

カレンダーに基づいた運用周期からみるサーバ群の平常負荷把握と予測

株式会社アイ・アイ・エム 技術本部
塩川孝雄、國本賢一、松本征大、三保谷聡、荒居真司

サーバ・ネットワークシステムが大規模・複雑化するなかで、安全運用の確保とIT投資を的確に行うには、重要サーバだけを把握するのではなく、全サーバの平常負荷（平熱）を把握した予測管理に重点をおくことが肝要である。

サーバの負荷は複雑な変動を示すので、平常負荷の把握は非常に困難に思えるが、実態はカレンダーに基づいた実業務の運用周期に連動している。

今回、その平常負荷と運用周期の関係に注目し、過去から未来にわたり日付を特定できる週、月、年などの複数の周期モデルを用い、平常負荷の把握と予測を支援する運用SEツールを開発した。本稿は、平常負荷の把握と予測の手法を述べ、大規模サーバ群の予測管理における実用性を事例から検証する。

Grasp and forecast of usual load on a server, using cycle of operation based on calendar.

Takao Shiokawa, Kenichi kunimoto, Yukio Matsumoto, Akira Mihoya, Shinji Arai
Engineering Headquarter, IIM Corporation

The Server network system became larger and more complexed. To operate the system safely and to invest in it accurately, it's vital to use the forecast management based on grasp of usual load(normal temperature) for whole servers, not the important servers only.

Though it seems difficult to grasp usual load on each servers because it fluctuates intricately, their behavior interlocks with the operation cycle of practical business based on a calendar.

We developed new tool for operation SEs to support grasping and forecasting the usual load on a server. We focused on the relation between the usual load on a server and the cycle of its practical business, and used several types of cycle models (for example weekly, monthly and yearly) in the tool. In this paper, we show the way of grasping and forecasting the usual load, then test its usefulness in forecast management on large-scale server cluster.

1. はじめに

サーバにおける平常負荷とは、「長期の増減傾向」と「複数の周期」の複合化したものに、「散発的なバラつき」が付加されたものである。

一般に予測は当たらないとされるが、サーバ負荷の予測管理は欠かすことができない。汎用的な予測計算では、時系列データの複雑な計算になりがちで、特に多種多様な多数のサーバを抱えるユーザでは、計算式の複雑さや処理時間により実用性が確保できない。

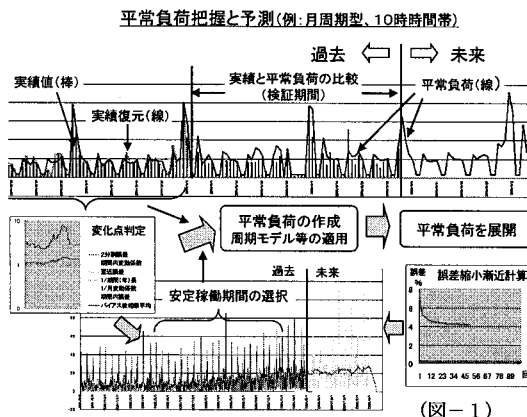
この解決方法として、運用経験則から考えられる「カレンダーに従って日付を特定できる複数の汎用的な運用周期モデル」を作成することとした。

例えば、月曜日という週周期モデルの要素の平均値を実績値により計算し、過去・未来の全日付において他のモデルの要素と併せて総合平均値として展開する。

近似線と適切な周期モデルを適用して、「誤差縮小の漸近計算」を行うことで、概ね全ての周期を網羅した実

用的な平常負荷を模擬することが可能となり、また未来方向に自動的に展開できることを確認した。

図-1は週周期と月ピークが顕著な事例である。



この機能を運用SEツールに搭載する際、サーバ台数が多い場合の実用性を確保するため、「安定稼働期間の自動判別」と「立体的な鳥瞰表示」の機能を付加して、短時間で負荷全体の分散状態と変化を視覚的に把握できるよう工夫した。

