

ITモダナイゼーション Summit  
COBOLコンソーシアムセッション

# マイグレーション時に 考えなければならないこと・進め方 (簡易版)

2015年4月23日

NEC

クラウドプラットフォーム事業部

シニアエキスパート 飯島 裕一

# 本説明の前提と内容

「リホスト手法」によるマイグレーションを想定しています。

初めてマイグレーションを検討されている方を対象としています。

- COBOLさえあれば、オープン化できる？
- リホストだから、設計・評価は不要？
- マイグレーション(リホスト)のリスクは何？

マイグレーションを進める上での主要なポイントについて、Q&A形式にまとめています。

- Q1.マイグレーションを検討したいが、何からすべきか？
- Q2.効率の良いリホスト・プロジェクトの進め方は？
- Q3.効率のよい評価方法は？
- Q4.その他の懸案事項は？

# Q1. マイグレーションを検討したいが何からすべきか？

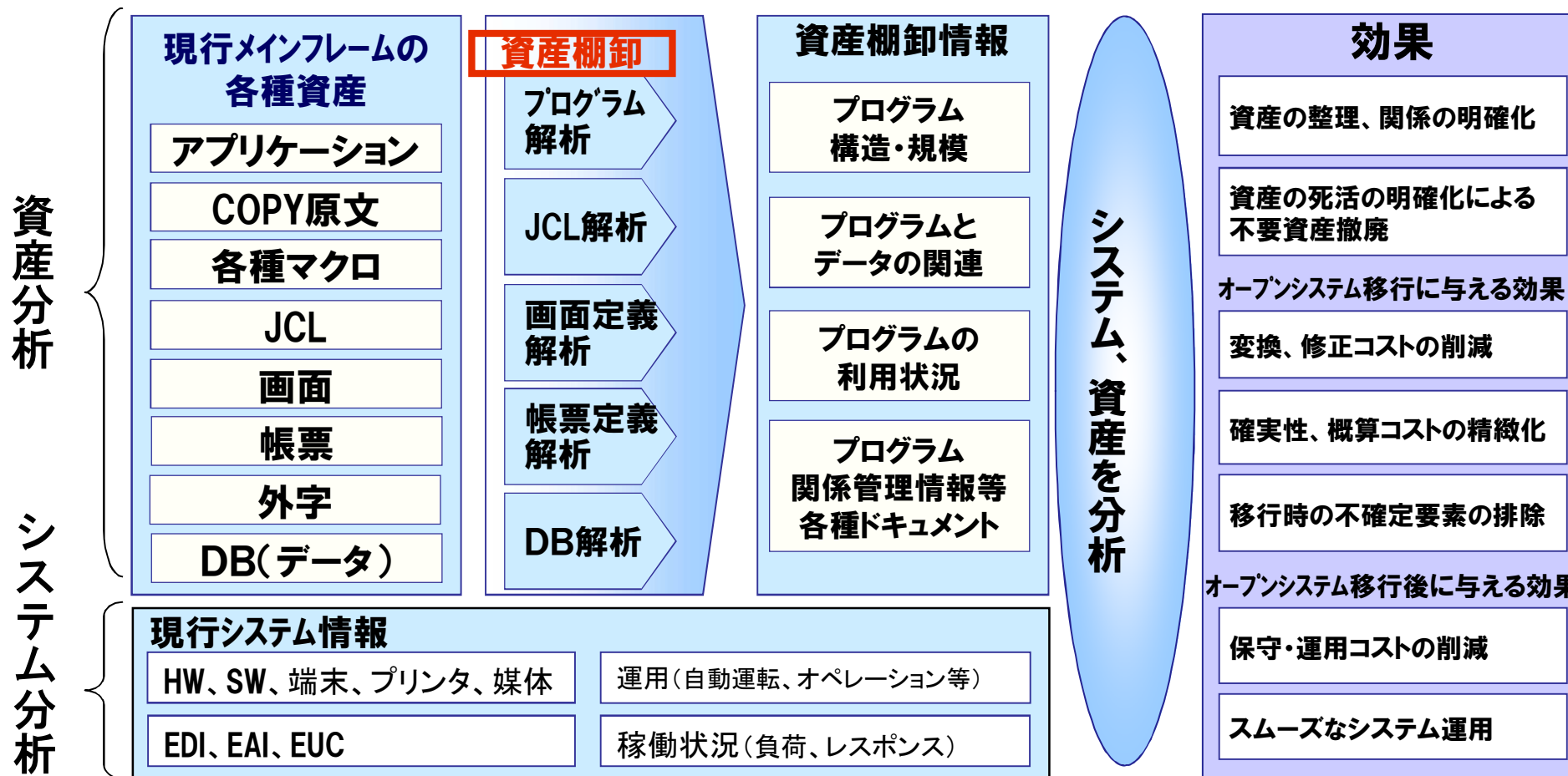
## A. 現行COBOL資産の棚卸から始めてください。

- どのマイグレーション手法を選択するにしても、現状のアプリケーション資産の量や種類を把握しておく必要があります。
- 資産はCOBOLプログラムだけではなく、帳票や画面などの各種定義体などの数量を把握します。

