperなどの外部機器を使用し、特定の機器との間の通信量だけを測定するようにします。

また、サイジングを終えてネットワーク構成の決定を行う場合に見落としがちなのは、通信の遅延時間です。MetaFrameではキーボードやマウスでの入力（上り）から画面へのレスポンス（下り）までの時間があまりに長いとユーザの操作に支障が出たり、クレームの元になったりします。このための指針として、シトリックスでは最大300msという値を公表しています。個人的な経験からいえば、実際には200msでも、ユーザには、体感的に遅延が目立つと思います。

テストにおいて、何をもちてシステムの限界に達したと判断するかは、通常、下記のような基準のうち、1つか2つの条件を満たしたところとします。

- ・CPUの利用率が平均して80%を超える状態を示すようになった
- ・テストスクリプトが実行を終えるまでの時間が1ユーザ時の115%に達した
- ・Processor Queue Lengthが1CPUあたり12を超えるようになった
- ・ある処理を行った場合の処理時間が規定値を超えた

この条件は絶対的なものではなく、使用するアプリケーションや、処理時間の遅延がユーザにあたる影響などを考慮して調整すべきです。システム側の物理的な最大能力を評価するのではなく、ユーザ側の観点から実用的な最大サポート能力を評価することが重要です。

また、基本的にMetaFrame XP/ターミナルサービス環境ではCPUの数やクロックには比例した結果になりやすく、また使用するメモリ量は固定的に求まります。このため対象となるアプリケーションでの結果をあるサーバで得られた場合には、同一のCPUアーキテクチャを採用したサーバ間では比較的正しい値を推測することが容易にできます。

ここで同一アーキテクチャといったのは、CPUが次の世代になり、2次キャッシュの容量や最適化が進むと、また違った挙動となるためです。

#### 4.2.6 サーバ1台のメモリ量とサーバ台数の算定

サーバ1台あたりの最大ユーザ数が求めれば、必要なメモリ量は下記の式より求めることが可能です。

システムの使用量 + (サーバの最大同時ユーザ数 × ユーザ1人あたりの使用量)

ここで、システムの使用量としては128MB程度を想定するとよいでしょう。ユーザ1人あたりの使用量はパフォーマンスカウンタのTerminal Service SessionsのWorking Set Peakにより求めることができます。自動実行ツールを利用していたセッションの結果からは、自動実行ツール自身が使用していたメモリ量を無視するようにしてください。

ベンチマークの結果から1台のサーバでサポートできる最大人数が求まっていると思います。

この結果と上記のメモリ量で、ターミナルサーバ1台のメモリ量は決定することができます。ただし、ターミナルサーバ1台で全ユーザをサポートするのは稀です。耐障害性を確保する意味でも、複数台のサーバを用意する必要があります。1台あたりの最大ユーザ数と、サーバファーム全体でサポートしなければならない人数から下記の計算でサーバ台数を決定するとよいでしょう。

冗長度 = 1.5 ~ 2.0 (高いほど冗長構成)

サーバ台数 = 冗長度 × (最大ユーザ数 / サーバ1台での最大ユーザ数)

たとえば、ベンチマークの結果は1サーバあたり最大100ユーザ同時であったときに、冗長構成をとり、サーバファーム全体で最大同時1000ユーザをサポートするのであれば、

