
メインフレームのパフォーマンス改革 2004

－ お客様が感動したチューニング効果とコスト削減のプロローグ －

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)入社
2003 年 富士通(株)退職
2004 年 (株)アイビスインターナショナル設立
代表取締役

■ 論文要旨 ■

オープン全盛期の現在でも富士通メインフレーム上で 4,000 システムが動作している。CPU 本体のハードウェア費用が下がってきた今、「メインフレームのパフォーマンス改革」を提案する。システムの稼動状況を分析し、適切な管理と改善をすることにより、オープン化よりも簡単にコストを削減し、性能・信頼性の高いシステムの構築が可能となる。本編では事例を中心に、(1)稼動状況分析フェーズにおける重大問題をお客様と富士通に提起。また、(2)お客様が感動したチューニングーリコンパイルのみで夜間バッチの終了が 40 分短縮、システム起動プロシジャの変更で 10 秒以上のオンラインレスポンスが 1 秒未満に改善、プログラム環境定義の修正で一日 7,200 秒の CPU 削減ーといった事例を紹介する。これらの改善作業を行い、次期システム導入に向けて、お客様が主導権を握り、作業・コストの見える化によりコスト削減と品質向上を課題として現在取り組んでいる。

■ 論文目次 ■

1. はじめに	《 3》
1. 1 当社概要	《 3》
1. 2 背景	《 3》
1. 3 メインフレームの性能評価の特殊性	《 4》
2. 仮説のブレイクダウン	《 4》
3. 現状の問題点と解決策（事例）	《 4》
3. 1 システムの稼動状況を分析できない	《 4》
3. 2 適切な管理と改善ができていない	《 5》
3. 3 コストを削減できない	《 9》
4. むすび	《 10》

■ 図表一覧 ■

図 1 夜間 CPU 使用率の推移	《 6》
図 2 オンライン処理時間の改善状況	《 8》
図 3 改善後の CPU 使用状況	《 10》
表 1 10 万件のデータでテストした結果	《 7》
表 2 オンラインの処理時間の推移	《 7》

1. はじめに

1. 1 当社概要

株式会社アイビスインターナショナル（所在地：東京都）は、2004年1月に設立した富士通メインフレーム（GS21, GS シリーズ）の性能コンサルティングを事業とする企業である（ホームページ：<http://www.ibisinc.co.jp/>）。

代表の有賀光浩(Aruga 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年間はシステム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。この間に対応したシステム数は国内外合わせて1,000以上に及ぶ。

弊社は完全に独立した企業として、お客様が富士通メインフレームをより快適に、そして適切なコストで使って頂くためのノウハウを提供している。

1. 2 背景

(1) なぜ今メインフレームなのか？

オープンシステム(*A)が主流であることは事実である。現在、富士通メインフレーム上で4,000システムが動いていることも事実である。富士通は2003年にGS21シリーズのフルラインナップを出荷、2006年に次期GS21投入も計画している[文献(2)]。

メインフレームを利用しているお客様は、日頃からコスト削減、人材確保、オープン化といった課題に取り組んでいる。ITの2007年問題が議論されているが、お客様だけの問題ではなく、メインフレーム技術者不足や性能・信頼性を維持するスキルの低下も大きな問題だと認識している。

(2) パフォーマンス改革 2004 とは

メインフレームの処理能力は向上し、CPU本体のハードウェア費用は徐々に引き下げられている(*B)。半面、高性能のCPUと大容量のメモリをどうやって使えばよいのか悩んでいるお客様も見受けられる。今こそ、システムをブラッシュアップし改革に着手する絶好のタイミングだと考える。

現在のシステムの稼動状況を分析し、適切な管理と改善をすることにより、オープン化よりもコストを削減し、性能・信頼性の高いシステムをお客様主導で構築して頂けると確信している（仮説）。 本論文では事例を中心に紹介したい。

