

---

# メインフレームのメタボ診断の応用

－ CPU ダウングレード実現へのプロセス －

株式会社アイビスインターナショナル

---

## ■ 執筆者Profile ■



有賀 光浩

1985 年 富士通株式会社入社

2003 年 富士通株式会社退職

2004 年 株式会社アイビスインターナショナル  
設立 代表取締役

## ■ 論文要旨 ■

GS から GS へ CPU をリプレースするとき、CPU 能力を 2 ～ 3 割上げることが慣例だった。しかし、最近ではコスト削減のため CPU 能力を低下（ダウングレード）させる要請が増えている。現場では長年の経験と勘に頼って実施していることが多い。

昨年発表した GS メタボ診断はお客様システムでの実践を重ね、メタボの検出から CPU ダウングレードを実現するためのツールとして進化している。

この過程で、GS のリプレース難易度チェックシートが生まれ、後に CPU ダウングレードの実現性を評価する手法も確立した。具体的には、最初に 3 つの簡易的な検証を実施する。3 つとは、①CPU の余裕度、②ダウングレード後の性能予測、③リスク評価である。ダウングレードが現実的であると判断されれば、次に 3 つの本検証と改善を行う。本論文では実例を使い検証方法を紹介する。

CPU リプレースに関するリテラシーを高め、お客様の満足度を高めていきたい。

## ■ 論文目次 ■

<b>1. はじめに</b> .....	《 3》
1. 1 当社の概要	
1. 2 背景	
<b>2. GS メタボ診断の効果（事例）</b> .....	《 5》
2. 1 GSのメタボとは	
2. 2 実例：Xシステムの稼動状況	
2. 3 CPU/I/O頻度分析の活用例	
2. 4 オンライン処理の稼動状況	
2. 5 バックアップ・リカバリ運用	
<b>3. CPU リプレースに関する考察</b> .....	《 10》
3. 1 GSリプレースの難易度の可視化	
3. 2 CPU性能と能力の違い	
<b>4. CPU ダウングレードの実現性評価手法</b> .....	《 12》
4. 1 CPU 余裕度の把握	
4. 2 ダウングレード後の性能予測	
4. 3 リスク評価	
<b>5. 評価と課題</b> .....	《 15》

## ■ 図表一覧 ■

<b>図 1</b> 富士通メインフレームの出荷状況 .....	《 3》
<b>図 2</b> バッチジョブの処理時間の推移 .....	《 4》
<b>図 3</b> GSのメタボ化 .....	《 5》
<b>図 4</b> Xシステムの6日間の稼動状況 .....	《 6》
<b>図 5</b> CPU使用率とIOPSの推移 .....	《 6》
<b>図 6</b> CPU使用率とIOPSの関係 .....	《 7》
<b>図 7</b> CPU/I/O頻度分析 .....	《 7》
<b>図 8</b> オンライン処理の稼動状況 .....	《 9》
<b>図 9</b> CPU性能を20%下げたときのCPU使用状況（シミュレーション） .....	《 12》
<b>図 10</b> CPU性能を20%下げたときの処理時間の予測（シミュレーション） .....	《 13》
<b>図 11</b> 夜間バッチの稼動状況（D） .....	《 14》
<b>表 1</b> 1995年と2005年のシステム環境の比較 .....	《 4》
<b>表 2</b> システムのCPU/I/O頻度 .....	《 7》
<b>表 3</b> バッチジョブのCPU/I/O頻度 .....	《 8》
<b>表 4</b> CPUとI/Oのバランスがくずれる原因 .....	《 8》
<b>表 5</b> GSのリプレース難易度チェックシート .....	《 10》
<b>表 6</b> CPU性能と能力の区別 .....	《 11》

# 1. はじめに

## 1. 1 当社の概要

株式会社アイビスインターナショナル（所在地：東京都）は、富士通メインフレーム（GS21, PRIMEFORCE）の性能コンサルティングを事業としている（ホームページ：<http://www.ibisinc.co.jp/>）。

代表の有賀光浩(Ariga Mitsuhiro)は富士通株式会社で 18 年間 SE として活躍、1992 年からの 11 年間は共通技術部門でメインフレームの性能に関する技術支援、顧客システムの性能トラブル対応を担当した。対応システム数は国内外合わせて 1,000 以上に及ぶ。

富士通退職後、2004 年に株式会社アイビスインターナショナルを設立。お客様が満足感を持ち、安心してメインフレームを使い続けて頂くためのコンサルティングを提供し、FUJITSU ファミリー会の論文投稿などの情報発信も行っている。

## 1. 2 背景

### （1）メインフレームの国内市場

（社）電子情報技術産業協会（JEITA）の調査によると、2008 年度のメインフレームの出荷台数は 601 台（前年度比 84%）、金額は 1,197 億円（前年度比 72%）と大幅に減少した。これは、オープン化が進んだというより、CPU リプレースを延期した影響だと考える。10 年前の実績と比較すると出荷台数は 20%、出荷金額は 16%まで減少している。稼働台数は公表されていないが、出荷実績から算出すると 2008 年度は約 5,400 台と予想する。

2008 年には IBM が System z10 EC と BC を発表し、今年は富士通が GS21 1600 と 1400 を発表している。

### （2）富士通メインフレームの発展

メインフレームは時代遅れで古臭いコンピュータというイメージを持たれているかもしれない。富士通は、図 1 に示すようにおよそ 3 年ごとに新しい CPU を出荷している。（以降、富士通のメインフレームを GS と略す）。

