

# 大規模データ移行におけるリスク回避

塚本 俊 成石 直子

## Risk Management in Large Volume Data Migration

Suguru Tsukamoto Naoko Nariishi

プロジェクトを実施する上で、リスクを回避するためには、どこにリスクが潜んでいるのか、どのような点を考慮しておかなければならないのかを認識することが重要である。データ移行においても同様にリスク回避がプロジェクト成功のために重要であるが、一般的にこの認識が甘く、トラブルを引き起こすプロジェクトが少なくない。本論文は、このような現状を踏まえ、より実践的な面から、大規模データ移行において陥りがちな問題点と、リスク回避のための考慮点を計画フェーズ、設計開発フェーズ、実施フェーズの3つのフェーズより体系化して述べる。

In order to minimize risks in a project, it is indispensable to know where risks might be hidden. Generally speaking, data migration is usually given low priority, despite its significance in determining the eventual success of a project. It is therefore important to pay attention to the risks which might arise as a result of data migration failure. In this paper, issues in risk management with regard to large volume data migration are analyzed. This analysis categorizes data migration in to three phases: planning; design and development, and execution.

Key Words & Phrases : データ移行, 移行計画, リスク管理, 移行テスト, データ検証  
data migration, data migration plan, risks management, migration test, data inspection

### 1. はじめに

プロジェクトを実施する上で、どこにリスクが潜んでいるのか、どのような点を考慮しておかなければならないのかを認識することは、リスクを回避する上で重要である。あらかじめ、リスクを把握し、対応策を策定しておくことで、問題発生時にも冷静に対処することができる。

データ移行においてもリスク回避がプロジェクト成功のために重要であるが、データ移行に関しては、一般的にこの認識が甘く、トラブルを引き起こすプロジェクトが少なくない。

データ移行に関する文献としては、プロジェクトマネジメントに関するデータ移行の一般的プロセスや開発方法論 [1][2] や、データ移行の事例紹介 [3][4][5] を述べたものはあるが、データ移行のリスク回避について体系的に掘り下げた文献は少ない。

本論文は、このような現状を踏まえ、より実践的な面から、大規模データ移行において陥りがちな問題点と、リスク回避のための考慮点を整理する。

### 2. データ移行の開発工程とフェーズ

#### 2.1 データ移行システムの構成要素

現行システムから新システムへ、データを移行するためには、図1のように以下の4つの構成要素からなる「データ移行システム」の構築が必要となる。

- (1) データ取得  
現行システムからの移行元データの取得。
- (2) データ変換  
移行元データを新システムで使用可能な移行後データの形式へ変換。
- (3) データ検証  
データ取得、変換結果の妥当性の検証。

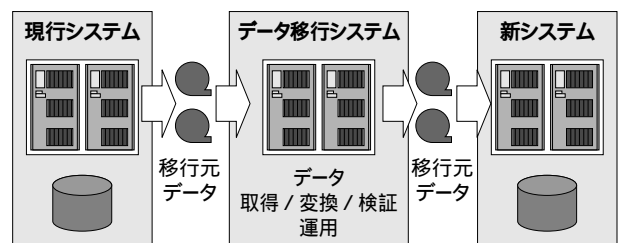


図1. データ移行システムの構成要素

提出日：2003年8月29日 再提出日：2004年10月29日

#### (4)運用

データ取得、変換、検証の各作業の実施。

### 2.2 データ移行システムの開発工程

データ移行システムの開発工程を、日本IBMの開発方法論 [1] で述べられているアプリケーションシステムの開発工程と統合して整理したものを図2に示す。

本論文では、データ移行システムの開発工程の計画局面を「計画フェーズ」、要件定義～開発実施/統合テストの局面を「設計開発フェーズ」、リハーサルと本番移行の局面を「実施フェーズ」として対応づけ、計画フェーズでは、計画立案の観点より、設計開発フェーズおよび実施フェーズでは、データ移行システムの構成要素の観点より、リスク回避の考慮点を述べる。