(3) 富士通メインフレームの問題

問題を一言でいうと「メインフレームに関する情報がオープンでないこと」。

例えば、今年(2004年)7月に東京国際フォーラムで盛大に開催された「富士通ソリューションフォーラム2004」。3日間のプログラムのうちメインフレームに関わるセミナーが一つもない(情報発信が不十分)。また、弊社ホームページに一番アクセスの多いドメインはfujitsu.co.jp(8%)であり、単純な機種名(GS8900など)・装置名称(OCLINKなど)での検索が多いこともわかっている(情報共有が不十分?)。

この対処方法は、お客様自身が具体的に欲しい情報をSEまたは営業に示し、欲しい理由を理解させ、情報を提供してもらうしかない。これをお客様にお願いするのは非

常に困難である。

1. 3 メインフレームの性能評価の特殊性

メインフレームでも LINUX でも Windows でも性能評価，チューニングの考え方は基本的に変わらない。逆に，サーバが一つに集約され，OS・ミドルウェアが極めて論理的に動作し，ツールも整備されているためメインフレームの評価の方が容易な面も多いとも考える。OS，データベース等の幅広い専門技術が必要なのはオープンシステムも同様である。

2. 仮説のブレイクダウン

前述の仮説をブレイクダウンし，否定形で書きかえると①～④のようになる。一つひとつ大事な問題であり検証を進めていくが，このすべてを解決することにより，お客様の大きなメリットを見出そうと考えている。現在，①～②は検証済，③は検証中，④は今後の予定である。本論文では①～③について論述する。

- ① システムの稼動状況を分析できない。
- ② 適切な管理と改善ができない。
- ③ コストを削減できない。
- ④ システムをお客様主導で構築できない。

3. 現状の問題点と解決策（事例）

本章では，弊社が実施した性能コンサルティングで，大型メインフレームを使っている2社の事例を中心に紹介する。いずれも10年以上前から使われている基幹系システムである。データベースはNDBとVSAMである。

- ・ A社（流通業） GS8900/10 OS: OSIV/XSP
- ・ B社（製造業） GS8800/40 OS: OSIV/MSP

3. 1 システムの稼動状況を分析できない

システムの稼動状況を分析するための一般的なツールとしてPDL/PDA（*C）がOS標準で提供されている。10年前は普通に使われていたツールだが，システムの大規模化，複雑化に伴い，適切にデータ採取ができるシステムはもはや2割以下だと推測する。

PDL/PDAは会社にたとえると帳簿や財務諸表に相当する重要なデータである。メインフレームをお使いのお客様は是非ともシステム管理者に確認して頂きたい。

- (1) 事例1（重要度★★★）【A社】 ※重要度を最大★3つで表す

多くのお客様で発生している事象であり早急の対応が必要である。

- 現象：PDAで出てこないレポートやSMQN名（*D）がある。（OS共通）
- 問題：PDL採取の際，データロスが発生する。
 - ・ PDAのA3レポートのLOST COUNTSで確認ができる。

[PDA A3 レポートの出力例]

A3. SAMPLING RESOURCE INFORMATION

RESOURCE NAME	SAMPLING RATE		LOST
	V	D	COUNTS
* AMSG	00:10:00:000	00:10:00:000	1,810

- 原因：PDL 実行の際のバッファが小さいためデータロストが発生している。
オンラインプログラム数や DASD 本数が増えたため、バッファを大きく確保しないと PDL のデータ出力が間に合わなくなっている。
- 対処：PDL の出力バッファを拡張する。
XSP は PSQA(*E) 領域を使用するので注意が必要。XSP AF II V03061 からは EPSQA(*F) 領域に獲得するよう改善した。

(2) 事例 2 (重要度★) 【A社】

PDL/PDA を 18 年間使用して初めて見た現象。X0 レポートが出ていれば問題なし。

- 現象・問題：PDA で X0 レポートが出力されないことがある。(XSP のみ)
- 原因：調査依頼の結果、PDA 実行時リージョン(*G)不足が原因と判明。
エラーメッセージは何も出力されないため使用法誤解ではなく障害である。
- 対処：PDA 実行時に RSIZE=1024K 以上を指定する。

