

## 7V-3

## システムレベルアップの高速化

未成 徹、金田 吉信、富沢 哲夫

(富士通株式会社)

## 1.はじめに

近来の計算機システムにおいては、エンドユーザ指向重視のシステム開発に重点がおかれており早く、より簡単で、より信頼度の高いシステムに対するユーザ要求が高まっている。

これらのユーザ要求は計算機システムにおけるすべての機能（導入から運用、保守作業にいたるまで）にわたっている。

今回、システムレベルアップ作業における問題点の解決をテーマに研究・開発作業を行う機会を得たので、その概要について説明する。

## 2.現状の問題点

ほとんどの計算機システムにおいては、一括修正を適用することにより、システムを構成するプログラムのメンテナンス作業およびシステムレベルアップ作業が行われている。

システムレベルアップ作業での現状の問題点を整理すると以下のように分類される。

## ① 作業時間が長い

$$\text{システムレベルアップ作業時間} = \text{一括修正適用作業時間} + \text{修正確認時間}$$

$$(9\text{時間}) \quad (\alpha\text{時間})$$

一括修正適用作業時間（9時間）は、4世代累積された一括修正を適用するのに必要な時間である。

システムレベルアップ作業については、一括修正の適用作業だけに留まらず、前後作業および修正確認作業が必要である。

図2.1に示すように一括修正の適用作業とその前後作業のために約9時間程度の時間が必要であり、修正確認のため $\alpha$ の時間が必要である。

その結果、システムレベルアップ作業のために、二日間を要することとなる。

## 【影響】

- ① システムレベルアップ作業を行う機会がない。
  - プログラムの障害が長い間修正されない。
  - レベルアップされたプログラムの最新機能を使用することができない。
- ② ユーザ／SEの負担が大きくなる。
  - 平日にシステムレベルアップ作業が、ユーザ業務の停止が不可能なため実施できない。したがって、休日／深夜等を利用した作業を行わざるを得ない。

## ② 作業手順が複雑でミスをおかしやすい。

システムレベルアップ時の作業手順が難しいため、ミスをおかしやすい。

## 【影響】

- ① トラブルを誘発しやすい。
  - システムレベルアップ作業において、最もトラブル発生件数の大きいものがファイル容量拡張に関するものである。フィールド部門から寄せられるインシデント件数においても、大きい割合を示している。
  - インシデント総件数 226件中、ファイル容量拡張関連で 75件発生 (31%)。
- ② トラブル発生時に手戻り作業時間が必要である。
  - さらに、時間を必要とする要因となる。

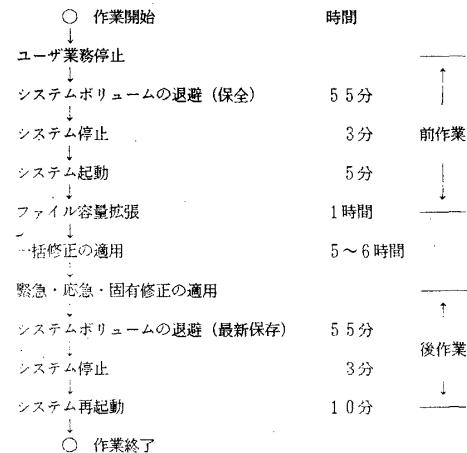


図2.1 現状のシステムレベルアップ作業における作業手順

## 3.問題点の原因

では、なぜ現状のシステムレベルアップ作業においてこのような問題点が発生するのか、その原因について考えてみた。

## ① 時間がかかる原因

## ① 一括修正の適用時間が長い。

- 第一に、一括修正に含まれる内容として、障害修正に値する部分と機能追加に値する部分とが混在しているため、修正件数そのものが多くなる。

- 第二に、従来の一括修正の適用方式は、モジュール単位の入換方式を採用している。そのため修正件数が多いとファイルの満杯状態が発生することになる。ファイルの満杯状態が発生すると、自動圧縮処理が行われるファイルの自動圧縮処理のためには、以下の時間が必要となる。

【例】LINKLIB : 40分

LPALIB : 20分

このことでも一括修正の適用時間は長くする一つの原因であると考えられる。- 第三に、使いやすいシステムを提供する目的で、一括修正により様々な機能追加部分を提供している。短期間に機能追加部分の開発作業を完了し一括修正により提供するという作業が繰り返され、平均1.5ヶ月に一回の周期で一括修正の提供が行われている。ユーザ側においても当然新機能を早く使用したいと考えるため、一括修正の適用作業を実施せざるをえない。ユーザ先の業務の関係でシステムレベルアップ作業を度々実施できないユーザについては、連休等を利用して累積された一括修正を利用し、適用作業を実施するであろう。このことも、一括修正の適用時間を長くする一つの原因であると考えられる。

## ② システムレベルアップの作業時間が長い。

システムレベルアップ作業では、一括修正の適用作業だけでなく前後作業が必要であることは前述した。この前後作業について見てみると、時間を必要とする作業に、システムボリュームの退避作業とファイル容量拡張作業がある。

- 第一に、システムボリュームの退避作業については、一括修正の適用作業の前後（保全と最新システムの保持の目的）に実施するため計二回、約2時間程度の時間が必要となる。

- 第二に、ファイル容量拡張作業については、一括修正の適用作業によるシステムファイルの容量増加分については、一括修正の適用前に必ず実施なければならない。ファイル容量拡張のためにはおよそ以下の時間が必要となる。