### 3. 計画フェーズでのリスク回避

計画フェーズでは、後続の設計開発フェーズ、実施フェーズの方針と計画を策定する。ここでは、アプリケーションシステムの開発工程の計画、および要件定義局面でのデータ移行に関連する内容を踏まえて作成することにより、後続フェーズで方針や計画が安易に変更してしまうリスクを抑えることができる。

そのために必要となる主要な方針および計画は以下のものになる。

- (1)データ移行実施方針
- (2)設計開発計画
- (3)検証計画
- (4)リハーサル計画
- (5)本番移行計画

#### 3.1 データ移行実施方針

データ移行実施方針は、各計画を立案するための基本方針となるものである。これには以下の項目を含む方針を、ここで定義しておくことが重要になる。

データ移行実施方針で定義されるべき内容は、普遍的なものであり、その定義が変わることは、各計画に影響が及ぶことを関係者全員が十分に認識してお

かなければならない。

- ・データ移行の対象
- ・データ移行の方法
- ・データ移行の体制
- ・検証方法
- ・その他の前提および制約事項

#### 3.2 設計開発計画

移行するデータ項目数およびデータ量が大规模であるほど、データ移行システムは複雑になり、計画の重要性が増す。設計開発計画には、以下の項目を盛り込む必要がある。これらの項目が綿密に作成されないと、完成したデータ移行システムの品質が不十分になったり、移行後データの品質に欠陥がでたりするリスクが高くなる。

- ・設計開発スケジュール
- ・設計開発体制
- ・テスト計画

設計開発スケジュールは、アプリケーションシステムの開発工程との依存関係(図2)を考慮して策定する必要がある。例えば、データ移行システムの外部設計が開始できるのは、アプリケーションシステムの外部設計で定義されるデータ定義、およびデータベース設計の完了後でなければならないというような制約がある。

データ移行システムで作成した移行後データの品質を検証するために、設計開発フェーズでできる限りアプリケーションシステムで移行後データを使い込むテスト計画とすることにより、移行後データの品質上の欠陥やアプリケーションシステムとの設計上の不整合を早期に発見することができる。

その他、システムの移行やセンター移転など、設計開発フェーズに影響のある事象もできる限り事前に洗い出しておくことで想定外の影響を受けるリスクを下げることができる。

#### 3.3 検証計画

検証計画では、移行元および移行後データ、そして

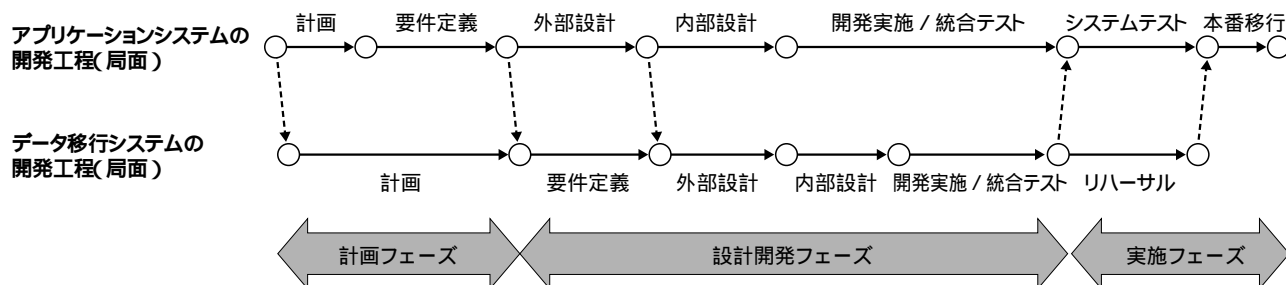


図2. データ移行システムの開発工程とフェーズ

開発したデータ移行システムそのものを検証対象として立案する。

最終的には、移行後データの品質確保が最重要であるが、そのためには、移行元データとデータ移行システムも含め、以下の各項目を盛り込んだ検証計画を立案しておく必要がある。

- ・ 検証対象
- ・ 検証方法
- ・ 検証タイミング
- ・ 検証手順
- ・ 検証実施体制
- ・ 検証基準(品質達成基準)