サーバ台数 = 2.0 × (1000/100) = 20台

となります。

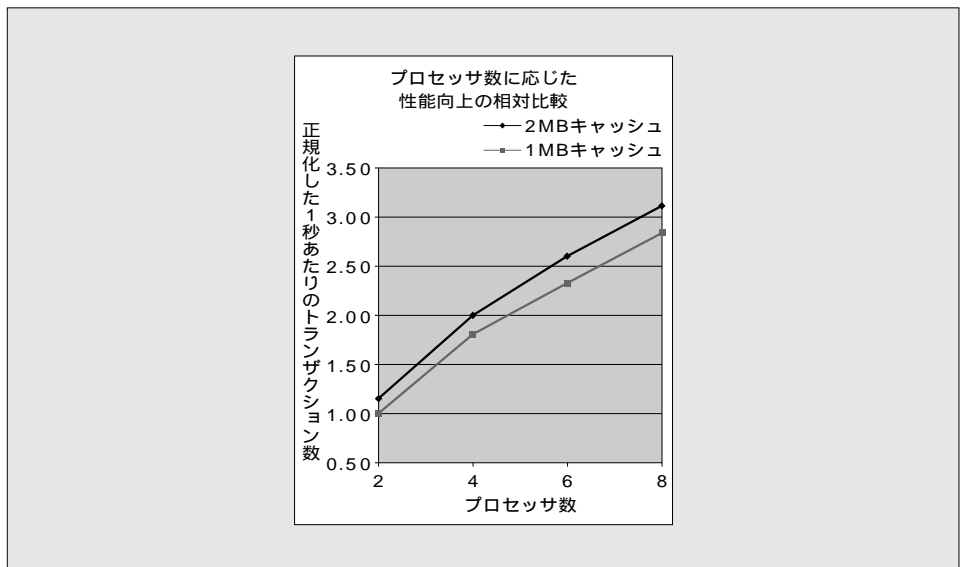
昨今、サーバ1台あたりの値段が下がっています。またMetaFrame XPのサーバライセンスは無制限台数にインストールが許可されています。したがって、ある程度余裕をもたせた台数での構成をお勧めします。トランザクションに顕著な差がありません。

#### 4.2.7 キャッシュメモリの影響

MetaFrameサーバの性能は、CPUの周波数にほぼ正比例します。逆に高価な2次キャッシュ、3次キャッシュの大きさは、性能に比例しません。

図15は、同じCPU仕様のMetaFrameサーバで、1MBキャッシュと2MBキャッシュのトランザクション数の増えかたを示しています。

図15



2GBのメインメモリとPentium III Xeonプロセッサ700MHzを搭載したProLiant 8500サーバにMetaFrameをインストールしてテストを実施してみました。キャッシュサイズを1MBの場合と2MBの場合で比較した結果が図15です。結論として、クライアントユーザに対する応答時間に識別できるほどの変化は見られませんでした。より大容量なキャッシュを使用した場合、性能データは、カーネルと割り込み処理が向上していることを示していますが、これが性能や密度（プロセッサあたりのユーザ数）を向上させるには至っていません。コンパクトテストラボの分析としては、MetaFrameサーバ環境の場合、512KB以降、キャッシュサイズを増やしても、向上はほとんど、あるいはまったく見られません。