(3) 事例 3 (重要度★★★) 【B社】

AVM/EX(*H) を使用しているお客様では至急確認が必要。

- 現象：CPU 使用率レポート (B1 レポート) は普通に出力される。(MSP のみ)
- 問題：AVM/EX 配下で PDL (CPU データ) を採取すると CPU 使用率の精度が極端に悪化しているにもかかわらず、ユーザはその値を信じてしまう。
(例) 実 CPU の CPU 使用率
・ B1 レポート 95% → 実際は 62%
- 原因：AVM/EX 配下の PDL CPU 使用率レポート (B1 レポート) の値が正しくないのは 10 年以上前から仕様通り (対処方法あり)。しかし、マニュアルからは数値が全く役に立たないようには読み取れず誤解される可能性が大きい。
- 対処：PDL Z1 レポートを使用する。
または
PDL のオプション 1 機能+ファームウェア PA 機能を使用する。
・ PDL の EXEC 文 `PARM=' OPT1, PA=YES'`

3. 2 適切な管理と改善ができていない

改善ができない理由は 2 点考えられる。

① 問題を認識できていない

→ 現状を把握できていない or 目標値 (管理指標値) がない
or 異常値に気づいていない

●目標値 (管理指標値) とは、

・ 随時見直し、変わっていく (コントロールすることが管理である)

- ・お客様のシステムごとに異なる
「CPU 使用率の限界値は 70%」も「CPU を 90%以上使用する」もありうる

●異常値とは、

- ・自システム内の異常値を見つける
- ・他システムと比べての異常値を見つける
→ お客様間のネットワークが重要、ファミリー会など有効に活用すべき

② 効果的な改善策が実施できていない

→ 内部要因 or 外部要因

例えばチューニング事例 1 については、現状を把握できていなかった内部要因と、外部委託したプログラムであったという外部要因が重なっていた。

(1) チューニング事例 1 (重要度★★★)【A社】

■成果：夜間バッチの CPU 時間が 1 時間以上削減 (システム全体の 30%相当)
夜間バッチ終了時間が 40 分以上短縮

■現象：夜間に 1 時間以上 CPU 使用率が 100%となる (図 1)

注) 棒グラフ VM1 (仮想システム 1) と VM2 (仮想システム 2) が CPU 使用率, 折れ線グラフは 1 秒間の IO 回数.