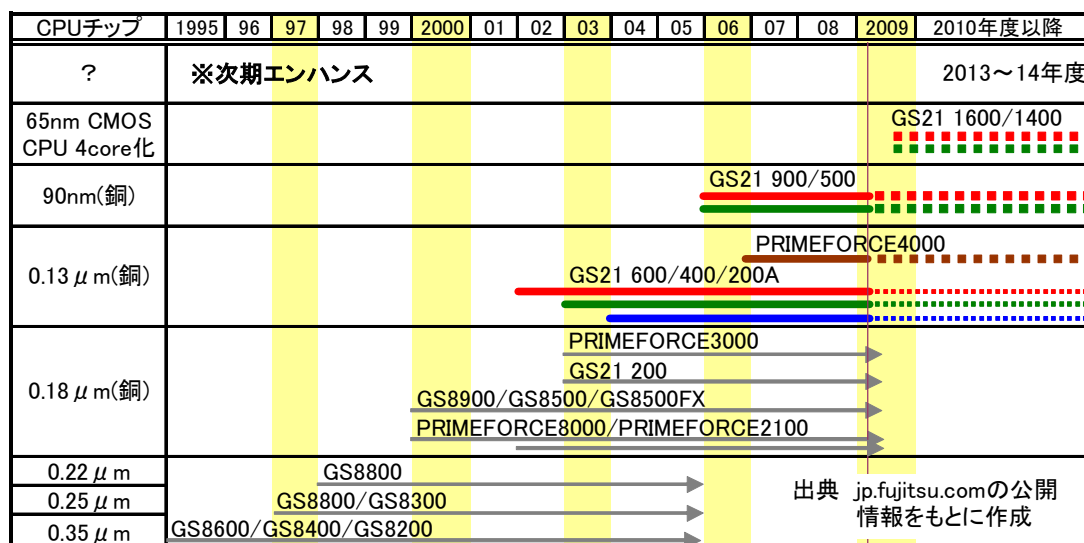


図 1 富士通メインフレームの出荷状況

OS は、大規模システム向けの MSP-EX が 1990 年に、中規模システム向けの XSP が 1991 年に出荷されてからは大きな変化はない。ミドルウェアの中核である AIM (V20)や RDB II (現在の SymfoWARE)，仮想化システムを提供する AVM-EX もほぼ同時期に出荷されている。

ディスク装置は、1995 年に GS 初のディスクアレイ F6493，1997 年に F6495 が出荷された。ユーザプログラムを修正せずに、劇的な性能向上を図ることができた。

1980 年代に作られた業務システムも、1990 年代に OS 関連の移行を行っていれば、現在のハードウェア上で問題なく動作する。

### (3) 性能の向上

最近 10 年間でシステム性能がどのくらい向上したかを把握するために、表 1 に示すシステム環境を使って考える。CPU が 2 倍強性能向上したことに比べ、メモリ搭載量は 8 倍、ディスクの I/O 性能は 13 倍も向上している。これは、CPU と I/O のバランスが崩れたことを意味している。

表 1 1995 年と 2005 年のシステム環境の比較

		1995 年	2005 年
CPU		GS8400/10	GS21 400/10N (2 倍強)
メモリ搭載量		128MB, 最大 512MB	1GB (8 倍), 最大 32GB
チャンネル		BMC 4.5MB/S	OCLINK 17MB/S (3 倍強)
ディスク装置		F6427, ディスクキャッシュ搭載 平均 I/O 時間 20ms	ETERNUS6000 平均 I/O 時間 1.5ms (13 倍)
プログラム	言語	COBOL	COBOL
	データベース	NDB, VSAM	NDB, VSAM 現在は SymfoWARE で開発 (1 倍未満)
	リージョン	およそ 1~4MB/ジョブ	SymfoWARE 使用時は 10MB 以上あり
	性能品質	良好	SymfoWARE を使うと CPU 頻度が高くなる傾向がある

注) 2005 年のカッコ内は 1995 年との性能比

1995 年当時、単体処理時間が 1 時間だったバッチジョブは、2005 年にはハードウェアの性能向上だけで図 2 に示す通り約 1/10 (6 分) に短縮する。I/O 時間の短縮が大きく、結果として CPU 占有率 (=CPU 時間/処理時間) が 29%と高くなり、CPU 依存型になったと言える。GS21 モデル 200A や PRIMEFORCE の下位モデルだと CPU 占有率は 50%を超えている。

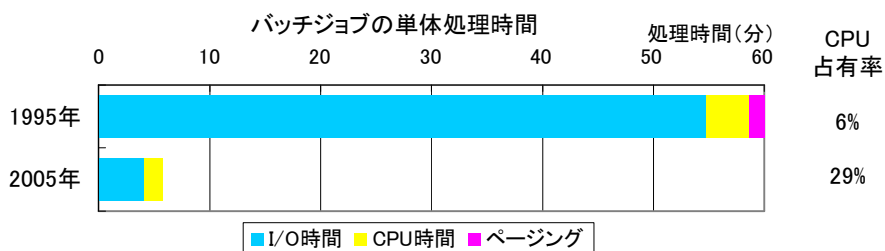


図 2 バッチジョブの処理時間の推移

## 2. GS メタボ診断の効果（事例）

GS のメタボ化については、2008 年の論文「メインフレームのメタボ化とその診断手法」で論述している。本章では、GS のメタボについて簡単に説明した上で、GS メタボ診断の具体的な効果を事例を使って紹介する。

### 2. 1 GS のメタボとは

GS のメタボとは、図 3 中央「メタボな状態」に示すように、外のメタボと内のメタボによりシステムが膨張している状態と定義する。各々について説明をする。

外のメタボ（過剰投資）とは、必要以上に大きいハードウェアを導入している、または効果的に使われていない状態をいう。

- 例
- ・CPU を平均 20% しか使っていない。
  - ・性能上効果がないのにメモリを増設した。
  - ・システム記憶（SSU）を導入しているのに使っていない。

内のメタボ（リスク）とは、ソフトウェアが必要以上のリソースを使用する、またはシステムの信頼性、安定稼動を阻害する可能性がある状態をいう。

- 例
- ・重い業務プログラムが動くと CPU 使用率が 100% になる。
  - ・オンラインレスポンスが数 10 秒の業務を仕方なく使っている。
  - ・DISK 障害が起きるとリカバリできない状態で運用している。

