

ES/1 NEO

MFシリーズ

MF-SCOPE

パフォーマンス・チューニング作業

第3版 2024年 2月

©版權所有者 株式会社 アイ・アイ・エム 2024年

© COPYRIGHT IIM CORPORATION, 2024

ALL RIGHT RESERVED. NO PART OF THIS PUBLICATION MAY
REPRODUCED OR TRANSMITTED IN ANY FORM BY ANY MEANS,
ELECTRONIC OR MECHANICAL, INCLUDING PHOTOCOPY RECORDING,
OR ANY INFORMATION STORAGE AND RETRIEVAL SYSTEM WITHOUT
PERMISSION IN WRITING FROM THE PUBLISHER.

“RESTRICTED MATERIAL OF IIM “LICENSED MATERIALS – PROPERTY OF IIM

目次

第 1 章 パフォーマンス・チューニング作業	1
1.1 本章の使用方法.....	1
JOB01n	2
JOB02n	3
JOB03n	4
JOB04n	5
JOB05n	6
JOB06n	7
JOB075	8
JOB095	9
JOB105	10
JOB115	12

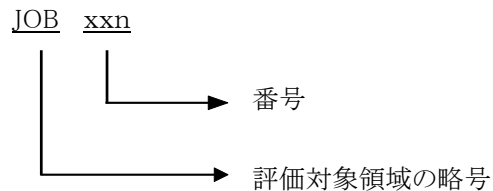
第1章 パフォーマンス・チューニング作業

1.1 本章の使用方法

MF-SCOPEでは、各種のチューニング・ヒントを出力します。しかし、MF-SCOPEが出力するメッセージだけでは、そのすべてを説明することは困難です。この章では、MF-SCOPEが出力するチューニング・ヒントに対応したチューニング作業について解説します。なお、チューニング作業の具体的な実施方法で疑問などがあれば、弊社担当者へお問い合わせください。各種の方法で、チューニング作業の支援を行います。

出力されるチューニング・ヒントには、参照コードと重要度が付加されています。参照コードは、評価対象領域の略号と3桁の番号により構成されています。この参照コードでは、本章の各ページの上段に示されたページ識別名に対応付けられています。

<参照コード/ ページ識別名の形式>



JOB01n

【説明】

業務プログラムが実行する際のイニシエータ待ち時間が長くなっています。対象ジョブの実行クラスを変更するかイニシエータの割当を調査してください。

【解説】

ジョブの実行は①入力処理、②変換処理、③実行処理、④出力処理、⑤ページ処理に大別することができます。

①入力処理	②変換処理	③実行処理			④出力処理	⑤ページ処理
イニシエータ待ち時間	データセットの 排他制御	装置割当処理	実行時間	出力待ち	印刷・穿孔	

図 1.1

ジョブを実行するには、ジョブ・ストリームをカード読取装置、TSO/TSSのSUBMITコマンドやインターナル・リーダ経由でジョブ入力サブシステム(JES/JSS)に渡さなければなりません。その後、入力処理と変換処理を行い実行可能状態になります。この入力と変換処理では、次のことを行います。

①入力処理

入力されたジョブ・ストリームはJCL部とSYSIN部に分割されてスプール・データセットに保管されジョブ番号が付られます。通常、この処理に必要な時間は非常に短いものです。また、RJEからの入力の場合は転送時間も含まれます。

②変換処理

入力されたJCLを内部テキストと制御ブロックに展開し再度スプール・データセットに書き出します。

③JES3 (IBM)、JSS4 (日立)、JES/E (富士通)を使用している

場合には、変換処理が終了した後に装置割当処理を行います。

前述の処理が完了した時点で実行可能状態となり、指定されたジョブ・クラスのキューに優先順位を基につながれます。その後、対象のイニシエータがアイドルとなった時点でそのジョブ・クラスで選択優先順位の高いジョブを実行の対象にします。このイニシエータ待ち時間は、ジョブがシステムに入力されてから先頭のジョブステップが開始されるまでの時間間隔をいいます。従って、対象のイニシエータがすでに他のジョブを実行しているか、または優先順位の高いジョブが待っている場合にこのイニシエータ待ち時間が長くなります。また、この待ち時間には、自動運用などにより”HOLDキュー”に置かれていた時間もふくまれますので注意してください。