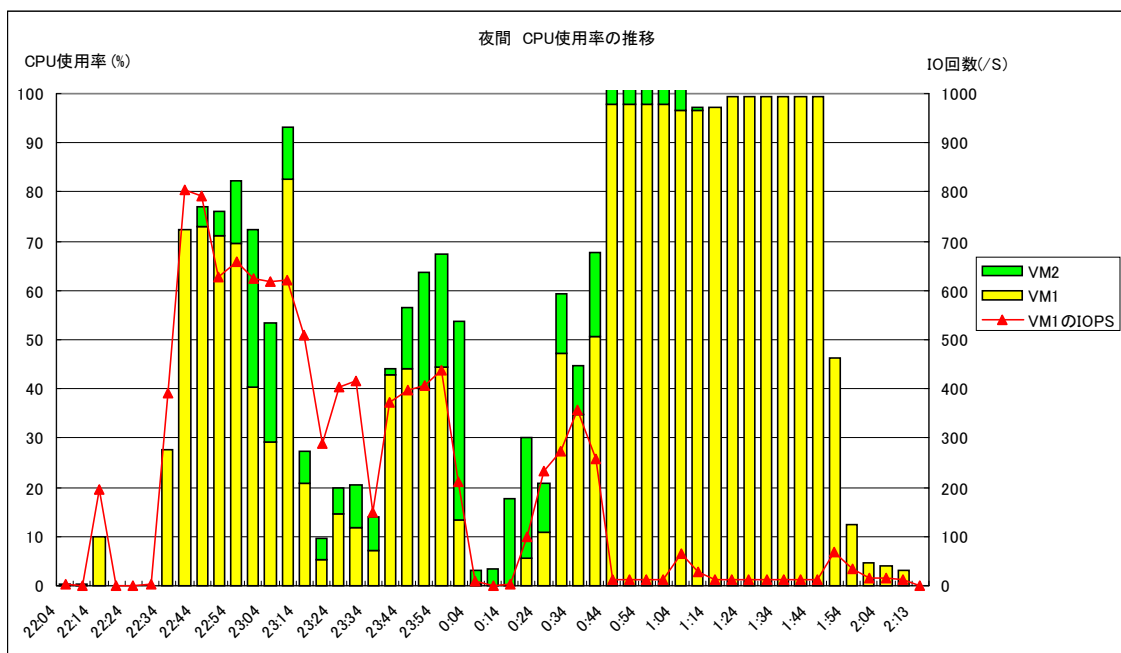


図 1 夜間 CPU 使用率の推移

■問題：CPU を使うだけ (棒グラフ) で I/O がほとんど出ていない (折れ線グラフ).
システム管理者は、何の業務プログラムなのか正確に把握していなかった.

■原因：UNIX サーバにデータ転送するためのコード変換を行っていた.

→ コード変換はメインフレームでやりたくない処理の一つである.

(高価な CPU を使って PC でも簡単にできる処理をやってはいけない)

更に、ロードモジュールに CHECK オプション(*I)が付加されていた。

■対処：COBOL コンパイルの CHECK オプションをはずす。(リコンパイル実行)

テストの結果を表 1 に示す。今回のプログラムは、テーブルサーチが多かったため大幅に削減される効果が出た。外部委託したプログラムであり、デバッグオプションの確認までできていなかった様子。

表 1 10 万件のデータでテストした結果

	CHECK オプション付	CHECK オプション無
処理時間	10 分 03 秒	1 分 21 秒 (▲86%)
CPU 時間	8 分 08 秒	1 分 01 秒 (▲87%)

なお、デバッグオプションの確認は COBOL 標準ライブラリでも提供されている。

(2) チューニング事例 2 (重要度★) 【A社】

■成果：10 秒以上の一時的なレスポンス悪化が 1 秒未満に改善
オンラインジョブ全体の CPU 使用時間が 35%削減

■現象：オンラインレスポンスが一時的に 10 秒以上となる (表 2)

表 2 オンラインの処理時間の推移～7/04 と 7/28 は改善前，7/29 は改善後 単位：秒

	PROG000	PROG001	PROG041	PROG210
7/04 レスポンス(最大)	12.631	20.649	39.912	14.849
7/28 レスポンス(最大)	6.850	18.596	10.929	13.704
7/29 レスポンス(最大)	0.004	0.122	0.853	0.007
7/28 と 7/29 の削減率	▲99.9%	▲99.3%	▲92.2%	▲99.9%
7/04 レスポンス(平均)	0.030	0.053	0.160	0.036
7/28 レスポンス(平均)	0.009	0.019	0.087	0.010
7/29 レスポンス(平均)	0.000	0.006	0.049	0.000
7/28 と 7/29 の削減率	▲100.0%	▲68.4%	▲43.7%	▲100.0%

■問題：最も重要かつレスポンスを要求するオンライン処理であり，業務上影響が出ている可能性がある。

■原因：AIM-SNAP(*J)が取得されていた。

SNAP ファイルとバッチジョブのワークが同一ボリューム上にあり，一日に 3 回バッチジョブが実行されると I/O 競合によりレスポンスが悪化した。

■対処：AIM-SNAP をはずす。(システム起動プロシジャの変更)