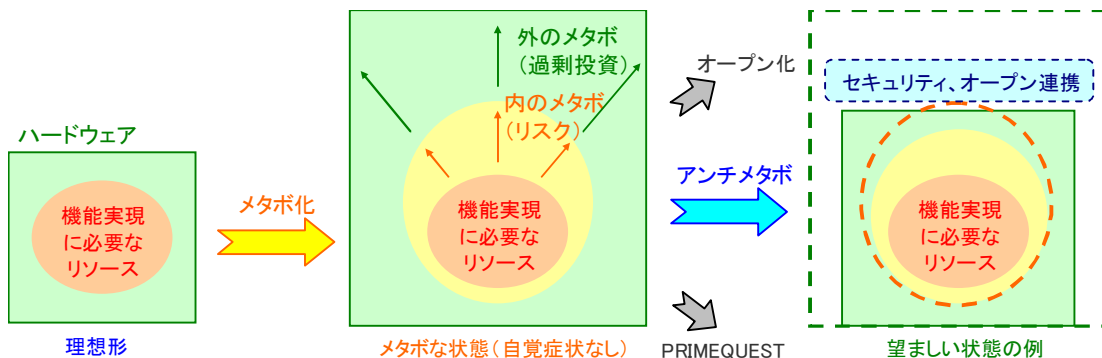


図 3 GS のメタボ化

GS のメタボは、人のメタボと似ていて「病気」ではない。即ち、今すぐにトラブルを誘発するものではないが、放置しておいても改善することはない。オープン系サーバの平均 CPU 使用率は 20～30% と言われており、GS よりもはるかにメタボである。GS は CPU 使用率が 100% になっても問題なく運用できる優れたサーバであり、この特長を生かすことが運用上のポイントである。

GS メタボ診断はメタボな状態を検知し、無駄を無くすこと、問題が発生する前に適切に対処することを目的としている。従来のキャパシティ管理は、業務量が増加すること（右肩上がり）が前提でリソース不足や性能トラブルの未然防止を目的としている。多くの GS では業務量が減少傾向（右肩下がり）にあり、過剰投資の防止や性能品質の向上を重要視している。

コスト削減のため、CPU 能力を低下させる（ダウングレード）要請が増えている。しか

し、作業プロセスが標準化されておらず、現場では長年の経験と勘に頼って実施されることが多い。CPU 使用率がどんなに低くても、CPU をダウングレードすれば CPU 時間が延びるため処理時間は必ず遅くなる。処理時間の延びを抑えるには内のメタボの改善としてデータベースやユーザアプリのチューニングが必要となることもある。処理時間がユーザ業務の許容範囲内に収まっているか判断するためには移行後の処理時間を予測することも重要である。

## 2. 2 実例：Xシステムの稼動状況

図 4 は、Xシステムで性能データを 6 日間（A～F）取得し、CPU 使用率、ジョブ数、多重度と IOPS（1 秒間の I/O 回数）の推移を表したグラフである。

最大 CPU 使用率（10 分間の平均値の最大）は 6 日間ともほぼ 100%である。ジョブ数、多重度、IOPS には大きな変化はない。

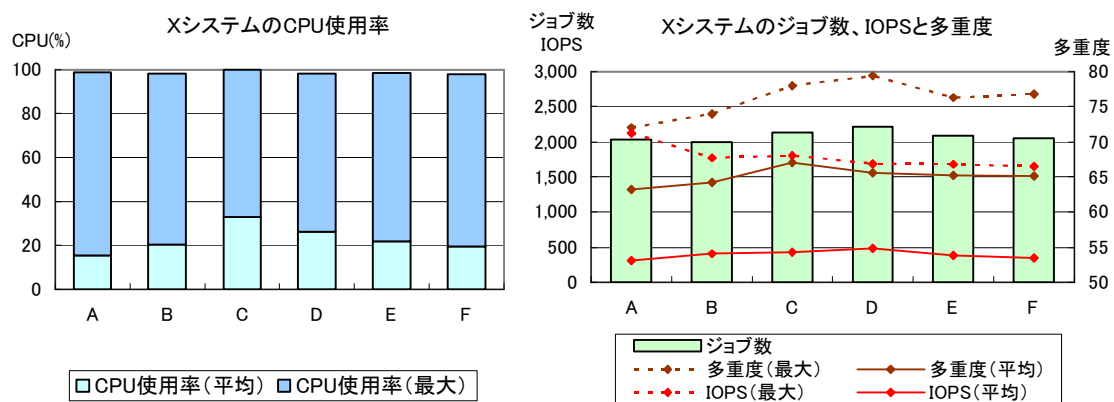


図 4 Xシステムの6日間の稼動状況

以降、平均 CPU 使用率の高い C と D のデータを使って考察を進める。

図 5 は、CPU 使用率（棒グラフ）と IOPS（折線グラフ）の 10 分ごとの推移を表している。CPU 使用率と IOPS の山は類似しているが、C の午後は CPU 使用率が高くなっていることがわかる。

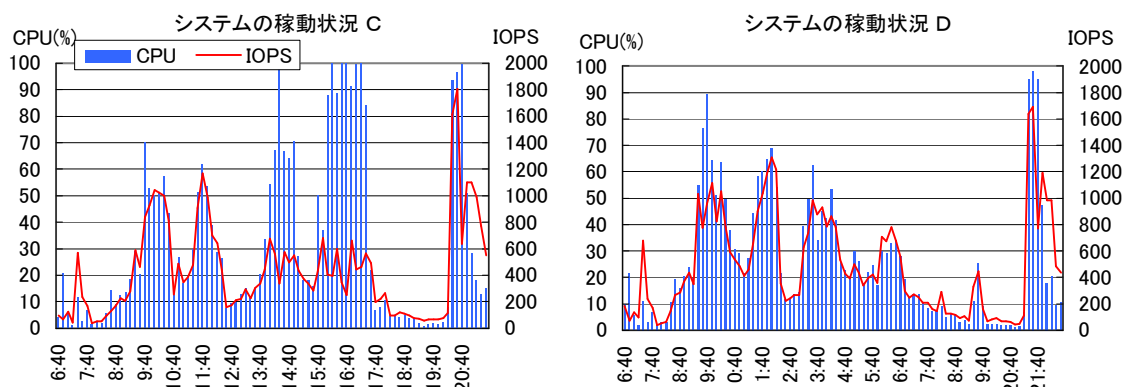


図 5 CPU 使用率と IOPS の推移

図5と同じデータを使いCPU使用率とIOPSの相関関係を表すグラフを図6に示す。およそ1,700 IOPSでCPUが100%に達することがわかる。Cではこの相関からずれる集まり（左上の円内）が見られる。これが、正常なのか異常なのかを客観的に判断することはまだむずかしい。

