第20回 COBOLコンソーシアムセミナー

クラウド時代。「COBOL活用のすゝめ」

マイグレーション技術を適用した自治体クラウド開発実証事例

SYSTEM'S

次の進化へ、システムズができること

株式会社システムズ

To the next evolution

2011年 4月15日

株式会社システムズ

マイグレーション事業本部

中本 周志／高尾 勇次

目 次

１．自治体クラウド開発実証事業（北海道）事例概要

1　お客様のご紹介

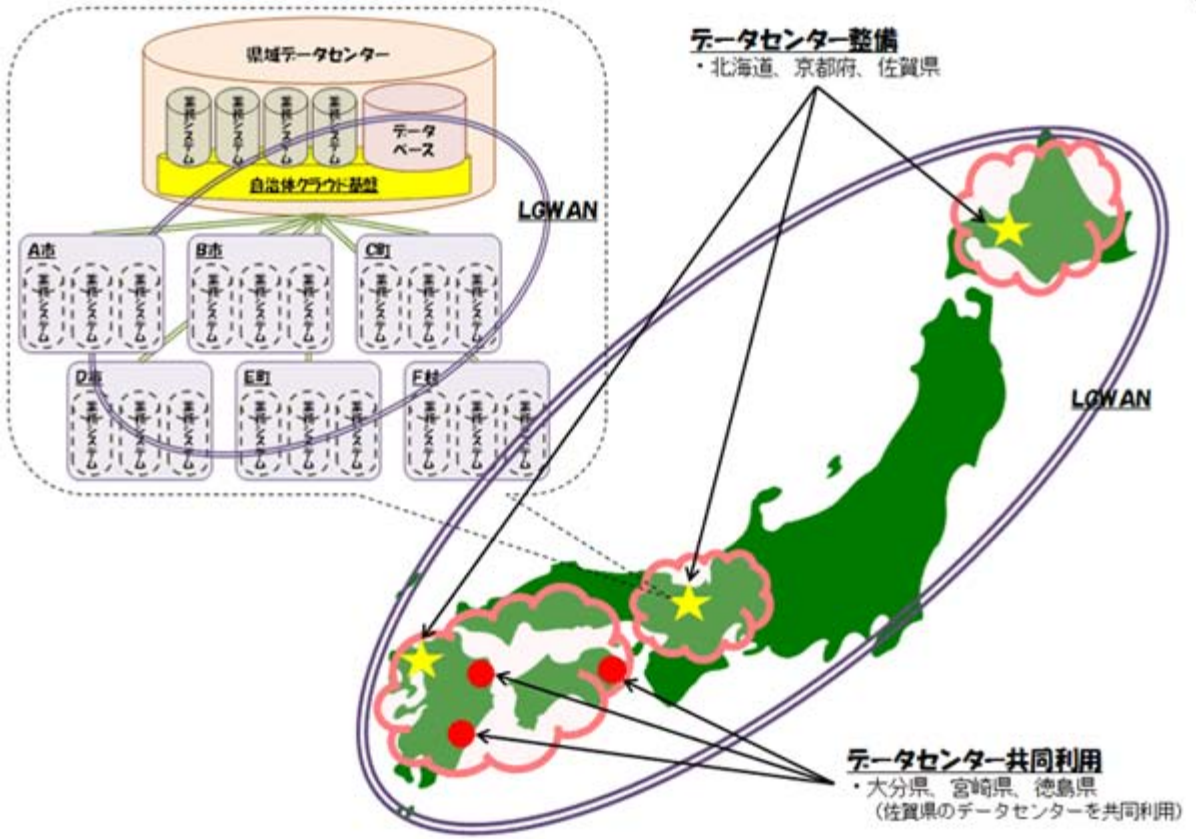
北海道
●所在地
北海道札幌市中央区北3条西6丁目
http://www.pref.hokkaido.lg.jp/
北海道は日本の総面積の約20%超を占める約83,400平方キロメートルの面積を持ち、179の市町村が存在します。これまでも、住民サービスの向上、行政の効率化、高度化、地域経済の活性化のために、道と市町村が協力しながら北海道独自の共同アウトソーシング構想である「北海道電子自治体プラットフォーム構想」(HARP構想)を推進してきました。
今回の総務省の『自治体クラウド開発実証事業』に積極的に参画することにより、最新のクラウドコンピューティングの技術やビジネスモデルを習得し、今後のHARP構想に反映させながら、いっそう効率的な次世代型の電子自治体の実現をめざそうとしています。