なお、本稿で述べる全ての時系列データは、1時間間隔で採取した3年間以内の稼働実績(CPU使用率)と1年以内の予測データである。

2. サーバの平常負荷

最初に、人間の平熱(体温)に相当するサーバの平常負荷の実態がどのようなものか述べることにする。

システム管理者は平常負荷を把握する場合、当該日と同一条件の過去の日を選んで平均値を算出する。例えば、当該月曜日の負荷が他の週の月曜日より高い傾向にある場合、複数の月曜日の負荷を採取して、その平均値をもって月曜日の負荷と考える。しかし、毎月の第一営業日に月次処理を行い、その日が偶然に月曜に当たっている場合は、高めの値を平均値と見なしてしまう恐れがある。これでは、月曜日としての平常負荷とはいえない。このように、単独の周期要素だけによる平均値を平常負荷とすることはできない。実際には、曜日他に1日の時間推移による変化、月や四半期処理、または年次処理といった長短の周期による複雑な変動要素があることから、平常負荷は複合したものになる。

一方、サーバ負荷は周期以外にも多くの要因が内在して変動している。ここで、運用経験則に基づき次の様に負荷を分解してみる。

- ① 自然増減する負荷(継続)
- ② 周期負荷(継続)
- ③ 散発的な負荷(継続)
- ④ 一時的な変動負荷(非継続)
- ⑤ 急な増減負荷(変化後継続)

まず、平常負荷は、安定稼働期間の実績値より把握でき、未来も継続する負荷であるものとし、①の自然増減と②の周期的負荷の複合と考える。

$$\text{平常負荷} = \text{①自然増減負荷} + \text{②周期負荷}$$

①の自然増減する負荷は、一般に最小二乗法による近似線計算により模擬できる。②の周期負荷については、次の3項で詳細を述べる。

③の散発的に発生する負荷は、安定稼働状態の負荷推移から①と②を除いた残差であり、独立同一分布の白色ノイズによる負荷と考えられる。本来は平常負荷の一部とみるべきものである。この散発的負荷を平常負荷の一部として3項で後述する周期負荷のようにモデル化が

できれば、定常負荷の把握がよりの確になると思われる。

しかし、実際には発生原因が多様で、④の一時的な変動負荷との見極めが難しい。このため、③の散発的負荷は本稿では平常負荷として取り扱わず、「実績値」と「①自然増減負荷+②周期負荷」との誤差として扱う。

ここでは、負荷のバラつき具合を可視化する程度(5項で後述)に留め、今後の課題とする。

④の一時的な変動負荷は、臨時処理や障害などによるものであり平常負荷ではない。分析に当たって事前に除去することが望ましいが、本稿では上記③の散発的な負荷の一部と考える。

⑤の急な増減負荷は、新規の運用開始や設備増強、プログラムの変更、または計画的な処理件数の拡大・縮小などによる不連続なもので無視することは出来ない。平常負荷の把握では、この不連続な変化点を検出して、安定稼働期間を選定することが大きな課題となる。このため、変化点の検出方法について5項の運用SEツールの項で触れる。

3. 周期モデルの定義

さて、サーバ負荷の周期を捕捉することが出来る汎用的な周期モデルとはどのようなものであろうか。次に、業務用サーバの負荷の周期的な変化要因を列挙してみる。

- ・1日の時間推移による負荷の変動
- ・異なる曜日による負荷の変化
- ・休祭日と平日間の負荷の違い
- ・月初め、中旬、月末等の処理による負荷の変動
- ・2ヵ月毎の処理による負荷の変化
- ・四半期毎の処理による負荷の変化
- ・半期毎の処理による負荷の変化
- ・年度毎の処理による負荷の変化
- ・年間を通した季節変動の変動等

この他、ユーザ独自処理による周期負荷が存在する。

例えば、曜日毎に平均値が異なるケースでは、曜日を要素とした周期モデルが考えられる。各曜日はカレンダーに基づき日付をグループ化でき、全日付を網羅することができる。月周期では、曜日に相当するものが無いが、曜日の代わりに、例えば営業日を番号付けする等により1ヵ月間内の全日付を分類することができる。

以上の運用上の経験則からくるヒントを基に、「周期モデル」の成立基本条件を次のように考えた。

周期モデルが成立する基本条件
カレンダーの全日を網羅し、日付を特定できるユニークな複数の要素を有する。

また、周期モデルの機能目的を次のように考える。

周期モデルの機能目的

周期モデルの各要素で特定した日付の実績値の平均を求め、過去から未来の全該当日にその平均値を展開する。

このような日付を自動的に特定でき、機能目的を持たせた汎用的な周期モデルの例を次に列挙する。なお、この周期モデルは、5項の運用SEツールの開発でそのまま適用する。

① 週周期モデル

1週間の曜日毎による平均的な負荷変化を表す。
月曜の負荷は高く、土日の負荷は極端に低いなど、曜日による変化が想定される。
→月曜から日曜日までの各曜日を要素とする。