【対応策】

- 対象ジョブ・クラスのイニシエータ数を確認しその数を増やしてください。
- 業務プログラムのジョブ・クラスを変更してください。

JOB02n

【説明】

業務プログラムが実行する際のデータセットの排他制御待ち時間が長くなっています。

【解説】

ジョブの実行は①入力処理、②変換処理、③実行処理、④出力処理、⑤ページ処理に大別することができます。

①入力処理	②変換処理	③実行処理		④出力処理	⑤ページ処理
イニシエータ待ち時間	データセットの 排他制御	装置割当処理	実行時間	出力待ち	印刷・穿孔

図 2.1

ジョブの入力処理と変換処理が終了した後に該当ジョブがイニシエータに選択されて実行が開始される場合に、このデータセットの排他制御待ち時間が発生します。イニシエータはジョブを実行させる為に必要な制御情報をSWA（スケジューラ・ワークエリア）に作成します。この際に、該当ジョブが使用するデータセットの排他制御要求（ENQ）を行い、対象とするデータセットがすでに他のジョブで使用されている際に待ち時間が発生します。この待ち時間は、データセットを確保しているジョブが解放するまで待たされます。これは、ジョブ間やTSO/TSSユーザとバッチジョブ間などの競合により発生します。この競合状況はパフォーマンス計測ツール（IBMはRMF、富士通はPDL/PDA、日立はSAR/D）によってレポート（ENQによる競合）することができます。このレポートを調査してジョブの実行時間帯の変更を検討してください。また、この中には、TSO/TSSでSUBMITしたジョブが使用するファイルを、すでに自分が排他制御で使用していた場合に排他制御待ちが発生する単純なユーザ・ミスもあります。

【対応策】

- 競合状況を把握して必要であれば業務プログラムの実行時間帯を変更してください。

JOB03n

【説明】

業務プログラムが実行する際の装置割当処理時間が長くなっています。

【解説】

ジョブの実行は①入力処理、②変換処理、③実行処理、④出力処理、⑤ページ処理に大別することができます。

①入力処理	②変換処理	③実行処理		④出力処理	⑤ページ処理
イニシエータ待ち時間	データセットの 排他制御	装置割当処理	実行時間	出力待ち	印刷・穿孔

図 3.1

ジョブの入力処理と変換処理が終了した後に該当ジョブがイニシエータに選択されて実行が開始される場合に、データセットの排他制御と装置割当処理が実行されます。イニシエータはジョブを実行させる為に必要な制御情報をSW A(スケジューラ・ワークエリア)に作成します。この際に該当ジョブが使用するデータセットの排他制御要求(ENQ)を行い、状況によっては待ち時間が発生することがあります。そして、装置割当ルーチンを使用して次のような装置割当処理を行います。

- ◎カタログされているデータセットについては、STEP CAT/JOB CAT、修飾名、マスター・カタログの順に検索し装置名やボリューム通番を取得します。
- ◎ボリューム通番から装置番号への変換と、必要であればマウント・メッセージを出力します。この際、ディスク・ボリュームについてはマウント処理が完了するまで待ちますが、テープ・ボリュームについてはマウント・メッセージの出力だけでそのマウントの完了は待ちません。(テープ・ボリュームのマウントの遅延時間は“LONG WAIT”(長時間の待ち状態)となり、実行時間に含まれます。)