システム起動プロシジャ内に AIM-SNAP 取得指令が入っていた。

お客様は AIM-SNAP は日常的に取得してはいけないことを知っていたが，外部にまかせていたシステム起動プロシジャの内容精査が漏れてしまっていた。

(3) チューニング事例3 (重要度★★) 【B社】

■効果：オンラインジョブのCPU時間が一日7,200秒削減
 オンライン処理時間が80%以上短縮

■現象：オンライン処理時間の遅延(図2)によりメッセージが多量に滞留する。
 バックグラウンドで動作するためシビアなレスポンスは要求されていない。

注) オンライン処理時間 = CPU時間 + CPU以外の時間 + 排他待ち時間
 CPU以外の時間とは、CPU待ち時間、I/O時間など

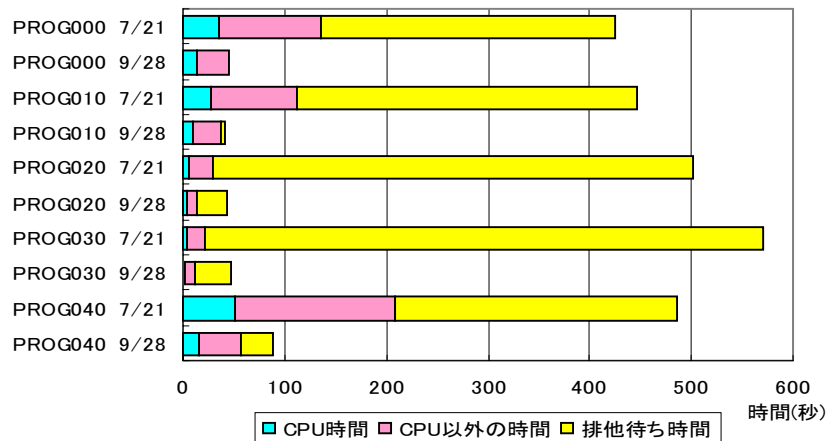


図2 オンライン処理時間の改善状況 7/21→9/28

■問題：滞留したメッセージが夜までに処理しきれず業務に影響を与えた。
 プログラムの構造上、滞留したメッセージはデータのカラ読みなども発生し
 資源を無駄に使用する。

- 原因：①AIMの排他制御(*1)によるCPU時間の増加。
 ②CPUネックによるCPU待ち時間の増加。
 ③処理時間の遅延による排他待ち時間の増加。

■対処：①について

- ・特定ファイルのアクセスをREAD-ONLY(排他をかけないREAD)に改良。
- ・特定ファイルの「ページ単位の排他」を「データセット単位の排他」に変更。

以上、プログラム環境定義(PED)を変更

⇒ 1トランザクション当たりのCPU時間が約50%削減した。

②について

- ・CPU使用量が多く、かつCPUバウンドジョブであるが、CPUプライオリティを通常のオンラインよりも上げた。

⇒ 4CPU構成のため優先度が低いジョブへの影響は小さいと判断。CPU待ち時間が50%以上削減、他ジョブへの影響も見られなかった。

③について

- ・①、②の効果で排他待ち時間が10%以下に減少した。

■*1の解説

AIM/VSAMにAIMが排他をかけるときENQ/DEQ(*K)マクロを発行している(GTFト

レースの解析から判明)。VSAM を全件サーチする処理だったため、このオーバーヘッドが VSAM アクセスマクロよりも大きくなってしまった。私自身初めてのチューニングであり、マニュアルや資料に記載されている内容ではない。

3.3 コストを削減できない

(1) 仮説設定

一般的に「メインフレームは高い」と言われている。この根本原因は以下の2つだと仮説をたてた。

- ① 必要以上に大きいマシンの提案を受け、それを導入する。
- ② CPU 使用率が低いにもかかわらず維持費の高いマシンを使い続けている。

①は検証途中だが現時点での結果を(3)で紹介し、成果は次回の論文で紹介したい。

②は今後の案件である。再リースで GS8600, GS8800 をお使いで、コスト削減を図りたいと強く望んでいるお客様とともに是非とも検証を行いたい。

(2) コスト削減の方向性

GS21 のハードウェアは、富士通の PRESS RELEASE に「GS21 600 モデルグループは、従来機種 (GS8600) と比較し価格性能で最大約 3.3 倍の向上を実現」との記載があるように価格性能比は徐々に向上してきているらしい (未検証)。

これが事実と仮定すると、コスト削減のポイントはソフトウェア費用と保守費用となる。これらの費用は CPU の能力に依存するため、ハードウェアが安くなったからといって大きな CPU は不用意に導入すべきでない。

(3) A社での事例 (重要度★★★)

次期 GS21 の選定作業中。一番重要な作業プロセス。

GS8900/10 を使っている A 社では、日中の平均 CPU 使用率は 30% 前後、バッチジョブが流れると 40% 前後になる。夜間は 3.2 事例(1) **図 1** に示した通り CPU が 100% に達する時間帯が 1 時間以上あった。