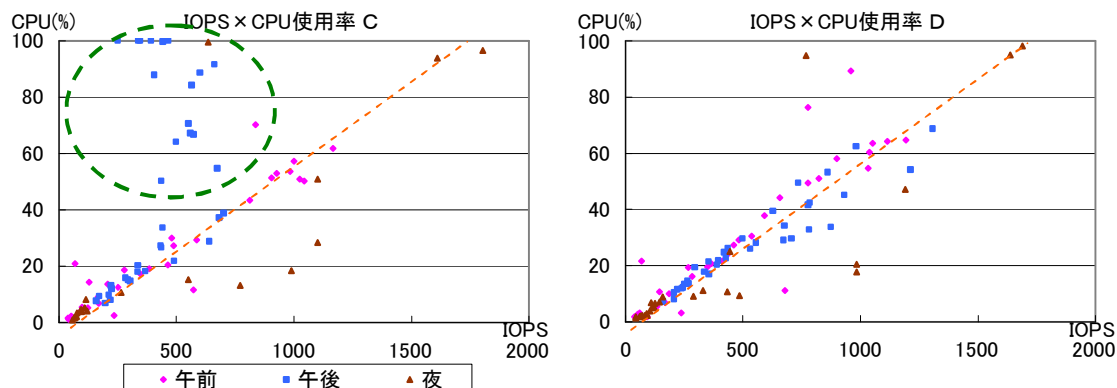


図6 CPU使用率とIOPSの関係

## 2. 3 CPU/I/O 頻度分析の活用例

CPU/I/O 頻度とはCPUとI/Oのバランスを示す指標で内のメタボを検出するのに有効なツールである。表2に示すように標準値が20～40になるように算出している。

表2 システムのCPU/I/O 頻度

CPU/I/O 頻度	0～20	20～40	40～80	80 超
傾 向	I/O 頻度が高い (I/O バウンド)	標 準	CPU 頻度が高い (CPU バウンド)	CPU ループ (CPU の無駄)

図5、図6と同じデータを使いシステムのCPU/I/O 頻度とIOPSの関係を表すグラフを図7に示す。CPU/I/O 頻度の多くが20～40の間に入っているが、Cでは80を超える集まり（右下の円内）があることがわかる。これは図6左の円と同じものである。

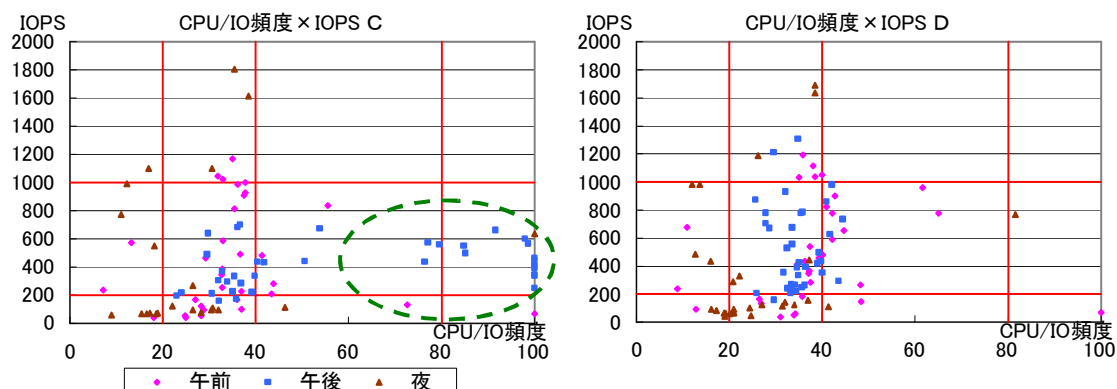


図7 CPU/I/O 頻度分析

システムのCPU/I/O 頻度は、同じ時間にCPU頻度の高いジョブとI/O頻度の高いジョブが

同時に実行されると相殺され標準値に近づいてしまう。そのため、ジョブごとの CPU/I/O 頻度を把握することが重要となる。CPU 時間の長いバッチジョブについて算出した結果を表 3 に示す。

表 3 バッチジョブの CPU/I/O 頻度

ジョブ名	日	開始時刻	処理時間	CPU 時間	EXCP 回数	CPU 占有率	CPU/I/O 頻度
####K010	C	15:57	1:41:19	0:42:57	601,970	42.4%	284
####DKRI	C	13:47	1:07:39	0:30:55	893,372	45.7%	137
####DKRI	C	15:34	1:20:36	0:25:51	616,812	32.1%	167
####HIGA	D	21:25	0:18:31	0:08:09	59,296	44.0%	546
####HIGA	C	20:25	0:17:30	0:08:03	59,287	46.0%	540
####MMNT	D	9:20	0:42:22	0:07:35	467,397	17.9%	65
####XFKA	C	9:33	1:09:07	0:07:25	1,279,614	10.7%	23
####XFKA	C	11:17	0:59:12	0:07:24	1,304,208	12.5%	23

図 7 でシステムの CPU/I/O 頻度を上げているジョブは、####K010、####DKRI が有力な候補である。更にプログラムまでブレークダウンすることができる（割愛）。

CPU/I/O 頻度が 100 を超えるようなプログラムはその原因がプログラム内にあることが多く、2006 年前期論文「メインフレーム変革！ お客様視点・現場重視の性能評価」で紹介した仮説検証型性能解析手法を使って改善を進めることが効果的である。CPU 頻度や I/O 頻度が高くなる原因の一例を表 4 に示す。

表 4 CPU と I/O のバランスがくずれる原因

	原因	備考
CPU 頻度が高い 【CPU の無駄】	COBOL のデバッグオプションがついている	COBOL 標準のプログラムで調査可能
	コンパイルと結合編集の不整合	
	製品のサブルーチンを CALL している	文字コード変換、ソート等
	DB に大量の排他をかけている	
	非効率なテーブルサーチ、ループ 無駄な SQL 命令の発行	プログラミングの問題
I/O 頻度が高い	カタログファイルへの I/O が多い	特に XSP は注意
	サブルーチンを CALL するたびにサブ ルーチン内で毎回 OPEN/CLOSE を繰り返している	サブルーチンの作りの問題
	ISAM を使っている	