#### キャッシュメモリ

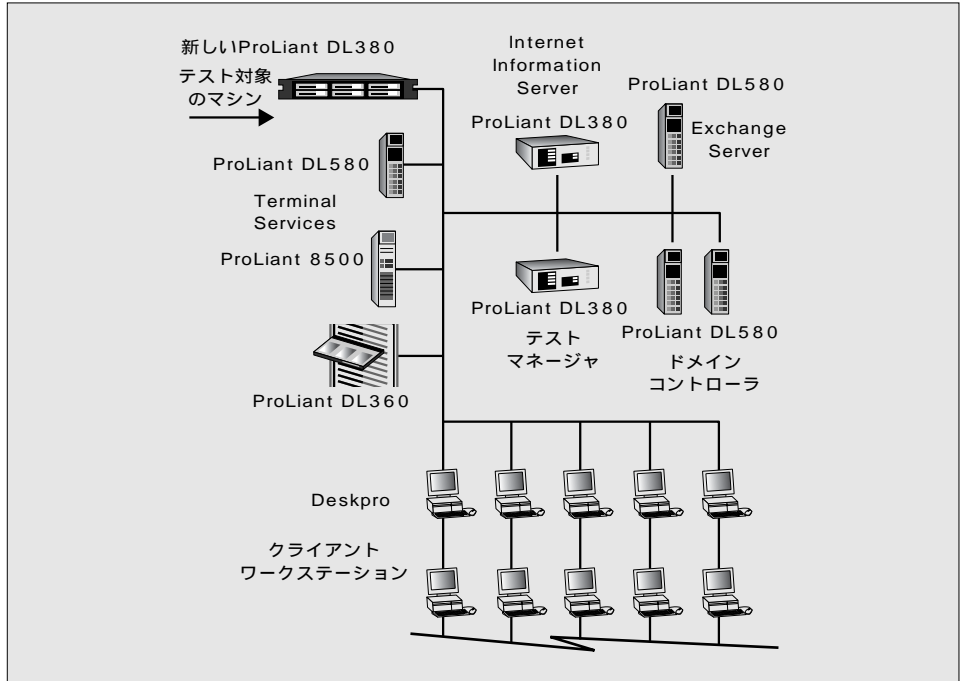
キャッシュメモリとは、プロセッサの近くに設置された小容量の高速なメモリを指し、ここに最近使用されたデータのコピーを保持することで、システム性能を向上させます。キャッシュメモリは、一般的にはプロセッサと同じ速度で稼動するため、キャッシュメモリに格納された情報へのアクセスは、メインメモリに格納された同等の情報にアクセスするよりも、はるかに高速です。ほとんどのプログラムは、同じ命令やデータを繰り返し使いますが、プロセッサは、この反復性のおかげで頻繁にキャッシュから情報を取得することができ、キャッシングの効果が得られます。プロセッサがキャッシュから取得できる機会が増えれば増えるほど、システム性能は向上します。実際、今日のプロセッサ設計者は、システム性能の向上を図るため、キャッシュを複数のレベルで搭載しており、通常は、小さなプライマリ（レベル1、または1次）キャッシュと、これより大きなセカンダリ（レベル2、または2次）キャッシュを構成しています。データが要求されると、プロセッサは、まず1次キャッシュにアクセスします。要求されたデータが1次キャッシュに存在しない場合、プロセッサは次に2次キャッシュへアクセスし、それでも存在しなければ低速なメインメモリにアクセスします。

理論的には、キャッシュが大きければ大きいほど、各プロセッサは、自分自身のローカルキャッシュにより多くのデータを保持することができ、その結果、各リソースへのアクセスで発生するシステム内の他のプロセッサとの競争が減るため、マルチプロセッサシステムのスケーラビリティと性能が向上するはずですが、しかし、キャッシュサイズを増やすと、他のプロセッサも、そこに格納されたデータと同じデータに別途アクセスする必要性も増えてきます。複数のキャッシュ間でコヒーレンシ（整合性）を維持および管理するために、システムバス上でトラフィックが発生しますが、このオーバーヘッドは、使用するプロセッサ数とキャッシュ容量の増加に伴って増えてしまいます。

### 4.2.8 ベンチマークセンター

ユーザが独自に開発したアプリケーションが、どの程度のサーバで、どれだけのユーザを同時にサポートできるかという情報は、実際にアプリケーションベンチマークを行わないと、測定することはできません。コンパックでは、自動測定ツールを使用して、最大500ユーザのMetaFrameベンチマークが可能な「MetaFrameベンチマークセンター」(図16)を1999年に、日本で開設しており、利用が可能です。

図16 MetaFrameベンチマークセンターのサーバ構成例



コンパクトでは、各種最新サーバのMetaFrameXPでのベンチマークをWebに掲載しています。

<http://www.compaq.co.jp/solution/sbc>

サーバを選択する際のひとつの目安にいただければと思います。このベンチマークは、MS-Officeを利用するユーザを、ヘビーユーザ、ミディアムユーザ、ライトユーザの3種類に分け、サーバのCPUが80%の稼働率の状態で、それぞれ同時に何ユーザの動作が可能であるかを測定したものです。

## 4.3

# ターミナルサービスとMetaFrame XPのライセンス

MetaFrame XPのユーザは、ターミナルサービスのライセンスとMetaFrame XPのライセンスが必要です。

### 4.3.1 TSCALの管理

#### 4.3.1.1 CALとTSCALが要るクライアントOS

MetaFrameサーバに接続するクライアントは、WindowsのCAL（クライアント・アクセス・ライセンス）とTSCAL（ターミナルサービス・クライアント・アクセス・ライセンス）の両方のライセンスをマイクロソフトから購入する必要があります。