② 月周期モデル

1ヵ月内の平均的な負荷変化を表す。
第一営業日、第二営業日などの月初、および月末処理、中旬のピーク負荷日などの月内負荷変動が想定される。

→1ヵ月内を分類してユニークにカレンダー上の日付を特定できるよう、平日（土・日、祭日などの休みを除く営業日）を分類する。

- ・月初めから数え、その順番毎に要素とする。
- ・月末から日を数え、その順番毎に要素とする。
- ・月の中日頃の数日（例：5日間）のグループを1つの要素とする。

③ 2ヵ月周期モデル

2ヵ月おきの処理を想定して、2ヵ月内の平均的な負荷変化を表す。
→奇数から偶数月の2ヵ月間を1つのグループとして、月周期モデルと同様に、営業日の初めと終わりから番号をふることで要素をつくる。

④ 四半期周期モデル

四半期毎の処理を想定して、四半期期間内の平均的な変化を表す。
→四半期を1つのグループとして、月周期モデルと同様に、営業日の初めと終わりから番号をふることで要素をつくる。

⑤ 半年周期モデル

→四半期周期モデルと同様の方法で、半年周期の処理を想定して要素をつくる。

⑥ 年周期モデルA（年ピーク）

→四半期周期モデルと同様の方法で、年周期の処理を想定して要素をつくる。

⑦ 年周期モデルB（季節変動）

年間を通して季節的に使用量が変化することを想

定し、年の緩やかな平均負荷推移を表す。

→年度の始めから1年の終わりまでを1週間（7日）毎に区切り、番号をふり、それぞれの番号を要素とする。

以上の各モデルに加え、実際は休日（土・日曜日以外）のような特例日がある。これは、例えば休日と平日による一つのモデルを設定することができる。しかし、本稿では、各モデル内の一要素として取り扱う方が実態に即していると考え、独立にモデルを作成しないこととした。

このため、上記の各周期モデルには、土日および祭日などの休日を1つの要素として設定することにした。

また、ユーザ独自の特別な日がある場合、その特別日も1つの要素と考えられる。このことは、前記の周期モデルの成立条件（各要素は日付を特定）を満足すれば、必ずしも周期的でなくともモデル作成ができることを意味している。

4. 平常負荷の算出

(1) 計算ルール

3項の周期モデルの定義に則り、実績値に各モデルを総合的に適用して平常負荷を導く計算方法を各種試行した。

その結果、一定の計算ルールに従って行われる計算方法であれば、計算手順が異なっても、計算途中での数値の違いはあるが、最終結果はほぼ同じになることを確認した。

モデル適用の基本的計算ルール

- ① 入力データ期間が各モデルとも同じである。
- ② 入力データは出力データから復元できる。
- ③ 周期モデルは、全体傾向の除去（変換）後に適用する。

（「ほぼ」の意味：誤差縮小による漸近計算の回数、およびモデル適用順の違いのため最終確認はできないが、実用的には誤差が小さくなり問題とならない）

(2) 計算の方式

実際の計算においては、全体の増減傾向（近似線）もモデルの一つと考えることができるので、以降の説明では傾向モデル（R）として記述する。

全周期モデル（S）は、個別の周期モデル（ s_1 , s_2 , s_3 ... s_n ）の相乗平均と考えられる。

また、バラつきを総合残差（N）とすることで、実績値（T）は次のような簡単な式で表現できる。

総合モデル(M)＝傾向モデル(R)×全周期モデル(S)
実績値(T)＝総合モデル(M)×総合残差(N)

ここで、以下に説明する計算プロセス(本項(3)に後述)を要約して先に提示しておく。

$T = M \cdot N = R \cdot S \cdot N$ (全て時系列データ)

T: 実績値

M: 総合モデル(定常負荷)

R: 全体傾向モデル

全体の傾向(直線・指数近似)

S: 全周期モデル

年・半年・四半期・月・週などの全周期

N: 総合残差(ノイズ:実績との誤差)

とおき、次を操作する。

N → 縮小 (漸近繰返し計算)

M → カレンダーに沿って展開

(過去⇄現在⇄未来)