## 2. 4 オンライン処理の稼動状況

2. 2 ではバッチジョブの CPU/I/O 頻度分析について考察したが、並行してオンライン処理の稼動状況を把握することも重要である。多くの GS 上では数 100 のオンラインプログラムが動作している。これを 1 本 1 本評価するのは非効率であり、特定のプログラム（重要なオンライン処理など）だけを評価すると問題を見落とす可能性もある。2008 年の論文で紹介したが、オンライン処理の稼動状況を一目でわかるように表したグラフが図 8 である。



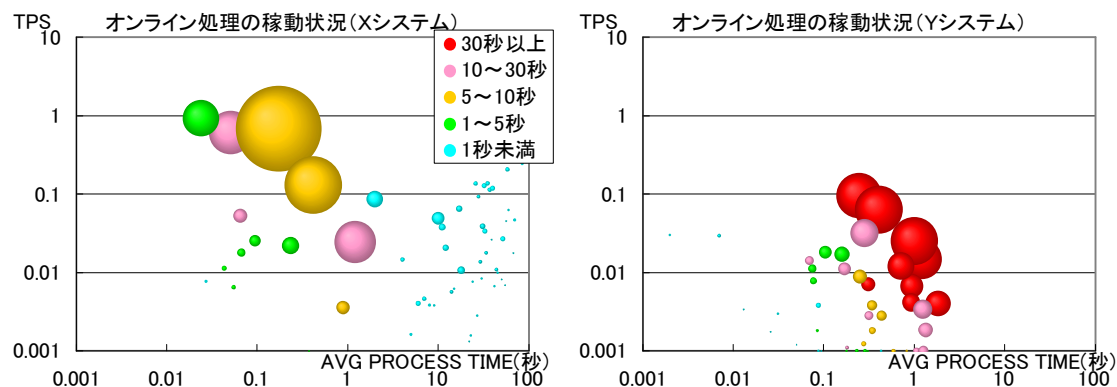


図8 オンライン処理の稼動状況

大きい円が使用頻度の高いオンラインプログラムで、右図にある濃い円は最大レスポンス時間が30秒以上を表している。Xシステム（グラフ左）は良好だが、Yシステム（グラフ右）は問題がありそうと判断する。オンラインレスポンスが30秒ということは、自分自身で30秒処理しているか、データベースの排他待ち等で待たされているのかのいずれかである。このようなプログラムについては、レスポンスの実態（ユーザが認識しているか）と妥当性を確認する必要がある。

オンライン処理に問題があるシステムは、他の問題を潜在していることが多い。

## 2. 5 バックアップ・リカバリ運用

今でもGSが使われている理由の一つに高信頼性がある。実際、本体装置やディスク装置のハードウェア障害は最近耳にしたことがない。ところが、GSメタボ診断を行っていると、データベースのバックアップ・リカバリ運用の不備を目にすることが度々あった。長年にわたり重大なシステム障害を起こさなかったのは単に運が良かっただけかもしれない。問題を整理すると以下ようになる。

### (1) 重大な問題

- ・データベースのバックアップ(JXATDUMP)を取得していない。

### (2) 中程度の問題

- ・HLF（更新後ログファイル）を退避していない。
- ・AIMを初期化モードで起動した後、バックアップを取得していない。
- ・データベースのリカバリを行ったことがない、手順書がない。
- ・データベースのバックアップに漏れがある。

### (3) 小程度の問題

- ・HLFバッファの枯渇が発生している。
- ・HLFの容量が小さく切り替わりが早い。
- ・トランザクション内でHLFに多量のログデータを出力している。
- ・ログファイル関連のボリュームが分散されていない。

AIMのログ環境は、システムを最初に構築したときから10年以上見直しをされていないことが多く、意識的に注意した方がよい。

### 3. CPU リプレイスに関する考察

GS から GS へ CPU をリプレイスするとき、CPU 能力を 2 ～ 3 割上げることが慣例だった。最近では、GS 上の業務のオープン化が進み、コストを抑えるために CPU 能力を低下（ダウングレード）したり、複数の GS を 1 台の GS に統合することも珍しくない。

#### 3. 1 GS リプレイスの難易度の可視化

CPU リプレイスを行うとき、現状分析を行わず、予算ありきで機種選定をしているケースが多く見受けられる。同等性能の CPU への移行なら問題ないとも考えるかもしれないが、これも正しくない。同等性能は平均値であり、速くなる処理もあれば遅くなる処理もある。クリティカルなジョブが遅延すれば性能問題となる。

GS の CPU リプレイスに関する指針がなかったため、性能から見たリプレイスの難易度を把握するためのチェックシートを作成した（表 5）。

表 5 GS のリプレイス難易度チェックシート

区分	チェックポイント	備 考
CPU	①CPU 能力が下がる、または単体 CPU 性能が下がる	CPU ダウングレード
	②移行前後でマルチプロセッサ (MP) を使っている	PRIMEQUEST も含む
	③CPU のアーキテクチャが変わる	PRIMEQUEST
AVM	④ネイティブシステムから仮想システム (AVM)、または仮想システム (AVM) からネイティブシステム	
メモリ	⑤メモリを減らす	
I/O	⑥チャンネルの能力（トータルスループット）が低下する	
	⑦DASD のアーキテクチャが変わる	内蔵ディスク
	⑧ボリュームを統合する	
メタポ 診断	⑨内のメタポである CPU 頻度が高いシステム、オンラインレスポンスが悪い 等	
他	・複数のシステムを一システムに統合する	見積もり必須
	・業務プログラムに大きな修正（DBMS の変更等）がある 新規業務の追加がある	見積もり必須

注）2008 年から PRIMEQUEST 上で XSP が動作する

#### ①～⑨での Yes の数による評価

- ・ 0 個      問題なし。①で CPU 能力が同等のときには事前の性能検証が必要。
- ・ 1～2 個      リスクがあるため事前に十分な検証が必要。  
①が Yes のときは処理時間が遅延することを周知徹底する。
- ・ 3 個以上      非常に危険。段階的に移行を進めるようにする。