株式会社HARP
●本社所在地
北海道札幌市中央区北1条西6丁目1-2
アーバンネット札幌ビル3階
http://www.e-harp.jp/
■設立:2004年9月21日
北海道の電子自治体専門の第三セクターとして設立。住民サービスの向上や行政運営の効率化、さらには地域経済の活性化を支える電子自治体を官民協働で創り上げていくためのシステムを利用する行政側とシステムを開発する民間側をつなぐ、公的性格と民間のノウハウを併せ持つ事業体。
「自治体クラウド開発実証事業(北海道)」を北海道から受託し、中心となって推進しています。

2－1. プロジェクトの背景と目的

自治体クラウドとは、近年さまざまな分野で活用が進んでいるクラウドコンピューティングを電子自治体の基盤構築にも活用していこうとするものです。
総務省では、平成21年度から自治体クラウド開発実証事業に取り組んでいるところです。この事業は、地方公共団体の情報システムをデータセンターに集約し、市町村がこれを共同利用することにより、情報システムの効率的な構築と運用を実現するための実証実験です。
この自治体クラウド開発実証事業には6道府県66 市町村が参加していますが、今後は、今回の実証実験の結果をもとに、特に財政規模の小さい地方公共団体において、このような情報システムの集約と共同利用を合わせた取組み通じ、効率的な電子自治体の基盤構築の実現、さらには地域を元気にする便利な行政サービスの提供に向けた取組みが期待されています。（総務省地域力創造グループ地域情報政策室）

自治体クラウドとは

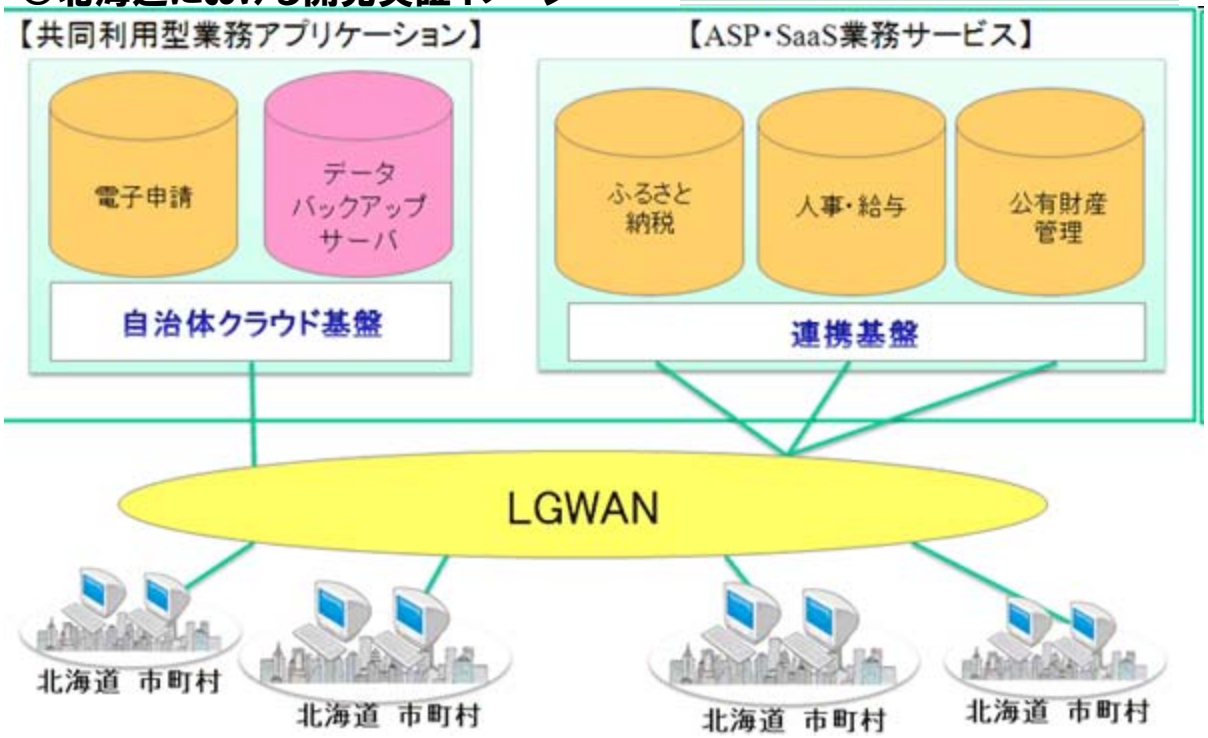


2－2. プロジェクトの背景と目的

「自治体クラウド開発実証事業（北海道）」の舞台となる北海道では、この事業を(株)HARP殿へ委託しており、北海道における開発実証項目となる、「データセンター機能実証」「データセンター間接続実証」「アプリケーション接続実証」の3つの開発実証実験を推進されている。

この度、北海道における開発実証項目のうち「アプリケーション接続実証」の基幹システムを含む多数業務のクラウド利用実証において、マイグレーション技術を適用した移行実証実験を検討されており、弊社のマイグレーション技術を活用した実証事業参加の機会を頂きました。