周期モデルを適用するには、計算ルールの③の通り事前に全体傾向の要素を除去(データ変換)しておく必要がある。これは、全体傾向による負荷の増減が周期要素の算出に影響を与えないようにするためである。

この除去方法には次の2つが考えられる。

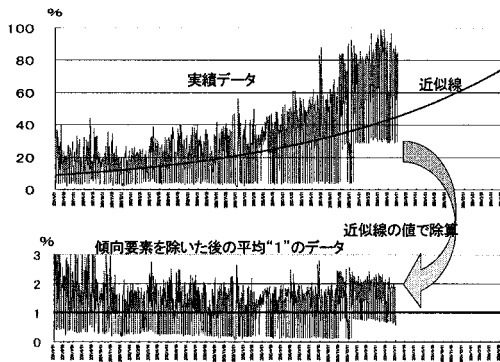
① 乗除計算中心

最小二乗法により作成した近似線の値で対象データ値を除算し、“1”を平均とする時系列データへ変換し、乗除計算を中心に計算する。

② 加減計算中心

対象データ値の対数を取り、“0”を平均とする時系列データへ変換し加減算を中心に計算する。

図-2は、乗除計算中心での計算において指数近似線を用いて平均“1”のデータへ変換する例である。



(図-2)

また、各周期モデルをどのような順番で適用するかにより、次の2つ方法が考えられる。

① 直列的計算

全体傾向を除去後、各周期モデルを順次に適用して計算する。

② 並列的計算

全体傾向を除去後、同じ時系列データを一齐に各周期モデルに適用して計算する。

なお、上記①の直列的計算では、周期モデルの適用順が影響し、漸近計算途中における残差の値が変化する。また、並列的計算では、全ての周期モデルを一齐に適用させるので、必要な漸近計算の回数はやや多いが安定している。

上記の方式を組み合わせた次の4つの計算方式によりプログラミングを行い、実データにより試行した。

① 乗除直列的計算方式

② 加減直列的計算方式

③ 乗除並列的計算方式

④ 加減並列的計算方式

この結果、扱うデータを全て“1”にすることで該当モデルを容易に適用外にでき、また計算途中の値が各モデルの実行順に影響されない③の乗除並列的計算方式を運用SEツールで採用することとした。

以降は③の乗除並列的計算方式を例に述べる。

(3) 計算のプロセス

実績値(T)を用いて総合モデル(M)とその総合残差(N)の作成、および逆変換による実績値を復元するまでの一連のプロセスの概要を列挙する。

なお、以下に記す「モデル」は、該当する時系列データそのものを指す場合にも用いることとする。

① 安定稼働期間の判定

(変化点の判定方法は5項で後述)

② 実績値の近似線作成

全体傾向モデルとして近似線を作成する。

(直線・指数近似の自動選択)

③ 傾向モデルによる変換処理

実績値を近似線の該当値で除算する。

* 傾向残差 = 実績値 / 全体傾向モデル

→ 平均“1”の時系列データ

④ 各周期モデルによる平均値の展開

上記③のデータを使用し各要素の平均値を算出する。この際、モデルの各要素が特定する未来期間の日付へも展開する。(→予測に利用)

⑤ 暫定モデルの算出

上記④の各平均値の相乗平均を算出する。

* 暫定モデル = 全体傾向モデル × (週周期モデル × 月周期モデル × 二ヵ月周期モデル × 四半期周期モデル × 半年周期モデル × 年周期モデルA × 年周期モデルB)^(1/7)

⑥ 総合モデルの累積

* 総合モデル = 総合モデル(1回目は“1”) × 暫定モデル

⑦ 総合残差の算出

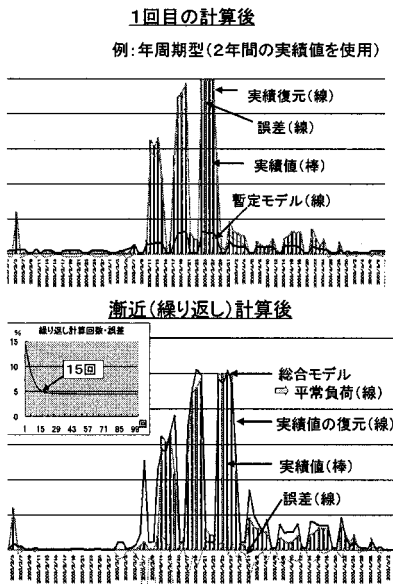
* 総合残差 = (週周期残差 × 月周期残差 × 二ヵ月周期残差 × 四半期周期残差 × 半年周期残差 × 年周期A残差 × 年周期B残差)^(1/n)