資産種別	対象資産
プログラム	COBOL、4GL (第四世代言語)、アセンブラ、業務簡易ツール、・・・ ※バッチ・オンラインそれぞれの規模の把握 ※DB・ファイルへのアクセス方式(READ/WRITE? SQL?)
JCL/JS	JCL、利用しているユーティリティ種別
帳票	フォーム定義体(バッチ帳票、オンライン帳票)
画面	画面定義体
ファイル/DB	表定義(リレーショナルDB)、スキーマ(ネットワーク型DB)、索引編成ファイル、相対編成ファイル、・・・
仕様書	上記対象資産の仕様書、および仕様書のメンテナンス状態

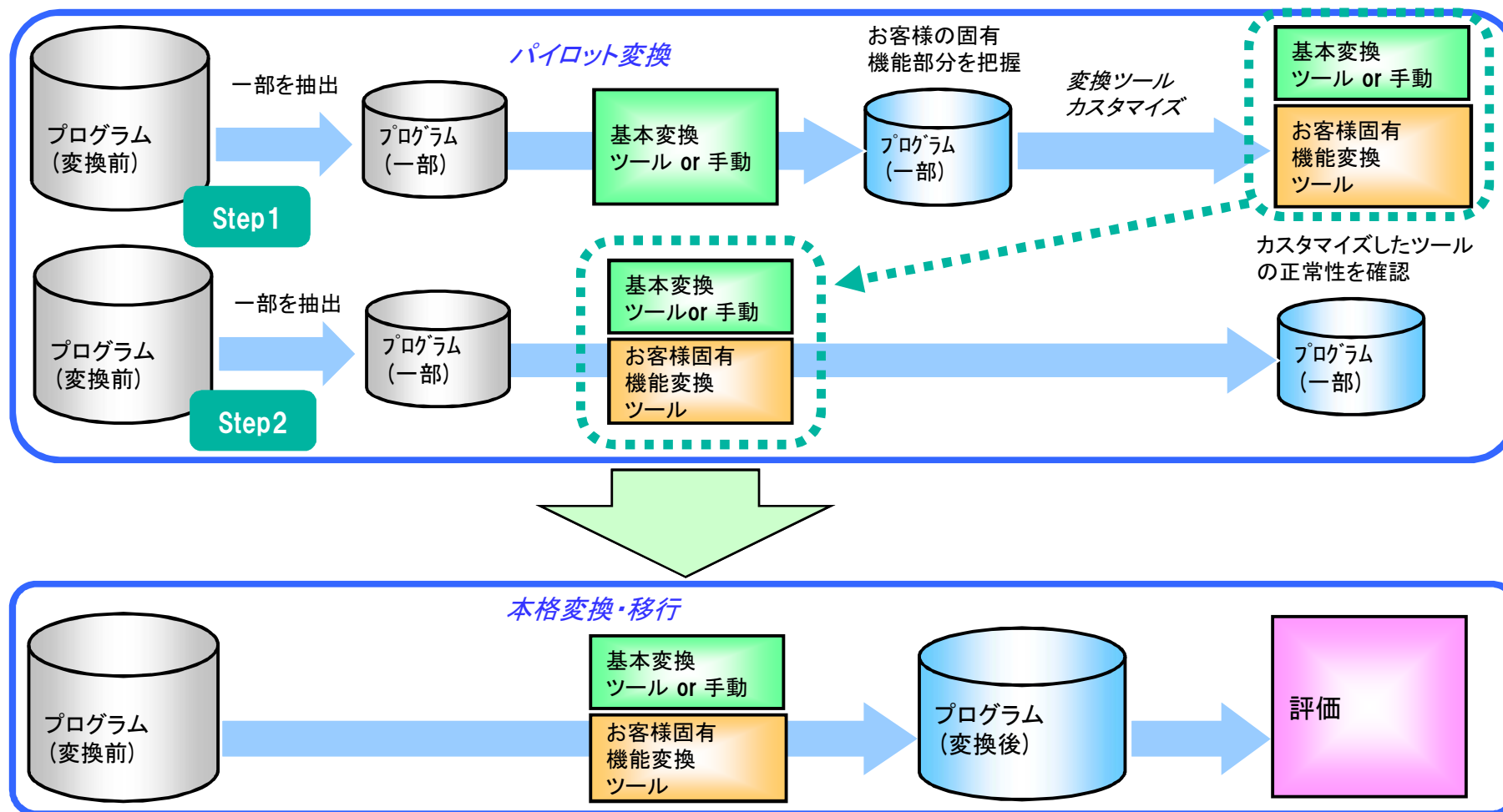
# (参考)資産棚卸の位置づけ

- 棚卸情報を活用し、メインフレーム・オフコン上の資産を整理
  - 保守・運用コストの削減、オープン化に際してのコスト/リスクを低減
- 外部インターフェースやインフラ、運用を含めたシステム全体の移行性を整理
  - 次期プラットフォームでどのように実現させるかを要件定義していく