○北海道における開発実証イメージ

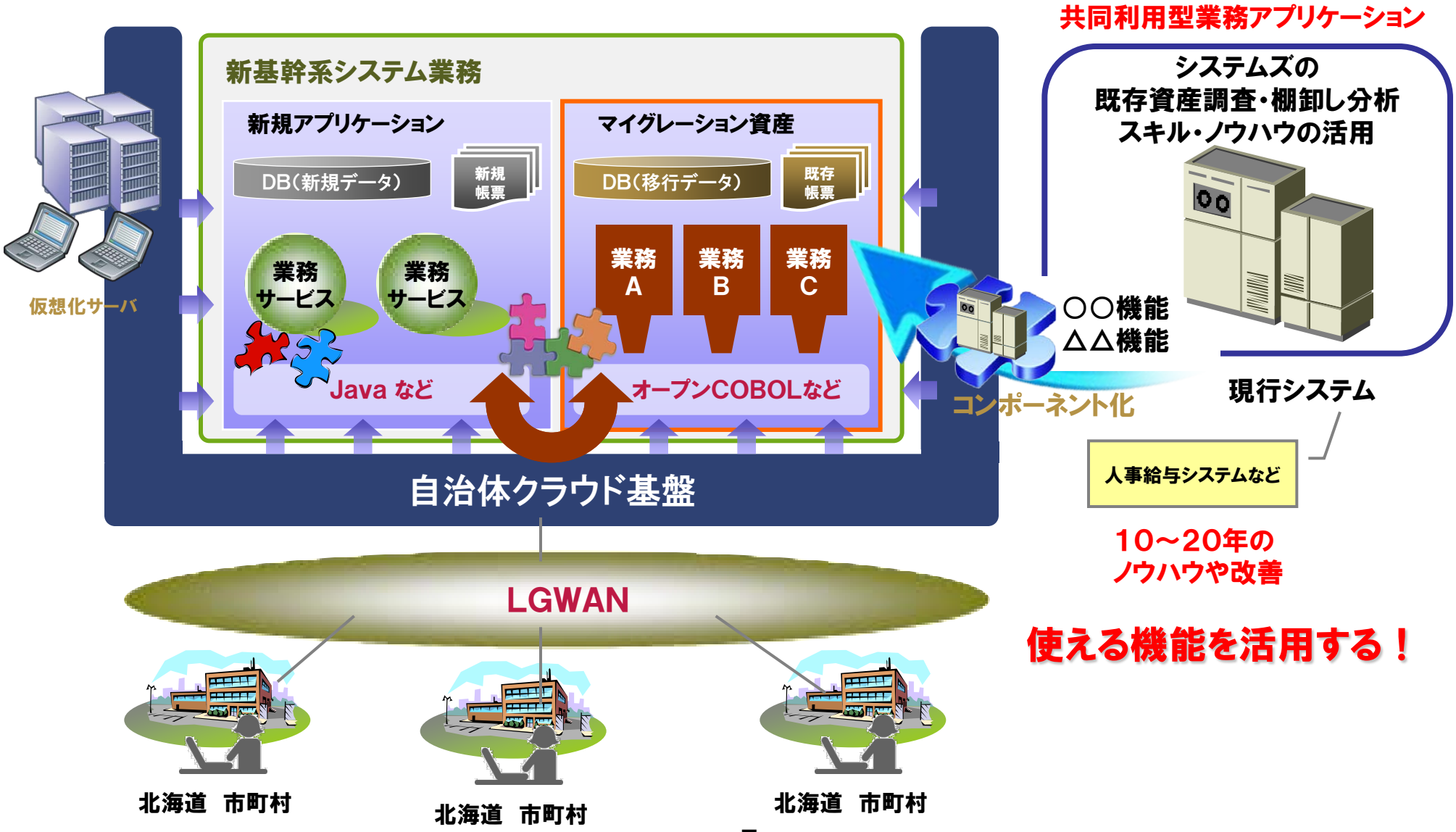


○実施項目

開発実証団体	大項目	小項目
北海道	データセンター機能実証	オンサイトバックアップ
		オフサイトバックアップ
		仮想化効果実証（障害切替え）※
		仮想化効果実証（サーバ台数減少）
		仮想化効果実証（拡張性）
	データセンター間接続実証	データセンター間接続
	アプリケーション接続実証	共同利用型業務アプリケーション接続
		ASP・SaaS業務サービス接続
		新規自治体の参加実証
		基幹システムを含む多数業務のクラウド利用実証