⑧ 誤差縮小漸近計算

上記⑦の総合残差を実績値に置き換え、②から⑦の操作繰り返す(10~50回程度)。但し、前回誤差より拡大した場合は打切る。

→実績値と総合(暫定)モデル間の誤差が漸次減少する。

図-3にピーク負荷日での誤差減少の例を示す。



⑨ 平常負荷、実績値の復元計算

⑧までに計算した結果から総合モデルを平常負荷として取り扱う。* 平常負荷=総合モデル
実績値は次の復元値と等しいことを確認する。

* 復元値(T) = 総合モデル × 総合残差

(4) 計算式

T=M・N=R・S・Nから、傾向モデル(R:近似線)により実績値(T)を割り“1”を平均とする時系列データを得て、更にS(周期モデル)で割り、N(残差)を得る計算式を以下に示す。

最初に、R=r₁, S=s₁, N=n₁と置き、他の周期モデル(n個)も次の(i)の一連の計算式のとおり展開して、暫定モデル(M₁)および残差(N₁)の相乗平

均を得る。

(i)

$$T = r_1 s_1 n_1$$

$$T = r_1 s_2 n_2$$

.

$$T = r_1 s_n n_n$$

$$T^n = r_1^n \prod_{j=1}^n s_j n_j$$

$$T = r_1 \left(\prod_{j=1}^n s_j n_j \right)^{1/n}$$

$$T = r_1 \underbrace{\left(\prod_{j=1}^n s_j \right)^{1/n}}_{M_1} \underbrace{\left(\prod_{j=1}^n n_j \right)^{1/n}}_{N_1}$$

ここで、上記操作を1回目とし、

$$M_1 = r_1 \left(\prod_{j=1}^n s_{j1} \right)^{1/n}, N_1 = \left(\prod_{j=1}^n n_{j1} \right)^{1/n}$$

と添え字を追加し、残差部N₁に対し上記の(i)と同様の操作を次の(ii)の通りm回繰り返す。

(ii)

$$T = M_1 \cdot N_1$$

$$T = M_1 \cdot M_2 \cdot N_2$$

.

$$T = M_1 \cdot M_2 \cdot \dots \cdot M_m \cdot N_m$$

$$T = \left(\prod_{k=1}^m M_k \right) \cdot N_m$$

$$T = \prod_{k=1}^m \left[r_k \left(\prod_{j=1}^n s_{jk} \right)^{1/n} \right] \cdot \left(\prod_{j=1}^n n_{jm} \right)^{1/n}$$

M N

以上により、総合モデル(M:平常負荷)、総合残差(N:ノイズ)を得る。

(iii)

$$M = M = \prod_{k=1}^m \left[r_k \left(\prod_{j=1}^n s_{jk} \right)^{1/n} \right]$$

$$N = \left(\prod_{j=1}^n n_{jm} \right)^{1/n}$$

$$T = M \cdot N$$

これにより、実績値(T)は復元できる。

5. 運用SEツールとして開発

システム運用で使用するソフトウェアは、コンピュータ室でシステムの制御や監視を行う運用管理システムと、事務室でシステム運用を間接的に支援する管理、分析、統計などの運用支援ツールに大きく分けられる。

今回は、後者の運用支援ツールとして、運用SEが蓄積データからサーバ全体の負荷を短時間に把握することを目的として開発した。

新機能の開発に当たり、筆者が「大規模サーバ群の負荷予測システムの構築に向けて：2006-EVA-17」において Visual Basic.Net (Microsoft の商標) で試験開発した機能を有効活用した。

運用SEツールとしての運用要件は次の通りである。

- ① 入出力はCSV形式の日付・時刻付の時系列データ
 - ・実績値の最長期間は3年以内
 - ・予測出力期間の最長期間は1年以内
 - ・全24時間の1時間間隔のデータ
- ② 多数のサーバ(数百台程度)を対象
 - ・市販のパソコン程度の性能で実用
- ③ 可視化機能を有する