●仮説①の否定

メーカーからの 1 回目の提案では、同等性能の CPU を提示していた (詳細は性能検証をして決定すること)。これは仮説①を否定した結果であった。

●同等性能の CPU への移行リスク

性能トラブルを起こす代表的なケースの一つは、同じ性能の CPU に単純移行したことである。

詳細説明は割愛するが、以下は基本知識として理解して頂きたい内容である。

- ・ 2 割の処理が速くなり、2 割が遅くなる。6 割はほぼ同等。この遅くなる 2 割をどう救うかがポイントである。
- ・ DASD 性能が向上するため、速くなりすぎる処理が発生し、逆に遅くなる処理も発生する。
- ・ 今回チューニング事例 1 で紹介した処理は、更に遅くなっていた可能性が高い。

●リスクヘッジ

- ・チューニングによりボトルネックを解消する。遅くなる分の改善を施しておく。
- ・何かあったときのために、チューニング項目を用意しておく。

前章で紹介した改善作業等を進めた結果、CPU の使用状況は**図3**に示すようになり、オンライン性能、バッチ処理時間も現状問題ない。現在の70%以下のCPU能力でも問題なく運用できる準備をしている。本体ハードウェアだけでも数千万のコスト削減となる。

移行後のシステム性能については、SLA で明確に取り決めをすべきである。

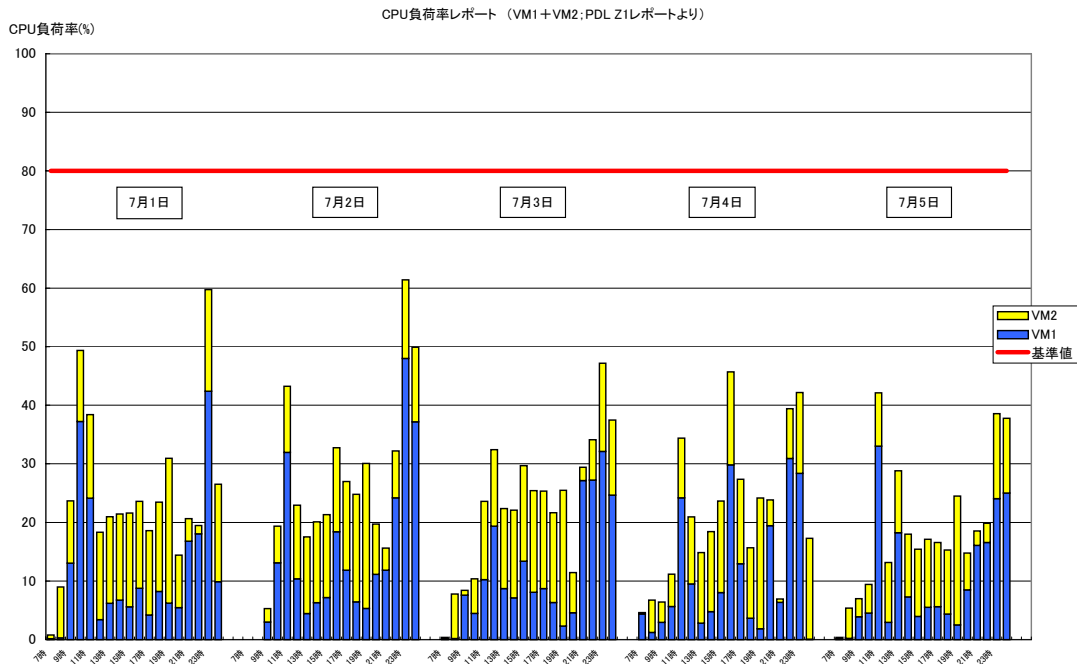


図3 改善後のCPU使用状況

(4) 今後の課題

単に小さい CPU を導入することが目的ではない。今後の課題として以下の内容がある。

- ・お客様が主導権を握り、富士通と協力し、論理的に最適なCPUを選定する
- ・移行を含めた作業・コストの見える化と作業品質の向上
- ・無駄、不足のないメインフレームシステムの構成設計
メモリ、チャンネル、DASD、SSU (システム記憶)、ソフトウェア 等