検証を実施する上で難しいのは、何を基準として検証完了とするかである。この基準はデータ移行システムでのデータ変換の仕様、新システムの業務内容およびデータ内容を熟知していなければ設定は難しい。そのため、早くから現行システムおよび新システム、さらに業務内容を熟知した担当者に参画してもらい検証計画を作成すべきである。

### 3.4 リハーサル計画

実施フェーズにおける本番移行を想定し、リハーサルを行っておくことは、本番移行時に発生するリスクを事前に回避することができるため有効である。そのための計画として、移行元データの取得から本番移行終了まで、一貫したデータ移行のリハーサル計画を作成する。これは、次で述べる本番移行実施計画を基にして作成する。

本番移行のトラブルを回避するためには、リハーサルを何度か繰り返す必要がある。リハーサル時には、次で述べる本番移行計画を基に経験を積み重ね、その内容を検証し改善していく。リハーサルを実施した後の評価、問題の対応期間も十分に確保しておくことで、問題の対応が未完了のまま次のリハーサルを迎えてしまうことが避けられる。リハーサルで発生した問題の対応策をいつ、どのように確認するかもこの計画で定めておく。

### 3.5 本番移行計画

データ移行システムの開発完了後、実際に本番移行する際の計画となる。これには以下の項目を網羅しておく。

- ・ 本番移行スケジュール
- ・ 運用方法
- ・ 作業手順
- ・ 管理手順
- ・ 実施体制
- ・ 本番移行環境
- ・ 移行完了基準

本番移行計画は、いかに短時間で、正確に移行するための運用方法、作業手順、管理手順、実施体制を事前にどれだけ作り上げられるかが、本番移行を成功させるための鍵となる。そのため、計画および手順は、極力シンプルにし、短時間で正確に実施可能な計画にしておく。

### 3.6 その他の考慮点

新システム全体を見たとき、データ移行以外の移行作業に対する、データ移行への影響の見極めも行わなければならない。

データ移行以外で影響を見極めておく必要のあるものとして、以下のようなものがある。

- ・ システム移行(機器、環境の切り替え)
- ・ 事務移行(事務処理手順の切り替え)
- ・ 運用移行(システム運用手順の切り替え)
- ・ センター移転(システム環境の切り替え)
- ・ その他のシステムの移行
- ・ フォールバック

## 4. 設計開発フェーズでのリスク回避

データ移行システムの設計開発フェーズでは、次に述べる点を考慮しておくことが重要である。

データ移行システムの4つの構成要素それぞれの観点でカテゴライズする。

### 4.1 データ取得

移行元データの取得に関しては、さまざまな制約の中で、いかに漏れなく正確にデータを取得し提供することができるかを考慮する。このためには、以下を考慮しておく必要がある。

#### ① データ取得対象の明確化

- ・ 新システムに関連するすべての周辺のシステム、外部機関によるデータ授受先まで含めて移行元データの対象を漏れなく洗い出す。
- ・ データ移行の対象データ項目をすべて洗い出す。同一データ項目が複数の移行元データとして存在するような場合を含めて、どのデータ項目がどの移行元データとなるかを明確にする。

#### ② 現行システムのデータ仕様

- ・ 現行システムの移行元データに影響する新規案件、変更案件を凍結させる。
- ・ 移行元データに不正なデータが入らないよう現行システムで対応する。
- ・ データ項目値として許容する範囲と不正データ項目値の範囲を明確にする。
- ・ 現行業務の制約(本番移行時は取引制限、入力制限、特殊入力させるなど)を行う。



現行システムでの対応が不可能で移行元データに影響がある場合には、運用上もしくは業務上のデータ入力を制限するなど、影響を抑える策を講じることを検討する。

データ移行システムは、新システムのデータ仕様だけでなく、現行システムのデータ仕様からも影響を受ける。実際のプロジェクトでは、仕様変更を完全に止めることは難しい。しかし、これを抑えなければ、移行後データの品質は安定しない。ここを認識し、最低限の変更に抑えるべく努力は惜しむべきではない。