また、サーバの多種多様な負荷パターンに対応できるよう、これまでに述べた機能を含め、次を開発することとした。

- ① 安定稼働期間の選定
 - ・急に増減する負荷の変化点の判定
- ② 平常負荷の把握と予測展開
 - ・前述した計算方式を搭載
- ③ 負荷分散の可視化
 - ・平常負荷からのバラツキを未来へ2σ幅で表示
- ④ 鳥瞰できる立体的な可視化
 - ・過去未来の全時間帯を立体表示

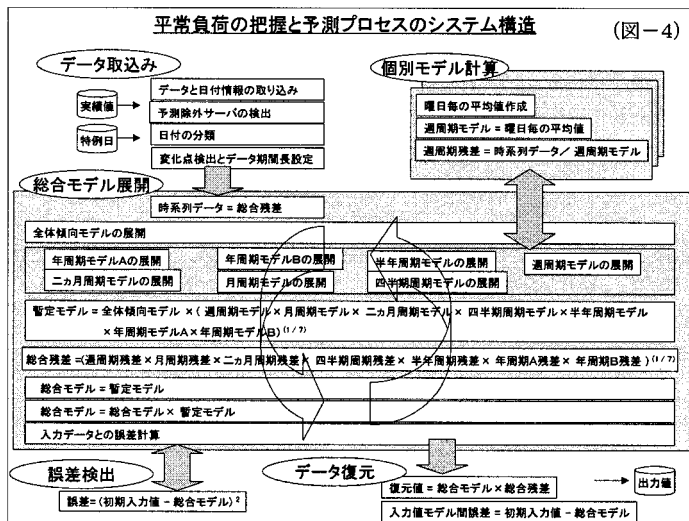
その他、出力結果を評価コメントする機能やサーバ間の横並び比較による優先選択機能などがあるが、一般的機能として本稿では割愛する。

以下に上記の①から④の各機能のシステム構造の概要と留意点を述べる。

(1) 安定稼働期間(変化点)の選定

既に2項で述べたように、平常負荷の把握では、過去の不連続な変化点を検出して安定稼働期間を選定することが、大きな課題である。

与えられた時系列データから2項で提示した⑤の急



な増減負荷(変化後継続)を判断して変化点を決定し、平常負荷を算出するのに使用する期間を選択する。

これには、実績値全体との誤差が最小になる期間、至近での誤差が最小になる期間等を幾つか試行した。

この結果、前述の「2006-EVA-17」で提案した「負荷推移を前後半で2分割して相似性を比較する方法」が単純で最も汎用的と思われる、機能を付加した。実際は、負荷変化点の月平均の変動係数が高いこと、および期間を長くをとることを加味して判断することとした。

(2) 平常負荷の把握と予測展開

図-4はこれまで述べたプロセスのシステム構造である。

周期モデルを適用した計算処理は、カレンダーテーブルに基づいて漏れなく忠実に行うプログラミングが必要である。もし、対象とする日付に漏れや重複がある場合は、漸近計算において誤差が縮小せず拡大し計算不能となる。このプロセスを1台のサーバについて24時間・24回行うことで全時間帯の負荷を網羅し、(4)で後述する全時間帯の立体表示が可能となる。

また、3項で述べた、祭日やユーザ独自の特別な日の指定は外部パラメータで与え、各モデルの一つの要素として取り扱うこととした。

(3) 散発的負荷(バラつき)の可視化

2項において散発的な負荷(継続)は、本来は平常負荷の一部とすべきであると述べた。そして、平常負荷と同様に一つのモデルとして定義できれば、より定常負荷の把握が容易になると思われるが本稿では今後の課題とした。