【対応策】

- JCLで指定しているデータセットのカタログ管理を調査してください。

JOB04n

【説明】

業務プログラムの処理経過時間が長くなっています。この原因として、業務プログラムが実行不可能状態の長時間の待ちが考えられます。

【解説】

ジョブの実行は①入力処理、②変換処理、③実行処理、④出力処理、⑤ページ処理に大別することができます。

①入力処理	②変換処理	③実行処理			④出力処理	⑤ページ処理
イニシエータ待ち時間	データセットの 排他制御	装置割当処理	実行時間	出力待ち	印刷・穿孔	

図 4.1

この実行処理のなかで実行時間はさらに次のように分類することができます。



図 4.2

業務プログラムが実行を開始してから終了するまでの時間間隔が実行時間であり、アクティブ時間と長時間の待ち時間に分類できます。アクティブ時間は資源管理プログラムによる実測値でありスワップ・イン状態かまたはスワップ・アウト状態の実行可能状態を示します。一方の長時間の待ち時間は、スワップ・アウト状態で実行不可能な状態を示し“LONG WAIT”時間です。この原因としては、テープ・ボリュームのマウント待ちや入出力動作の完了待ちなどがあります。また、特殊な業務プログラムの場合には、ユーザからの指示を待っていることもあります。

【対応策】

- 特殊な業務プログラムでユーザからの指示待ち状態かを確認してください。
- テープ・ボリュームのマウント待ちの場合は運用を見直してください。

JOB05n

【説明】

業務プログラムの処理経過時間が長くなっています。この原因として、業務プログラムは実行可能状態にありますがスワップアウトされていることが考えられます。

【解説】

ジョブの実行は①入力処理、②変換処理、③実行処理、④出力処理、⑤ページ処理に大別することができます。

①入力処理	②変換処理	③実行処理			④出力処理	⑤ページ処理
イニシエータ待ち時間	データセットの 排他制御	装置割当処理	実行時間	出力待ち	印刷・穿孔	

図 5.1

この実行処理のなかで実行時間はさらに次のように分類することができます。



図 5.2

業務プログラムが実行を開始してから終了するまでの時間間隔が実行時間であり、アクティブ時間と長時間の待ち時間に分類できます。そして、アクティブ時間はさらに常駐時間とスワップ・イン待ち時間に分類されます。アクティブ時間は資源管理プログラムによる実測値でありスワップ・イン状態かまたはスワップ・アウト状態の実行可能状態を示します。一方の長時間の待ち時間は、スワップ・アウト状態で実行不可能な状態を示し“LONG WAIT”時間です。この要因としては、テープ・ボリュームのマウント待ちや入出力動作の完了待ちなどがあります。また、特殊な業務プログラムの場合には、ユーザからの指示を待っていることもあります。常駐時間は、資源管理プログラムが実測した時間でスワップ・イン状態でプロセッサの使用や入出力動作および何等かの事象待ちや実行待ち状態を示します。これに対して、アクティブ時間と常駐時間との差がスワップ・イン待ち時間であり、スワップ・アウト状態ですが業務プログラムは実行可能状態にあります。この要因としては、資源管理プログラムへのパラメータの制御に伴うものであり、ドメインによるプログラム多重度の制御やシステム資源が過負荷状態によるプログラム多重度の制御によるものが含まれます。

【対応策】

- ドメインでのプログラム多重度の指定や重要度を確認してください。

JOB06n

【説明】

業務プログラムが同一ジョブステップ内で、同じデータセットに対して複数回のOPEN/CLOSE命令を実行しています。これは、入出力負荷を増加させますのでその必要性を確認してください。

【解説】

業務プログラムがデータセットを使用する為にはOPENマクロ命令を実行しなければなりません。この際、データセットがディスク・ボリューム上に存在する場合は、データセット情報を取得する為にディスク・ボリュームへの入出力動作を実行しなければなりません。また、データセットの使用が完了した時点で最終参照日などの情報を記録する為に再度ディスク・ボリュームへの入出力動作を実行します。従って、同一データセットへの複数回のOPEN/CLOSEマクロ命令の実行は、処理経過時間が長くなる要因や入出力負荷を増大させる原因となり、他の業務プログラムへも悪影響を与えます。