### ③ データ凍結(データ取得)時期

- ・ 移行元データとしての凍結時期、取得可能時期を明確にしておく。

現行システムの停止後でなければ、移行元データが取得できないものは、データ取得にかかる時間を、本番移行スケジュールで確保しておかなければならない。

## 4.2 データ変換

データ変換の仕様はデータ変換のメインであり、入念な検討が必要である。ここでの考慮事項は以下のようになる。

- ・ データ変換の仕様

データ項目ごとに、現行システムおよび新システムでのデータ項目対比、データ項目値対比を定義する。現行システムおよび新システムのデータ項目、データ項目値に変更があった場合の変更対応方法も仕様を含めて検討する。

- ・ 初期値の設定項目

初期値の定義を明確にしておく。

- ・ 想定外データの変換仕様

想定外データ発生時の考慮をしておく。

想定外データとしては、不正データ項目(不正レイアウト)、不正データ項目値、不正データセット名、重複データなどがある。

想定外データは発生確率が高いため、発生時の対応、レポート方法を明確にしておく。また、想定外データの存在が本番移行前に判明した場合には、現行システム側にて、想定外データを事前に修正するなどの対応を検討し、想定外データを極力排除しておく。

- ・ 移行元データの取得漏れおよび不備対応

想定した移行元データが存在しない場合の処理を考慮しておく。データの一部が存在しないために、データ変換が全面的に実施不能になるような設計は避けるべきである。

- ・ データ移行の処理タイミング

移行元データの取得タイミングや移行後データの作成タイミングと、データ変換の実行タイミングとに矛盾がないようにする。つまり、移行元データが取得で

きていないにもかかわらず、データ変換が実行されることのないよう事前に十分精査しておく。

- ・ 処理効率

データ変換の処理効率を上げるための工夫(処理グループに分割し、並行処理を行うなど)を検討する。

- ・ リカバリーリスタート設計

データ変換を行うジョブが異常終了した場合には、異常終了個所からリスタートが可能な設計をしておく。これにより、大量データ処理の最後で異常終了した場合などに手戻りを最小限に食い止めることができる。

## 4.3 データ検証

データ検証の設計においては、検証対象ごとに検証方法を確定させなければならない。検証方法によっては、データ変換そのものに影響を及ぼすものがありうる。検証の種類によっては、データ検証の作業を効率化し、検証にかかる時間を短縮化するために、データ変換へのデータ検証機能の組み込み(例えば、件数検証のための件数カウントロジック)を検討しなければならない。

その他、データ検証の設計上の考慮点としては、以下の点がある。

- ・ 検証帳票(検証対象となる出力帳票)の共通フォーマット化
- ・ 検証処理(件数カウントなどの共通化できる部分)の共通ルーチンおよびマクロ化
- ・ 検証単位(タスク、ジョブ)の統一
- ・ 検証ポイント(検証作業のチェックポイント)の設定  
検証用に出力する情報の品質を均一化し、設計担当者に起因する不具合を最小限とするために、極力共通化できるものはそれを前提とした設計をすると良い。

検証ポイントとして検証作業を行うデータ変換のチェックポイントを設定し、その時点までのデータ変換の処理結果をすべて検証する。その後、後続のデータ変換を進めることで、障害の影響が拡散することを防止できる。ただし、チェックポイントでは検証作業を一時中断することになるため、設定が多すぎたり、設定が早すぎたりしても効率は悪化する。よって、チェックポイントの設定時には、データ検証の作業全体を見通した考慮が必要となってくる。設定の目安として、時間的には1~2時間ごと、処理の分岐が3ヶ所以上となる個所を設定するのが妥当であろう。