4. むすび

本論文では以下の4つの仮説を設定した。

- ① システムの稼動状況を分析できない。
- ② 適切な改善と管理ができない。
- ③ コストを削減できない。
- ④ 将来システムをお客様主導でデザインできない。

以下に簡単に整理する。

・①について

特に事例1（データロスト）、事例3（AVM/EX 配下の CPU 使用率）で紹介した内容は、すべてのお客様で検証して頂きたい。また、富士通としてもフィードバックして欲しい内容である。

必要なときに正確なデータを取れる仕組みを作っておくことが重要である。PDL/PDA はお客様のためのツールである。

・②について

チューニング事例はあくまで事例ではあるが、事例1（デバッグオプションの削除）、事例2（AIM-SNAP 取得の停止）、事例3（AIM 排他制御の見直し）で CPU 時間の大幅な削減が実現できた。修正作業はすべて簡単なものである。

特に外部委託したプログラムや作業には、システム運用部門は目が向きにくい傾向にある。第三者（当然社内でもよい）による客観的な評価・問題指摘・改善や品質向上の仕掛けが重要である。

・③, ④について

A社における新機種導入にあたってのパフォーマンス改革は高いレベルで完了した。今後は、お客様間の視点で、特にコスト削減と高い品質向上を意識して作業を進めていくことが重要である。

CPU が1ランク下がるとハード・ソフト・保守費用がどれだけ下がるのか、お客様一人ひとりが改めて認識をする必要がある。

以上論述した通り、適切な管理・改善と、作業・コストの見える化をお客様が主導権を握って推進することにより、メインフレームでも品質向上を図りながら IT コストを 30%以上削減することが可能である。

メインフレームをよりよく使って頂くために、今後の課題は以下のように考えている。

- お客様間での情報共有、人材共有
- 再リースでメインフレームをお使いのお客様のコスト削減

また、富士通株式会社様には、お客様がいつまでメインフレームシステムを安心して使えるのか是非ともコミットメントして頂けるようお願いしたい。

以上

用語説明, 解説

A オープンシステム :

10年以上前から使われている言葉だが明確な定義がされていない。本論文では影響が少ないため定義をせず使用する。

B ソフトウェア費用, 保守費用はほとんど下がっていない。メインフレームはこれら固定費が高いと指摘されている。IBMはソフトウェア費用に変動的な考えを取込んでいる。

C PDL/PDA Performance Data Logger / Performance Data Analyzer

性能評価ツール。PDLがデータ収集, PDAがその編集。

UNIXのsarコマンド, Windowsのパフォーマンスモニターに相当。

D SMQN Secondary Message Queue Node

オンラインプログラムのこと。

E PSQA Pageable System Queue Area

仮想空間でシステムの制御表が配置される領域。アドレス16MB以下の基本領域でページング可能。この領域が枯渇するとシステムダウンする可能性がある。

F EPSQA Extended Pageable System Queue Area

仮想空間でシステムの制御表が配置される領域。アドレス16MB以上の拡張領域でページングが可能。

G リージョン

仮想空間でユーザプログラムが配置される領域。REGION。

H AVM/EX

VM(仮想マシン)を作成するためのソフトウェア。

I CHECK オプション

COBOLの機能。ポインタがプログラムのテーブルの外を指していないかチェックする。

J AIM-SNAP

AIM(オンラインを制御するミドルウェア)がDC(通信)とDB(データベース)にアクセスしたデバッグ用ログを取得する機能。

K ENQ/DEQ

OS資源のロックをかけるときENQマクロを発行し, 解くときDEQマクロを発行する。

参考文献

(1) 江川博之氏ほか, 「グローバルサーバ:GS21」雑誌FUJITSU 2002年11月号

(2) 経営執行役山中明氏, 「富士通のサーバ戦略」2004年7月22日

<http://img.jp.fujitsu.com/prir/jp/ir/materials/20040722/pdf/2.pdf>