一般的な業務プログラムでは、最初の初期設定時にOPENマクロ命令を実行し、処理完了時やプログラム終了処理でCLOSEマクロ命令を実行します。しかし、特殊な業務プログラムやデータセットについては、複数回のOPEN/CLOSEマクロ命令を実行する必要がある場合がありますので業務プログラムの仕様を確認してください。

【対応策】

- 複数回のOPEN/CLOSEマクロ命令を実行する理由を明確にしてください。明確な理由がない場合は、業務プログラムを修正してください。

JOB075

【説明】

業務プログラムが実行する際のページングによる遅延時間が長くなっています。ストレージの使用状況を調査して対処してください。

【解説】

業務プログラムは仮想記憶域で動作しており、実際にその業務プログラムが実行する際にのみその実行に必要なページ群が主記憶上に存在します。その他の残りのページ群は拡張記憶や外部記憶に追い出されています。そして、次に拡張記憶や外部記憶に存在するページが必要となった際には、そのページを主記憶上に持ってくる動作を行いその業務プログラムの実行を継続します。この動作がページ不在割り込み処理です。

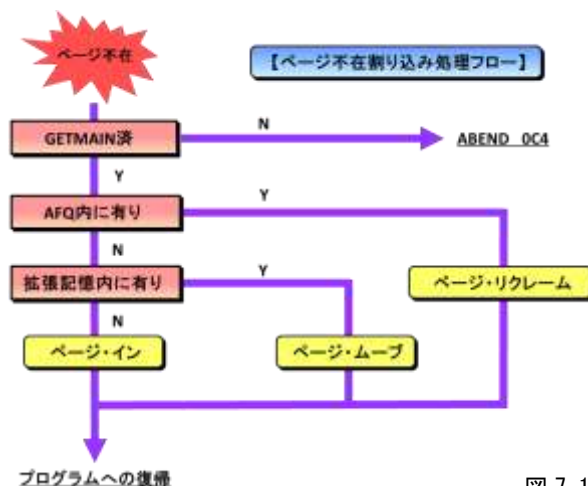


図 7.1

業務プログラムの実行中にページ不在割り込みが発生すると、VSM(仮想記憶管理プログラム)は、そのページがGETMAINされたものであるかを調査します。もし、GETMAINされていないページでのページ不在割り込みが発生した場合、業務プログラムは、「0C4」の異常終了となる。次に、GETMAINされたページでページ不在割り込みが発生した場合は、RSM(実記憶管理プログラム)やASM(外部記憶管理プログラム)の助けをかりて、ページ不在となったページを主記憶へ転送する必要があります。このページ転送を行う間、業務プログラムの実行は中断されます。この中断時間の事をページ不在時間と呼びます。このページ不在時間は該当ページが存在する場所により異なります。該当ページが拡張記憶に存在する場合は、拡張記憶から主記憶へのページ・ムーブ動作でページ不在割り込み処理を完了させます。この際の拡張記憶から主記憶へのページ・ムーブ処理時間は25から30マイクロ秒で非常に短い時間です。一方、該当ページがAFQ(利用可能フレーム待ち行列)や拡張記憶に存在しない場合は、外部記憶から主記憶へのページ・イン動作が必要となります。通常、このページ・イン処理時間は15から20ミリ秒程度で非常に遅くなります。

【対応策】

- ストレージ(主記憶や拡張記憶)の使用状況を調査してIPSパラメータでストレージ分離機能を使用してワーキング・セット・サイズを調整してください。
- 外部記憶装置をより高速な装置に移行してください。
- 半導体ディスクや拡張記憶機構の導入を検討してください。
- 主記憶を増強してください。