## 4.4 運用

時間的な制約はデータ移行システム全体に与える影響が大きいので、以下を十分に精査しておく必要がある。

- ・ データ取得の所要時間

移行元データの取得所要時間、取得タイミングを明確にしておく。

- ・データ移送時間  
データの移送(搬送)が伴う場合、その時間を測定しておく。
- ・データ変換の処理時間
  - ① データ検証作業を含めてデータ変換にかかる処理時間を正しく見積もっておく。
  - ② データ移行の作業時間として確保すべき時間は、一連のデータ変換および検証の所要時間の2倍以上確保しておく。これはデータ変換および検証の最後で致命的な問題が発生した場合でも、最初から処理をやり直すだけの余裕を持つためである。データ変換および検証に十分な作業時間が確保できない状況下で不具合が発生した場合、対応方法によってはサービスインへ直接影響するため、対応策を事前に十分検討しておく。
- ・キャパシティ見積もり  
移行するデータ量の増加を見越した設計(容量見積もり)を行うとともに、必要なリソースを確保する。リハーサル、本番移行のそれぞれの保管対象、保管期間を定めた上で、必要リソースの見積もりを行う。リハーサル期間中では最低限1世代前の移行先データはデータ検証の観点からも必要である。
- ・CPU資源  
本番移行中に、データ変換および検証処理以外(業務テストなどCPU競合する作業)を行わないように調整しておく。CPU資源はデータ移行システムの規模が大きければ大きいほど、移行所要時間に影響する大きな問題となるため、事前の調整が必要となる。

## 5. 実施フェーズでのリスク回避

実施フェーズのリハーサルおよび本番移行での考慮点は、リハーサルと本番移行での相違に焦点を当て、データ変換および検証の体制、運用の観点から列挙する。ここでのリスク回避の方法は、リハーサルと本番移行で同条件にできない点を明確化し、本番移行に限りなく近づけた状態でリハーサルを実施することにより、本番移行に備えることである。

### 5.1 データ取得

データ取得においては、以下のような項目では考慮が必要となる。

- ・データ取得のタイミング
- ・データ取得後の移行元データの移送方法  
リハーサルでは本番移行と同じタイミングでのデータ取得や同じ方法での移送が困難なものが多いため、

いかにして本番移行と同じように行うかを検討する

### 5.2 データ変換および検証

データ変換および検証の観点では以下の項目を考慮しておく。

- ・トラブル対応  
移行元データや移行後データの不備を即座に修正できるように、パッチツールを準備するなど、想定されるトラブルに短時間で対応可能な手順を確立し、リハーサルで確認しておく。
- ・リハーサル体制  
本番移行の実実施時と同等の体制、訓練を行えるように調整する。また、データ移行では想像以上の要員が必要となるため、必要となる役割を定義し、各部署への応援を含めて体制を確立させなければならない。オペレータを含め、本番移行実施時と同様の体制を、リハーサルにて確認しておくことが必須である。
- ・実施および検証体制  
専任のデータ検証体制を確立する。他と兼務した体制は極力避ける。データ変換および検証時間が長時間に及ぶ場合の体制、交代要員の手配を行う。

### 5.3 運用

実際の一連の本番移行における作業では、運用面でのミスも致命的となるため、少しのミスも犯さないための考慮が必要である。本番移行は、現行システム、新システムが停止可能な短い時間に集中して実施しなければならないことが多い。そのため、わずかな作業ミスやトラブルが命取りとなるのである。

- ・品質の確保と効率性  
データ変換のジョブ実行は自動化ツール(バッチジョブ運行管理ツールなど)を使用した運用とすることで手作業によるミスを防止し、品質と効率を上げることができる。
- ・運用のポイント  
極力人間判断を行わない運用とし、人的ミスが入る可能性を下げる。
- ・作業手順  
本番移行におけるすべての作業を手順化し、不明点の無いようにリハーサルにて確認しておくとともに、本番移行の担当者が事前に作業手順に慣れておく。
- ・管理手順  
進捗管理、障害管理はシンプルかつ正確に作成され、リハーサルにより確認し慣れておく。
- ・作業場所  
データ移行の作業場所が分散している場合、通信および連絡手段を確保しておく。
- ・実施環境  
サービスインに向け、本番移行の実施環境が、リ