このため、実績値と平常負荷の誤差（月単位の移動標準偏差）を求め、未来の平常負荷線の上下にその2σ値の近似線幅をグラフに可視化した。（詳細は省略）

(4) 鳥瞰できる立体的な可視化

実績値および平常負荷の未来値を全時間帯で週周期を横軸にして立体的グラフで表現することで、容易に全負荷を鳥瞰できるようにした。次の6項で幾つか事例を示す。

以上までの処理は市販のパソコンでサーバ1台当たり約2分であった。台数が多い場合は、夜間もしくは週末でのバッチ処理によるグラフ作成が必要となる。

6. 各種推移パターンによる検証

各種の推移パターンを有する数百台のサーバを対象に、開発した運用SEツールを適用して実用性を検証した。使用したグラフは次の様である。

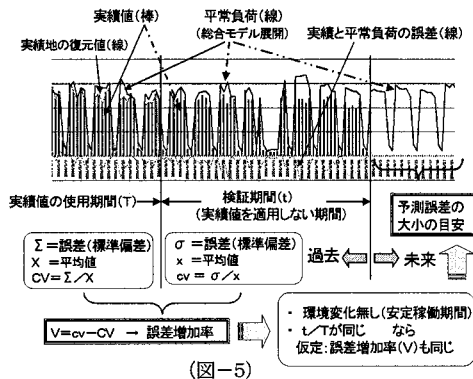
⑤ 全24時間立体のグラフ

実績と予測の鳥瞰的な把握用

⑥ 指定時刻の推移グラフ

長期および日々の確認用

また、数値による平常負荷の捕捉の程度や予測の誤差を評価することは一般に困難であるが、次に示す項目により、一定の検証期間で比較した数値により、未来での誤差を模擬する方法を試みた。（図-5参照）



T=実績値の使用期間長

t=検証期間長(実績値を適用しない期間)

X=T期間の平均値

x=t期間の平均値

Σ =T期間の実績と平常負荷の誤差(標準偏差)

σ =t期間の実績と平常負荷の誤差(標準偏差)

$CV = \Sigma / X$: T期間の誤差の変異(変動)係数

$cv = \sigma / x$: t期間の誤差の変異(変動)係数

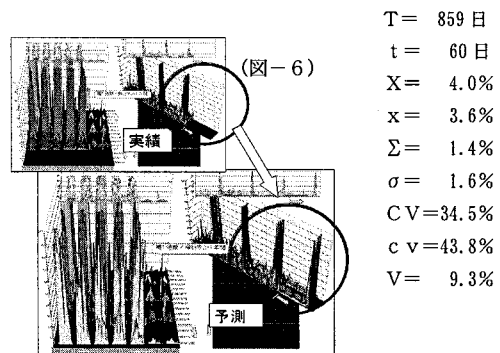
$V = cv - CV$: 変異(変動)係数の増率

ここで、VはT期間で作成した平常負荷をt期間に適用したときの誤差の増加率と考えられる。未来にt期間を設け同様の適用を行った場合、サーバの稼働環境が同じで、t/Tも同じ場合どちらも同程度の誤差が発生するであろうと仮定して、予測誤差の大小判断に使用する。

以上の考え方により、各種の負荷推移パターンの事例を次に示す。各図は、横軸1週の過去3年、未来1年間の立体的な負荷推移グラフである。また数値は上記のデータ項目の値である。なお、t期間は外部から与え、T期間は先の6項で述べた方法により変化点を自動検出して選択する。傾向モデルは直線が指数近似かを自動判断する。また、各周期モデルは各単位周期の1.2倍以上のT期間があれば自動的に適用することとした。

(1) 先鋭的な年周期型

図-6は年度末に処理が集中している例である。

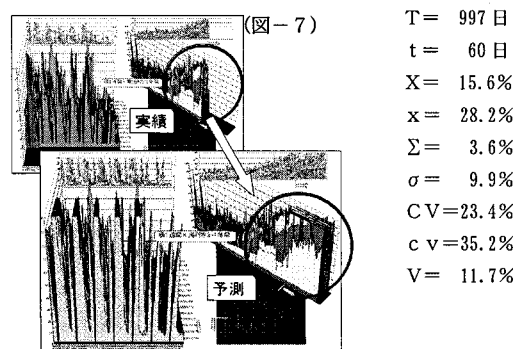


年度末処理を検証期間とするため、t期間を120日として、再計算した数値は次の様にVが増加しピーク時期が不安定であることを示している。

$T=841 \text{ 日}$, $t=120 \text{ 日}$, $X=3.7\%$, $x=8.2\%$, $\Sigma=1.1\%$,
 $\sigma=3.7\%$, $CV=31.05\%$, $cv=45.8\%$, $V=14.8\%$

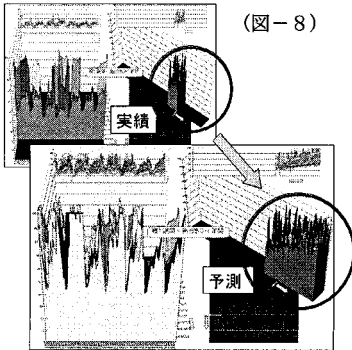
(2) 長期漸増型

図-7は、3年間に亘り増加傾向にあり、至近で負荷が100%に近づき、オーバする例である。



(3) 短期急増型

図-8は、稼働期間が5ヵ月と短い、安定した増加傾向であるためVが低く、比較的予測誤差は少ない。

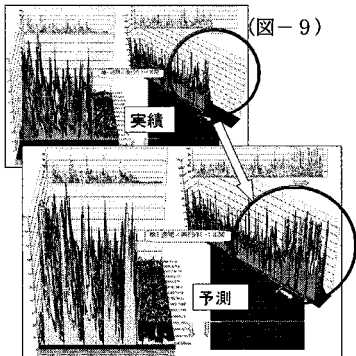


(図-8)