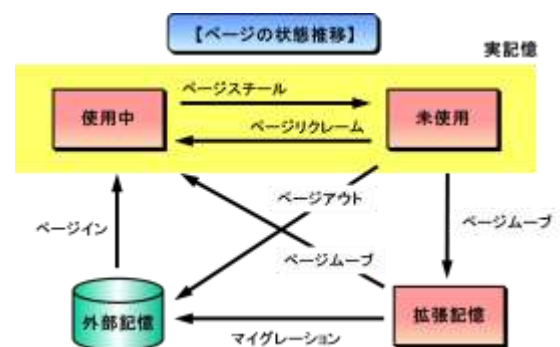


図 7.2

JOB095

【説明】

対象ジョブのページ処理で出力されるページ・レコードがありません。

【解説】

ジョブがシステムに入力されてからすべての処理を完了するまでに各種のレコードがSMF/SMSファイルに出力されます。ジョブ解析をする際に入力されたデータ群の中に一部または全部のジョブに対するページ・レコードがみつかりませんでした。

【対応策】

- SMF/SMSファイルにページ・レコードが出力されているか確認してください。これは、システム・パラメータ・ライブラリー(SYS1.PARMLIB)のSMF/SMS関連メンバーの指定を確認してください。
- 入力するデータをシステムの起動時から終了時までのデータ群を入力としてください。

JOB105

【説明】

業務プログラムがAIMやリレーショナルデータベースを使用した際、処理遅延の原因になり得る現象が発生しています。この現象には次の3つがあります。

- AIMで使用するバッファ不足
- デッドロック
- リレーショナルデータベースでのロック待ち
- AIMデータベースでアラームポイント超過

【解説】

1: AIMで使用するバッファの不足

AIM適用ジョブではトランザクションが開始してから終了するまでの間、データベースなどの実更新をバッファ上に蓄えておき、終了時点で一括処理を行うようにしています。更新データを基本的にバッファ上で管理することで無駄な入出力動作を抑止するのです。ところが処理に関連するバッファが満杯になると、途中で実更新が発生し実際に入出力動作が行われ、処理が遅れます。この現象に関連するバッファには次のものがあります。

バッファ名	用途	途中実更新の有無	定義方法
タスクログバッファ	タスク毎に存在し、更新されたデータの制御情報を維持します。実際の更新データはページバッファやデータセットバッファ上に存在します。	バッファ満杯時に発生	PED コマンド LOG BUFFER
ページバッファ データセットバッファ	更新データを保存するバッファでREAD/WRITE用バッファでもあります。	バッファ満杯時に発生	PED コマンド PAGE BUFFER
BOFバッファ	各タスク毎に更新前のレコードを維持するためのバッファです。		PED コマンド BOF BUFFER
HLFバッファ	トランザクション毎に発生したログデータを発生順にHLFやTLFに書き出す際のバッファです。		SIS コマンド BUFFER

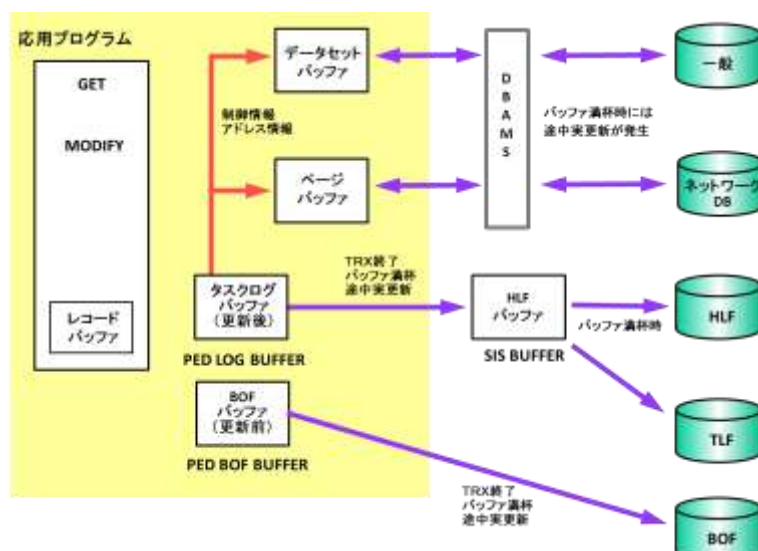
タスクログバッファが不足した際は途中で実更新が発生し、HLFバッファへの書き出しを行いません。この際にはタスクログバッファのサイズや数を見直す必要があります。タスクログバッファの不足によるHLFへの書き出し回数は次の式で求めています（厳密にはLOGWRITEマクロ命令によるHLFへの書き出し回数の一部も含んでいます）。