【例】LINKLIB : 50分

LPALIB : 30分

これらの前後作業のために時間を必要とすることが、システムレベルアップ作業全体の作業時間が長くなる原因となっている。

- (2) 作業手順が複雑でミスをおかしやすい原因  
 作業手順が複雑でミスをおかしやすい第一の原因として、システムレベルアップ作業の前作業として、ファイル容量拡張を実施しなければならないことが挙げられる。ファイル容量拡張作業において、以下に示すユーザミスを原因とするトラブルが発生している。
- ① 容量計算が必要である、計算ミスを誘発しかねない。
  - ② バラメタの指定ミス。
  - ③ 作業用ファイルのスペース不足。
  - ④ 異常終了時の作業ファイルの残留、再実行不可能。
  - ⑤ IPL忘れ、オプションインストール作業後にIPLしていない。

これまでに説明したシステムレベルアップ時の問題点と原因を整理すると図3-1のようになる。

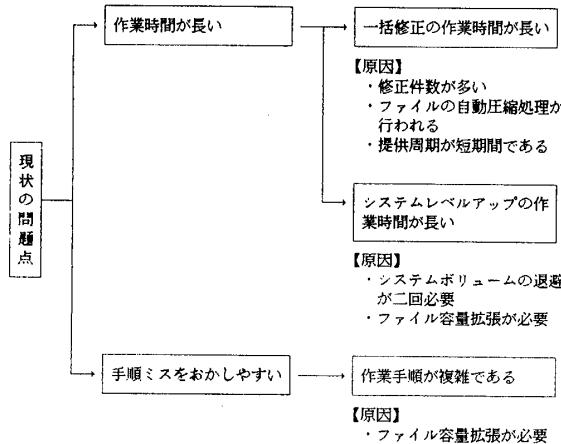


図3-1 システムレベルアップ作業時の問題点とその原因

#### 4. 問題点の解決策

これまで述べた現状のシステムレベルアップ作業における問題点を解決する目的で、研究・開発作業を行ってきたわけであるが、改善の第一段階を完了し、一応の成果を得ることができたため、その解決策の概要について説明する。

##### 4.1 ねらい

- 現状の問題点に対処するため、次の二点を改善のねらいとした。
- (1) 時間の削減
  - (2) システムレベルアップ作業時のトラブル削減（インシデント件数の削減）

##### 4.2 問題点の解決手段

以下の改善策により、問題点の解決をはかることを第一段階として実現方式の検討を行った。

###### (1) 一括修正適用方式の改善

一括修正の適用方式を従来のモジュール単位の入換え方式から、修正のあったファイル単位の一括入換え方式に変更する。  
 すなわち、ユーザ先システム上で基本システム提供以外のメンバ（オプションプログラム資産等）を退避し、ファイル一括入換え方式の一括修正を適用することにより入換え対象ファイルの先頭から基本システム提供メンバを一括して置換し、更に採否した基本システム提供以外のメンバを復元する方式である。

ファイル一括入換え方式による一括修正適用方式のイメージについて図4-1に示す。

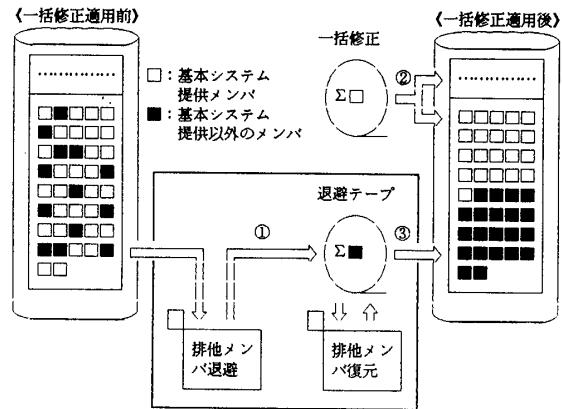


図4-1 ファイル一括入換え方式による一括修正適用方式

#### 5. 改善後の効果

一括修正を従来の形式によるものからファイル一括入換え方式にした場合のメリットとしては以下のようない項目が挙げられる。

- ① 適用時間が短縮される。
- ② 従来のモジュール入換え方式ではなく、ファイル単位に、しかもファイルの先頭から一気に入換える方式であること。加えて、ファイル入換え処理時に磁気テープを高速走行モードで動作させているため時間が早い。
- ③ ファイル容量拡張が不要となる。
- ④ 一括修正適用時にファイルが満杯状態とならない。
- ⑤ 一括修正適用時間が一定である。

従来は修正件数により適用時間が可変であった。

⑥ ユーザミスによるトラブルの削減（②と③で作業手順が減るための二次効果）

#### 6. 実測例

システムレベルアップ作業を従来の形式で作業した場合とファイル一括入換え方式により作業した場合の時間を比較し、表6-1に示す。

表6-1 性能測定結果

項目	作業手順	従来値	改善値
1 オプションメンバの退避	—	14分	—
2 業務停止	—	—	—
3 システム停止	3分	3分	—
4 システム起動	5分	3分	—
5 システムボリュームの退避	55分	55分	—
6 ファイル容量拡張	1時間	—	—
7 一括修正の適用	6時間	1時間8分	—
8 オプションメンバの復元	—	6分	—
9 緊急・応急・固有修正適用	—	—	—
10 システムボリュームの退避	55分	55分	—
11 システム停止	3分	2分	—
12 システム再起動	10分	10分	—
システムレベルアップ作業時間	9時間11分	3時間36分	—

\*一括修正の適用時に、処理時間の面で優れた効果が現れている。

#### 7. おわりに

作業時間の短縮をはかる目的で研究・開発を行ってきた本機能は、改善の第一段階としては一応の成果が得られた。しかし、まだ改善の余地があり、今後第二、第三のステップを踏み、よりよいものにしていきたいと考える。