T = 129日
t = 30日
X = 37.6%
x = 41.3%
 $\Sigma = 3.3\%$
 $\sigma = 8.0\%$
CV = 8.8%
c v = 19.3%
V = 10.0%

(4) 月周期性

図-9は毎月ピークが発生するケースの例であるが、定期的な周期が主でVが小さく、予測誤差が少ない。

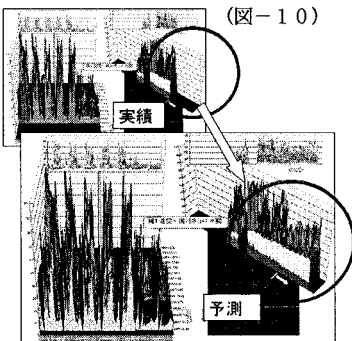


(図-9)

T = 833日
t = 60日
X = 7.0%
x = 11.2%
 $\Sigma = 1.8\%$
 $\sigma = 3.6\%$
CV = 26.0%
c v = 32.2%
V = 6.1%

(5) 乱高下負荷型

図-10は周期性が少なく、乱高下する負荷である。Vが大きく予測は参考程度となる。要注意のサーバといえる。



(図-10)

T = 259日
t = 60日
X = 22.8%
x = 19.0%
 $\Sigma = 4.1\%$
 $\sigma = 7.5\%$
CV = 18.1%
c v = 39.3%
V = 21.2%

事例に示す通り、負荷推移パターンや対象期間により一概に言えないが、ほぼ忠実に実績値による平常負荷を捕捉し、未来の全体負荷の推移を鳥瞰できることが分かる。これにより、多数のサーバを対象に現負荷の妥当性や当面の安全性を短時間に把握・予測することを支援できると思われる。

7. おわりに

昨今のサーバ台数の増加により、ピーク負荷がどのような日時に発生しているか、運用現場のシステム管理者でさえ把握が困難になっている。

本稿は、運用経験則によりサーバの周期性を掘り下げ、平常負荷(CPU使用率)の把握と予測について述べた。

これは、全体の傾向と複数の周期モデルによる平均値を把握して、カレンダー上で過去・未来に展開するもので、ユーザにとって比較的理解しやすい方法である。

しかし、一定量のデータを採取できて、はじめて意味のある情報が得られることから、サーバ負荷の場合では、短期でも最低3ヵ月、年周期では最低1年から2年以上のデータ蓄積が必要である。

今回は、周期モデルを主体に述べたが、周期性がなくても負荷実態を捕捉するモデルは存在する。例えば、曜日別の増減などの傾向モデル、散発的(バラツキ)負荷の程度を表すモデル、負荷変化点の自動検出による安定稼働期間を見極めるための機能などもモデルの一つと考えられる。

また、本稿で述べた平常負荷把握と予測の方法は、サーバ負荷に限らず、ホストコンピュータや通信回線、更には営業窓口、インターネットなど、カレンダーに基づいて稼働する業務負荷に広く適用できるものと思う。

今後も、システムの運用管理は常に改善が必要とされる。このため、システム運用に携わる方々が共有している運用経験則をモデル化する技術の向上策が引き続き望まれる。

参考文献

- i. 基本統計学「有斐閣ブックス：宮川公男」
- ii. 基礎統計学「東京教学社：佐藤喜代蔵」
- iii. 統計グラフのウラオモテ「講談社：上田尚一」
- iv. 情報処理学会研究報告 2004-EVA-11「サーバ群の運用における自動負荷予測手法：塩川孝雄」
- v. 情報処理学会研究報告 2006-EVA-17「大規模サーバ群の負荷予報システムの構築に向けて：塩川孝雄」
- vi. 時系列解析と予測(改訂第2版)「シーエービー出版：P. J. ブロックウェル R. A. デービス著 訳」