タスクログバッファの不足によるHLFへの書き出し回数

$$=[\text{HLF総書き出し回数}]-[\text{終了トランザクション数}]-[\text{ページバッファ不足(途中実更新)回数}]$$

ページバッファやデータセットバッファが不足した際にも途中で実更新が発生し、BOFへ更新前(BI)データの書き出しを行います。その為バッファのサイズや数を見直す必要があります。

BOFバッファ不足の際はBOFへの書き出しが発生します。この際はBOFバッファのサイズや数を見直す必要があります。



2: デッドロックとリレーショナルデータベースでのロック待ち

AIMやリレーショナルデータベースはデータ保全の目的で排他制御を行います。排他制御の占有方式には次の2つがあります。

◆ 動的占有方式

業務プログラムがアクセスする部分をアクセスするたびに占有することで、占有範囲を小さくし占有時間も短くすることを目的としています。この方式は業務プログラムの処理多重度を向上させることが可能であり、オンライン業務に適しています。

◆ 一括占有方式

業務プログラムの開始時に一括して全資源を占有する為、デッドロックが発生することがなくバッチ処理に適しています。

例えばリレーショナルデータベースで動的占有方式を選択した場合、占有単位は格納領域かあるいはページ単位となり、占有時間は「最初に占有単位のデータにアクセスした時点からトランザクション終了まで」となります。この資源の占有解除はトランザクション終了時にAIMが自動的に行います。

このようにトランザクション処理中には必要な資源を排他制御する為、並列実行するトランザクション間でデッドロックが発生することがあります。AIMではデッドロックの自動検出機能を提供しており、デッドロックが検出された場合は一般的に最後にアクセス要求したトランザクションがキャンセルされます。

リレーショナルデータベースで動的占有方式を利用する際には占有待ち制御について指示することができます。これはシステムやジョブ単位のチューニングマクロ「ISWAIT」で行います。

3: AIMデータベースのアラームポイント超過

データ件数の増加に伴ってデータベースの格納領域が不足し、その結果データベースが停止状態になることがあります。この領域不足を未然に防ぎ、レンジやサブレンジ単位にアラームポイントを設定することができます。また拡張インデックス利用時には、対応するデータベースへのデータ追加に伴って拡張インデックスの領域も不足する状態が発生します。この為拡張インデックスに対してもレンジやサブレンジ単位にアラームポイントを設定することが可能となっています。これらアラームポイントの超過が確認されることで、データベースの再編成の時期を知ることができます。



アラームポイントの設定

SCHEMAコマンドまたはXIFSCHEMAコマンドで、RANGEエントリ/SUBRANGEエントリのALARMPOINT句

データベースの格納領域を監視する際にはオーバフロー域の切り出し件数も監視対象とすべきです。このオーバフロー域の切り出しは、アクセス回数の増加などにより処理効率が悪化することが考えられます。

【対応策】

- 1: バッファ不足の際には対応するバッファの適正なサイズや数を検討してください。
- 2: 排他制御のよる待ち時間が長い場合は占有単位と占有時間を再考する必要があります。
- 3: アラームポイント超過時にはデータベースの再配置を検討してください。

JOB115

【説明】

ジョブステップの終了処理やジョブステップの初期化処理で遅延が発生しています。プロセッサで出力される項目の意味を下記に示します。

「JOBANLSTの出力項目」

対象となったジョブの一覧を出力します。

JOB (CLS)	ジョブ名 (ジョブクラス)
STEP_ELAPSE	ジョブステップの合計処理経過時間 (HH:MM:SS. TH)
DELAY	総ジョブステップ遅延時間 (HH:MM:SS. TH)
JOB_TERM	最終ジョブステップからジョブ終了までの遅延時間 (HH:MM:SS. TH)
RDR	入力日時 (YY/MM/DD:HH:MM:SS. TH)