## Q2. 効率の良いリHOST・プロジェクトの進め方は？

### A. 資産の一部をパイロット変換し、課題を把握・確認してから本格変換作業に入ります。



## パイロット移行を実施しない場合のリスク

- **移行手順が確立されていない**ため、本格変換作業が非効率的（移行手順を決めながらの作業となる）。
- **修正作業の洗い出しが不完全**なため、本来、ツールにより変換すべき作業が、結局手作業で実施される。
- 全体移行作業工数の**見積り精度が低い**まま。
- **動作検証未**であり、基本的な変換仕様ミスが、評価段階になってはじめて発覚する（変換ツールのバグを含む）。
- **製品のバージョン間の矛盾**などが評価段階になって発覚。

## Q3. 効率のよい評価方法は？

### A. 現システムとの並行評価をするのが効率的です。

■ テストデータ、テスト仕様に基づき、オンライン画面、画面レイアウト、バッチ帳票の出力結果のコンペアを行います。



## ■ 並行評価を行うために・・・

- **テスト仕様、テストデータの準備作業は、基本的に業務をよく分かっている方(通常はお客様)の担当となります。**
  - 基本的にお客様側での準備が必要となります。
  - コンペア結果でNGが出たとき、それが業務に影響するかどうかの判断も必要となります。
- **移行時に「機能追加・変更」はしないことを推奨します。**
  - 変更・追加すると、単純なコンペアが困難となります。
  - コンペア結果でNGが出たとき、それが、機能追加・変更によるものなのか、マイグレーション作業のミスなのかの切り分けに時間がかかります。



## Q4. その他の懸案事項は？

### A. リホストと言っても、システム的に変わる部分は多々あります。したがって、十分な事前設計が必要です。

OS	独自のインターフェース、サブルーチン、ユーティリティ、 <b>お客様限定対応等への対応</b>
言語の移行	ツールによる移行(手修正無しも可能だが、ツールのみでは移行後の保守に難) <b>アセンブラや移行対象外の言語の扱い</b> (C、COBOLで再作成) <b>ソースプログラム消失の場合</b> の対応(仕様書がなければ再作成困難)
オンライン	トランザクションタイプの違い(AP分割や再作成の可能性もある) トランザクション連携の実現( <b>複数台のサーバ間連携</b> など)
運用	統合運用環境の構築検討(メインフレーム並の運用は作り込みが必須) <b>コンソール応答</b> への対応 エンドユーザコンピューティング(EUC)の移行(エンドユーザの協力が必須)
データ移行	<b>MT、CGMT等の外部媒体の扱い</b> (オープン環境ではあまり利用されない) <b>NDB(構造型)からRDB(リレーショナル型)へ</b> 移行 コード変換(EBCDIC→SJIS) <b>マルチレイアウト</b> の対応
外部接続	社内システムとの連携(EAI)検討(通信手順、媒体、ミドルウェア等) 社外システムとの接続(EDI)検討(通信手順、媒体、ミドルウェア等)
帳票・画面	画面、フォームオーバーレイの移行(全く同じは難しい) <b>大量印刷</b> (どちらかといえばオープン系は苦手)、 <b>オンライン印刷</b>
外字	<b>外字登録数</b> (日本語EUC:X0208:940文字・X0212:1504文字、SJIS:3384文字) フォント移行
文字コード	コーティングシーケンス( <b>英数字、スペースのソート順変化</b> )

# \Orchestrating a brighter world

世界の想いを、未来へつなげる。

**未来に向かい、人が生きる、豊かに生きるために欠かせないもの。  
それは「安全」「安心」「効率」「公平」という価値が実現された社会です。**

**NECは、ネットワーク技術とコンピューティング技術をあわせ持つ類のないインテグレーターとして  
リーダーシップを発揮し、卓越した技術とさまざまな知見やアイデアを融合することで、  
世界の国々や地域の人々と協奏しながら、  
明るく希望に満ちた暮らしと社会を実現し、未来につなげていきます。**

Empowered by Innovation

**NEC**