① … CPU に余裕があっても処理は遅くなる。

② … MP 係数がどれだけ出ているか（出るか）把握がむずかしい。公表値よりも MP 係数が良い（性能が余計に出ている）ケースがある。

③ … PRIMEQUEST は CPU 特性が公開されていないので評価がむずかしい。

④ … AVM 配下で性能をコントロールするのはむずかしい。（3. 2 参照）

- ⑤ … 現状使用している仮想メモリ量と実メモリ量を正確に把握する必要がある。
- ⑥ … チャンネルに余裕があっても I/O 性能は低下する。
- ⑦ … 内蔵ディスクは一般のディスクアレイよりも性能が良いことがある。
- ⑧ … ボリュームの負荷が高くなり I/O 性能は低下する。
- ⑨ … CPU 頻度が高いプログラムは，CPU 性能に影響を受けやすい．システムに問題が潜在している可能性がある。

例 1 60MIPS の GS を 60MIPS の GS に単純移行

Yes は 0 … 問題ないが同等性能なので事前の性能検証が必要

例 2 60MIPS の GS を 50MIPS の GS にダウングレード

Yes は 1 つ (①) … 十分な検証が必要 (4 章で解説)

例 3 60MIPS の GS を 50MIPS の PRIMEQUEST (2 コアモデル) にダウングレード

Yes は 3 つ (①, ②, ③) … 移行は推奨しない

### 3. 2 CPU 性能と能力の違い

CPU 能力と性能を混在しているケースがあるが，この 2 つは区別して考える必要がある (表 6) ．

表 6 CPU 性能と能力の区別

	単体 CPU 性能が向上する (処理が速くなる)	単体 CPU 性能が低下する (処理が遅くなる)
CPU 能力が上がる (処理量が増)	一般的な CPU アップグレード	マルチプロセッサ化 (シングルプロセッサより遅くなる)
CPU 能力が下がる (処理量が減)	仮想システム (AVM) 化 (ネイティブより速い傾向あり)	一般的な CPU ダウングレード (★マルチプロセッサはリスク大)

仮想システム (AVM) は一般的に自動スケジューリングモードを使用する．ここで設定する CPU 配分は，CPU 能力をコントロールする機能であり，CPU 性能はコントロールできないことはあまり認識されていない．本件は，2006 年後期論文「メインフレーム変革！ 性能品質向上への挑戦 ―CPU 統合の見える化と改善―」で解説している．

マルチプロセッサ (MP) はそのオーバヘッドがシステムの特性により変化する．例えば，3 CPU モデルで MP 係数の公表値が 2.4 だとしてもこれは最悪値に近く，2.8 出ているシステムがあっても不思議でない．現行システムの MP 係数がどの程度なのか実測することができず (公開されていない) 予測もむずかしい．

シングルプロセッサでさえ業務処理の特性により性能は変化する．性能とは画一的なものではなく，ぶれるものだと認識することが重要である．

CPU リプレースを行うには，リスクを最小限に抑えることが重要である．

図 2 に示したように，最近の GS は CPU 依存度が大きくなっている．管理されている品質のよいプログラムの隣で管理されていない (本番業務でない) プログラムがグループして CPU を消費しているような例も多く見受けられる．CPU 頻度の高いプログラムはリスクの発生可能性を高めるため特に注意が必要である．

## 4. CPU ダウングレードの実現性評価手法

GS は安定稼働を保ちながら CPU 性能を段階的にダウングレードできる稀有なサーバである。しかし、ダウングレードの方法論は確立されておらず、経験と勘に頼っていることが多かった。どうすれば CPU ダウングレードのリスクを低減できるのか考察する。

CPU のダウングレードのリスクを評価するには以下のプロセスが効果的である。

A. CPU ダウングレードの現実性を簡易的に検証する（＝メタボ診断）

①→②→③→現実的か判断

B. 現実的と判断されたとき、本検証後改善をして判断する

①→②→③→改善→現実的か判断

① CPU 余裕度の把握

② ダウングレード後の性能予測

③ リスク評価

本章では、2. 1 で紹介した X システムを使い、CPU 性能を 20% 下げることが現実的かどうか検証プロセスを紹介する。なお、このプロセスは、複数のシステムを 1 つに吸収（統合）するケースでも応用できる。

### 4. 1 CPU 余裕度の把握

（1）簡易的な検証

図 4 の CPU 使用率の高い C と D について検証する。目的は、CPU ダウングレードの現実性の検証であり、それを保証するものではない。

CPU 性能を 20% 下げたとき、図 5 の CPU 使用率はどのように変化するかシミュレーションした結果を図 9 に示す。

- ・折線グラフ：元の CPU 使用率（グラフの間隔は 10 分間）
- ・棒グラフ：CPU 性能を 20% 下げたときの CPU 使用率（80% で頭打ちになる）

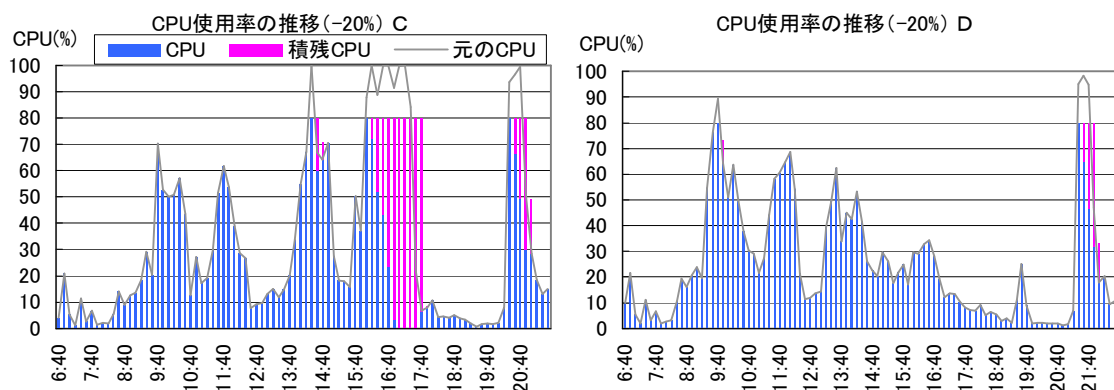


図 9 CPU 性能を 20% 下げたときの CPU 使用状況（シミュレーション）

CPU を積み残した時間帯は以下の通りである。

C : 14:20 から 20 分, 16:00 から 110 分, 20:20 から 40 分, 計 170 分間

D : 9:40 から 10 分, 21:20 から 40 分, 計 50 分間

C では 16:00 からの 110 分間継続して CPU がオーバフローする. この間のバッチ処理の遅延が運用に耐えられるか, またオンラインレスポンスに影響を与えないようにできるかがポイントとなる.