「JOBCTL10の出力項目」

JOB ELAPSE TIME	ジョブレコードから算出した処理経過時間 (HH:MM:SS. TH)
TOP JOBSTEP ELAPSE TIME	ジョブステップの合計処理経過時間 (HH:MM:SS. TH)
TOTAL DELAY TIME	総ジョブステップ遅延時間 (HH:MM:SS. TH)
MAX JOBSTEP DELAY	最大ジョブステップ間遅延時間 (HH:MM:SS. TH)
STEP NUMBER x TO y	最大値を記録したステップ番号
JOB TERMINATION DELAY	最終ジョブステップからジョブ終了までの遅延時間 (HH:MM:SS. TH)

【解説】

ジョブステップ間遅延時間は、ジョブステップが終了してから次のジョブステップが開始されるまでの時間間隔と最終ジョブステップが終了してからジョブが終了するまでの時間間隔を意味します。この際、ジョブステップ終了日時はジョブステップ終了レコードの書き出し日時、ジョブ終了日時はジョブ終了レコードの書き出し日時を意味します。

バッチジョブの実行処理は次のように処理されます。

	SMF レコードに記録されるタイムスタンプ
①ジョブ選択	
②ジョブステップ開始処理 ・システムが必要とする制御ブロックの作成 ・出口ルーチン処理	
③ジョブステップの実行 ・データセットの排他制御 ・装置割当 ・ATTACH マクロでサブタスクを生成し 業務プログラムを実行	ジョブステップ開始日時 装置割当開始時刻 業務プログラム開始時刻
④ジョブステップの終了処理 ・DETACH マクロでサブタスクを削除 ・データセットの解放 ・装置の解放 ・出口ルーチン処理 ・SMF レコードの作成・出力	ジョブステップ終了日時
⑤最終ジョブステップまで②～④の繰り返し	
⑥ジョブステップの終了処理 ・データセットの後処理 (JCL の DISP 句) ・装置の解散 ・出口ルーチン処理 ・SMF レコードの作成・出力	

上記③ジョブステップの実行は、EXEC文に指定された業務プログラムが実際に実行する部分となります。④ジョブ

ステップ終了処理でジョブステップ終了レコードが出力されます。ジョブステップ間遅延時間は、④ジョブステップ終了処理でジョブステップ終了レコードが出力されてから、次のジョブステップが開始するまでの時間間隔を意味します。

JOBCTL10プロセッサでジョブ単位に解析した際には、ジョブステップ間遅延時間はジョブステップ終了レコードを読み込まない為検査されません。ジョブステップ間遅延時間が発生している際には、ワークフローの長時間待ち(LONG WAIT)として示されます。

【対応策】

ジョブステップ間遅延時間が長い場合は、ジョブ実行時間帯のシステム状況を確認してください。何らかのシステム資源が過負荷状態になっている可能性があります。また、最終ジョブステップとジョブ終了間の遅延時間が長い場合は、SMFレコードの出力処理が長くなっていることもあります。ジョブが使用(割当)したDD文やデータセット数が大量の場合には、長くなる可能性があります。

この場合、IBMシステムではSMFPRMxxメンバーの下記のパラメータを確認してください。

DDCONS=YES | NO (省略値はYES)

DDCONS=NOに設定することで終了処理時間が短縮されます。

下記のパラメータはz/OS V1R10以降で有効です。

EMPTYEXCPSEC=NOSUPPRESS | SUPPRESS (省略値はNOSUPPRESS)

EMPTYEXCPSEC=SUPPRESSに設定することで、タイプ30に出力されるEXCPエントリの数が少なくなります。