リハーサル時とは変わる可能性があるため注意しておく。

・実施時期

本番移行を年末年始や連休に実施する場合、交通機関、交通渋滞、店舗営業時間、待機場所(ホテル)などリハーサルとは異なる状況に注意する。

・リハーサルの実施回数

リハーサルは最低4回実施する計画とするのがよく、3回以下とした場合、発生した問題の対策を十分確認することができないまま本番移行を向かえることになり、リスクが抑えきれない。

・リハーサルに潜むリスク

リハーサルで変更した個所はトラブルが発生するリスクが潜んでいる。リハーサルと次のリハーサルとの間隔は最短でも1ヶ月は確保しておきたい。それより短い期間だと評価、問題対応などが厳しいからである。

・リハーサルと本番移行の相違

リハーサルは本番移行とは異なる。意識の面もあるが、要員面、システム環境面、業務面など各面で違う面が出てくる。とかくリハーサルではテストの意識が強く、障害対応についても、本番時に発生したらという意識で臨むのではなく、次のリハーサルのために対応しようという意識が働いてしまうこともリスクを高める一因である。この違いも十分に認識しておき、本番移行実施時に初めて実施するもの、およびリハーサルと異なる個所については、十分に注意を払ってしかるべきである。

## 6. おわりに

実際の本番移行では、今まで述べたような点を考慮して綿密な準備が必要である。ただし、いかに周到に準備したとしても、必ず想定外の事故やアクシデン

ト、トラブルが発生する。これは人が作業する以上、ある程度避けられないものである。

しかし、いかに想定されるリスクに対し、事前に回避策を講じておくかにより、発生するトラブルを限りなく少なくすることができる。また、トラブルが発生した場合でも、その影響を最小限に食い止めることができる。

データ移行を軽視するなかれ、移行の成功なくして、新システムのサービスインはありえない。移行は一時のものといった考えを捨て、その重みを認識し、プロジェクトを成功に導いて頂きたい。

## 謝辞

本論文の執筆にあたっては、多数の有識者のご意見を参考にしたほか、プロジェクトメンバーより多くの助言を頂きました。あらためて深謝いたします。

## 参考文献

- [1] 日本アイ・ピー・エム、ADSG 第4版 第1刷、1999年10月
- [2] PMI Standards Committee, *Project Management Body of Knowledge-2000 Edition*, Project Management Institute, Newtown Square, Pennsylvania, ISBN1-880410-23-0, 2000
- [3] 日経BP社、特集システム移行危険がいっぱい、日経コンピュータ3月10日号、第569号、pp.40-55 2003年
- [4] 日経BP社、プロジェクト・マネジャ意思決定の瞬間、日経コンピュータ5月19日号、第574号、pp.168-171 2003年
- [5] 日経BP社、ITの新常識【システム統合】、日経コンピュータ6月2日号、第575号、pp.200-203 2003年



日本アイ・ピー・エム株式会社  
ICPアドバイザーPM

塚本 俊 Suguru Tsukamoto

### [プロフィール]

1987年、日本アイ・ピー・エム入社。金融系SIプロジェクトを中心に数々のシステム開発プロジェクト、データ移行プロジェクトに参画。いくつかの大規模データ移行のプロジェクトに参画し、プロジェクトマネジメントを実践。現在も金融系SIプロジェクトを中心に、プロジェクトマネジメント業務に従事。PMP、SPM会員  
SUGURU@jp.ibm.com



日本アイ・ピー・エム株式会社  
主任ITスペシャリスト

成石 直子 Naoko Nariishi

### [プロフィール]

1988年、日本アイ・ピー・エム入社。いくつかの金融・流通系のSIシステム構築プロジェクトに参画。複数の基幹システムプロジェクトにおいて大規模データ移行のプロジェクトマネジメントを遂行。2002年からコンピテンシーにてEPM(Enterprise Project Management)の推進展開およびPMメンタリングなどを担当。現在は金融系SIプロジェクトを担当。PMP、SPM会員  
NARIISHI@jp.ibm.com