C, D 共通で夜間 40 分間 CPU がオーバフローする. 元は 30 分間 90%を超えていたので, 影響は小さいかもしれない.

CPU ダウングレードが現実的なものなのかを判断する一つのハードルである.

## (2) 本検証

検証すべきデータをどうやって取得, 入手するかが課題である.

例えば地方自治体のシステムで考えると,

- ・ 7 月のデータを検証しても, 繁忙期の 3, 4 月でないと意味がないような気がする.
- ・ CPU 使用率の最も高い 1 日を分析しても, その 1 日だけでよいのか疑問が残る.
- ・ 業務のピーク日とシステムのピーク日は必ずしも一致しない.
- ・ 「CPU 使用率が高い=ピーク日」とする理由の一つは分析を楽にすることである.

等の問題がある.

〇月〇日を選択した言い訳を考えるのではなく, 自分の責任で CPU ダウングレードを行うとしたら, 少なくともこの日のデータを検証しなければ安心できないという当事者意識で取り組むべきである.

2 章に紹介したように, CPU 使用率, IOPS と CPU/IO 頻度により分析すべき日を絞り込むことは効果的である. PDL/PDA だけでなく SMF も合わせて分析をする.

## 4. 2 ダウングレード後の性能予測

### (1) 簡易的な検証

CPU 使用率がどれだけ低くても, CPU 性能が低下すれば CPU 時間が長くなるため処理時間は遅延する. この遅延を感覚ではなく, 可視化することが目的である.

CPU 性能が 20%下がると CPU 時間は 1.25 倍になる. CPU/IO 頻度から C, D の標準的な性能モデルを作成し, バッチジョブの処理時間が過負荷時と疎負荷時でどうなるかシミュレーションした結果を図 10 に示す.

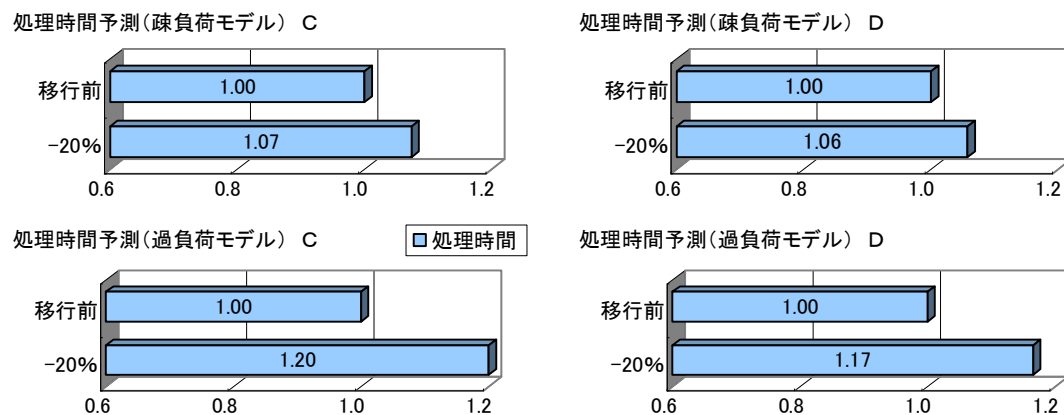


図 10 CPU 性能を 20%下げたときの処理時間の予測 (シミュレーション)

Cでは単体性能に近い状態（疎負荷モデル）で処理時間は1.07倍に、CPU使用率が100%に近い状態（過負荷モデル）では1.20倍まで遅延し、Dでは同様に1.06倍と1.17倍になることが予測される。処理時間の遅延によりジョブの多重度が上がると更に遅くなる可能性もある。処理時間1.2倍が、CPUダウングレードが現実的なものかを判断する二つ目のハードルとなる。

## （2）本検証

具体的なジョブやジョブネットがどれだけ遅くなるかを予測することが課題である。

図5のDについて、主な夜間バッチジョブの稼働状況を図11に示す。

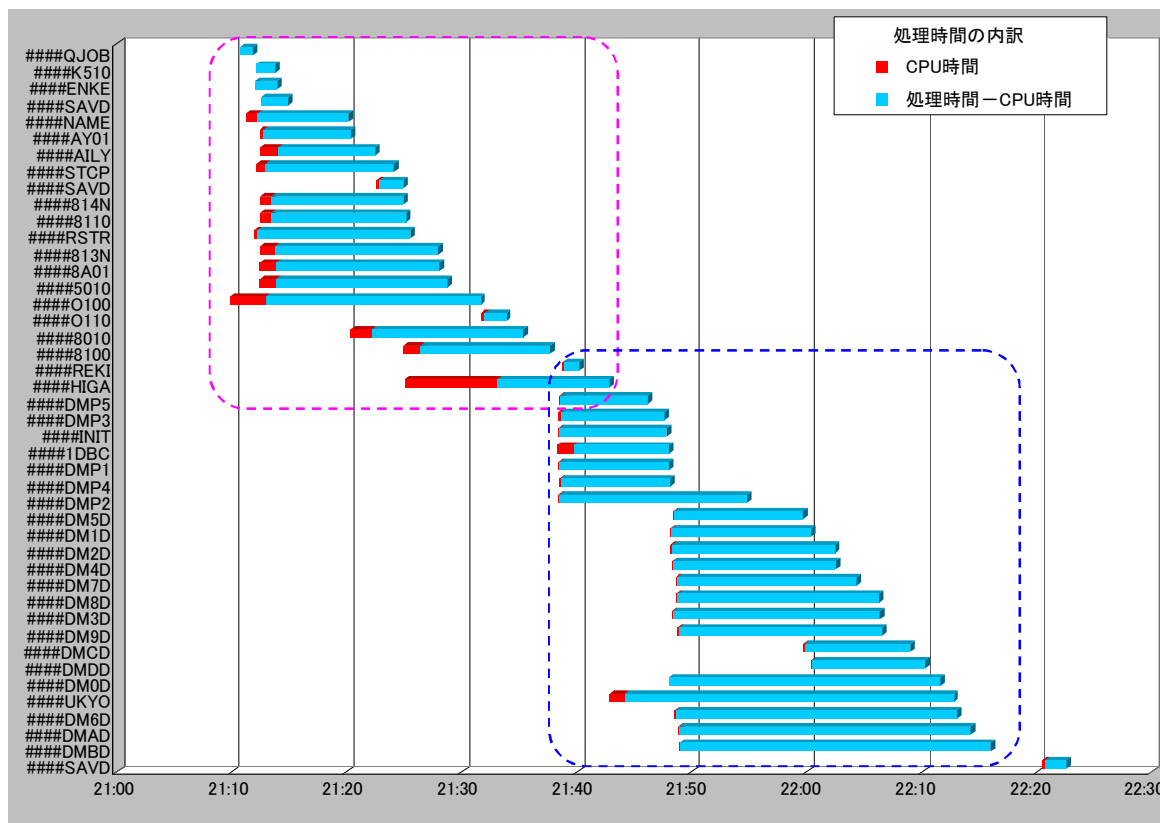


図 1 1 夜間バッチの稼働状況（D）

図5のCPU使用率の推移を合わせて見て頂きたい。

前半の21:10～21:40はCPUが過負荷状態なので30分が1.2倍の36分になる。

後半の21:38～22:16はCPUが疎負荷状態なので38分が1.06倍の41分になる。

即ち、22:16に完了しているジョブ群は約9分遅れると予測する。9分を短くしたいなら、CPU時間の長い#####HIGAや#####O100のチューニングを検討すればよい。#####HIGAは表3に示した通りCPU頻度が非常に高い(546)ので改善できる可能性が大きい。#####O100のCPU/IO頻度は58なのでCPU時間の削減はむずかしいと言える。

ジョブの処理時間を評価するとき、ジョブの実行待ち時間や装置待ち時間について言及している書物もあるが、GSでは時代遅れの議論である。仮にこのような事象があったとしてもメタボ診断で検出することができるので性能予測では関与しない。

## 4. 3 リスク評価

### (1) 簡易的な検証

システムにリスクが潜在していると、環境が変化したときに表面化する可能性がある。CPU ダウングレード時にはその危険性が高い。リスクを洗い出し評価することが目的であり、CPU ダウングレードが現実的なものなのかを判断する三つ目のハードルとなる。代表的なリスクを以下に示す。

#### ① CPU 頻度が高い

CPU ダウングレードにとって一番のリスクである。逆に、チューニングできれば成功のポイントとなる。業務上重要でない処理は止める決断も大事である。

#### ② レスポンスの悪いオンラインプログラム

2. 3 で説明したように事前に評価すべきである。CPU のダウングレード後にエンドユーザから遅くなったと言われない準備が必要である。処理時間が延びればデータベースの排他待ちも起こりやすくなる。

#### ③ システム上のボトルネック

ハードウェアだけでなく、OS やミドルウェアまで網羅して性能評価をする。仮想記憶量の大きさや増減はシステムの特徴をつかむために効果的である。

### (2) 本検証

上記①～③について原因分析をし、現実的かつ具体的な改善策を立案することが課題である。改善に過度な期待をすることは得策ではない。

①で、CPU 頻度が高いジョブばかりでプログラムのチューニングもできないなら、CPU ダウングレードは諦めた方がよい。

②で、あまりにもサービスレベルが悪く改善の余地がないようであれば、CPU ダウングレードは諦めた方がよい。

③で、システムや運用上の大きな問題は改善すべきである。

仮にダウングレードが実現不可と判断されても、この評価結果をシステムの信頼性やサービスレベルの向上に結びつけることが重要である。

## 5. 評価と課題

### (1) 評価

2008 年前半に GS のメタボ化を論文にまとめ、後半には実践することができた。

お客様の一番の関心は CPU リプレースとコストダウンである。GS メタボ診断は目的ではなく、最適な CPU リプレースとコストダウンを実現するための一つの手段である。この実現のため、以下の4点を実現した。

① CPU/IO 頻度分析の応用（システムからジョブ、プログラムへ）

② CPU リプレース難易度チェックシートの作成

③ メタボ診断と性能予測の連携

④ CPU ダウングレードの実現性評価手法の確立

## (2) 課題

CPU ダウングレードを実現するプロセスの普及.

段階的に CPU をダウングレードできるアウトソーシングがあり, これを実現するためのツールとして活用を推進したい.

以上

## 参考文献

- [1] 有賀光浩, 「メインフレーム変革! お客様視点・現場重視の性能評価 - 3割のコスト削減・性能向上・満足度向上運動-」 FUJITSUファミリ会2006年前期論文
- [2] 有賀光浩, 「メインフレーム変革! 性能品質向上への挑戦 -CPU統合の見える化と改善-」 FUJITSUファミリ会2006年後期論文
- [3] 有賀光浩, 「メインフレームのメタボ化とその診断手法」 FUJITSUファミリ会2008年論文

## 用語説明

GS21, PRIMEFORCE	富士通メインフレームの機種名
NDB	GSのネットワーク型データベース
RDB II, SymfoWARE	GSのリレーショナル型データベース
CPU/IO頻度	CPUとI/Oのバランスを示す指標
IOPS	1秒当たりのI/O回数
CPU占有率	CPU時間/処理時間
TPS	1秒当たりのトランザクション件数
AIM	オンラインシステムを支援するサブシステム
HLF	更新後ログファイル
PRIMEQUEST	富士通の基幹IAサーバ
MSP, XSP	富士通メインフレームのOS
AVM	仮想化を実現する製品
PDL/PDA	性能評価のための性能データ
SMF	課金データ