3. マイグレーション開発実証 適用イメージ

■ マイグレーションの適用による、基幹システム・多数業務のクラウド利用実証イメージ



4－1. プロジェクト範囲と概要（実証すべき範囲）

■ 弊社作業範囲

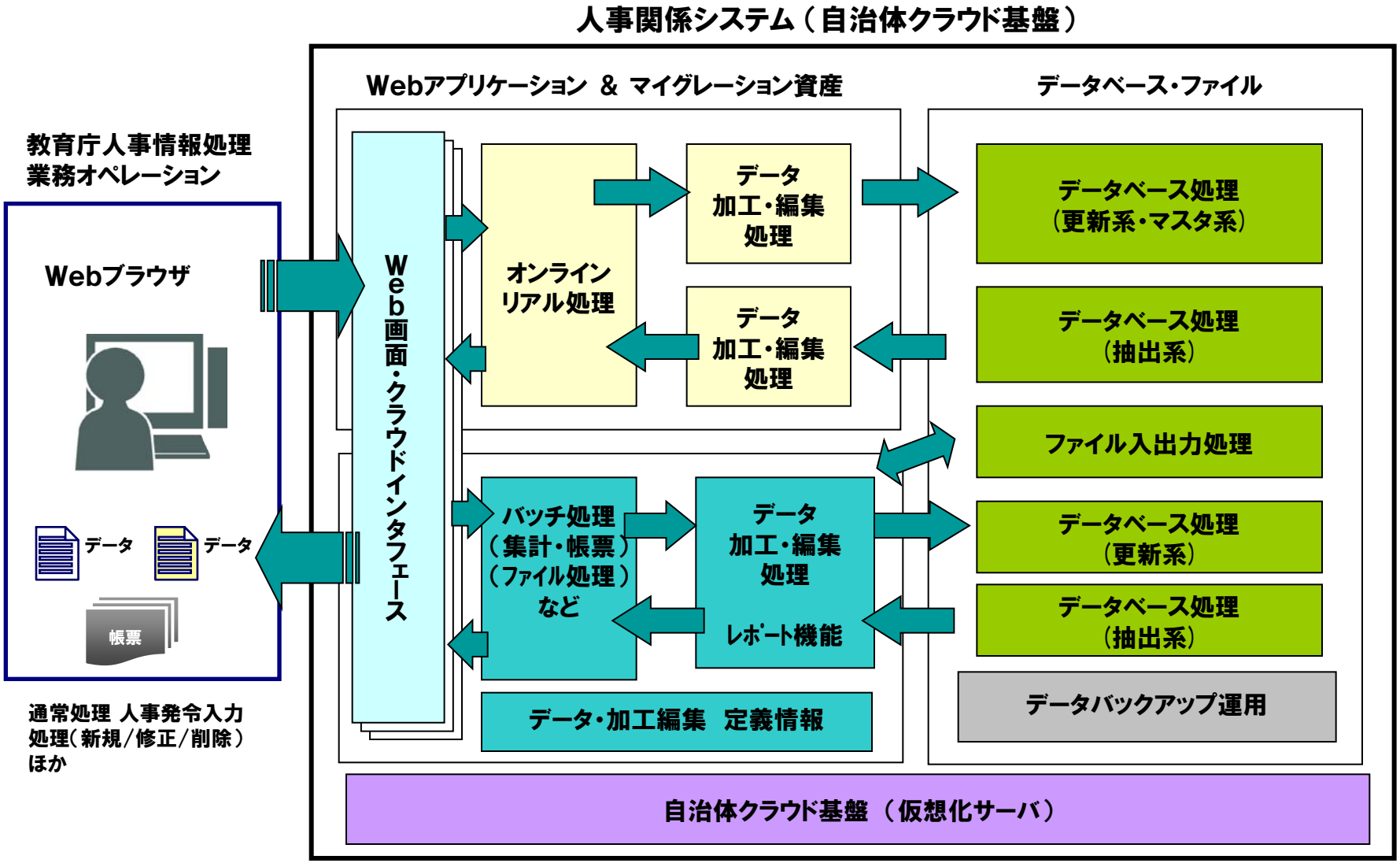
□ 対象業務システムは、道教育委員会の人事給与システムへのマイグレーション適用
□ 大型汎用環境からクラウド環境に移行する際の方法論と課題抽出の支援
□ 北海道自治体クラウド基盤上での開発（マイグレーション）と動作検証・評価

■ 作業概要（実証すべき範囲）

□ 教育庁人事情報処理
　－ 通常処理 人事発令入力処理（新規/修正/削除） ほか
□ マイグレーションの有用性を見出せる部分の棚卸し、評価
　－ オンライン画面機能のみならず、バッチ処理や帳票出力などの機能検証まで
　－ 既存リソース（プログラム、JCL、環境定義など）の活用対象の調査・分析
□ 既存システム資産の移行性（ポータビリティ）の実証
　－ 既存リソースの移行性検証と変換、移行

4－2. プロジェクト範囲と概要（実証すべき範囲）

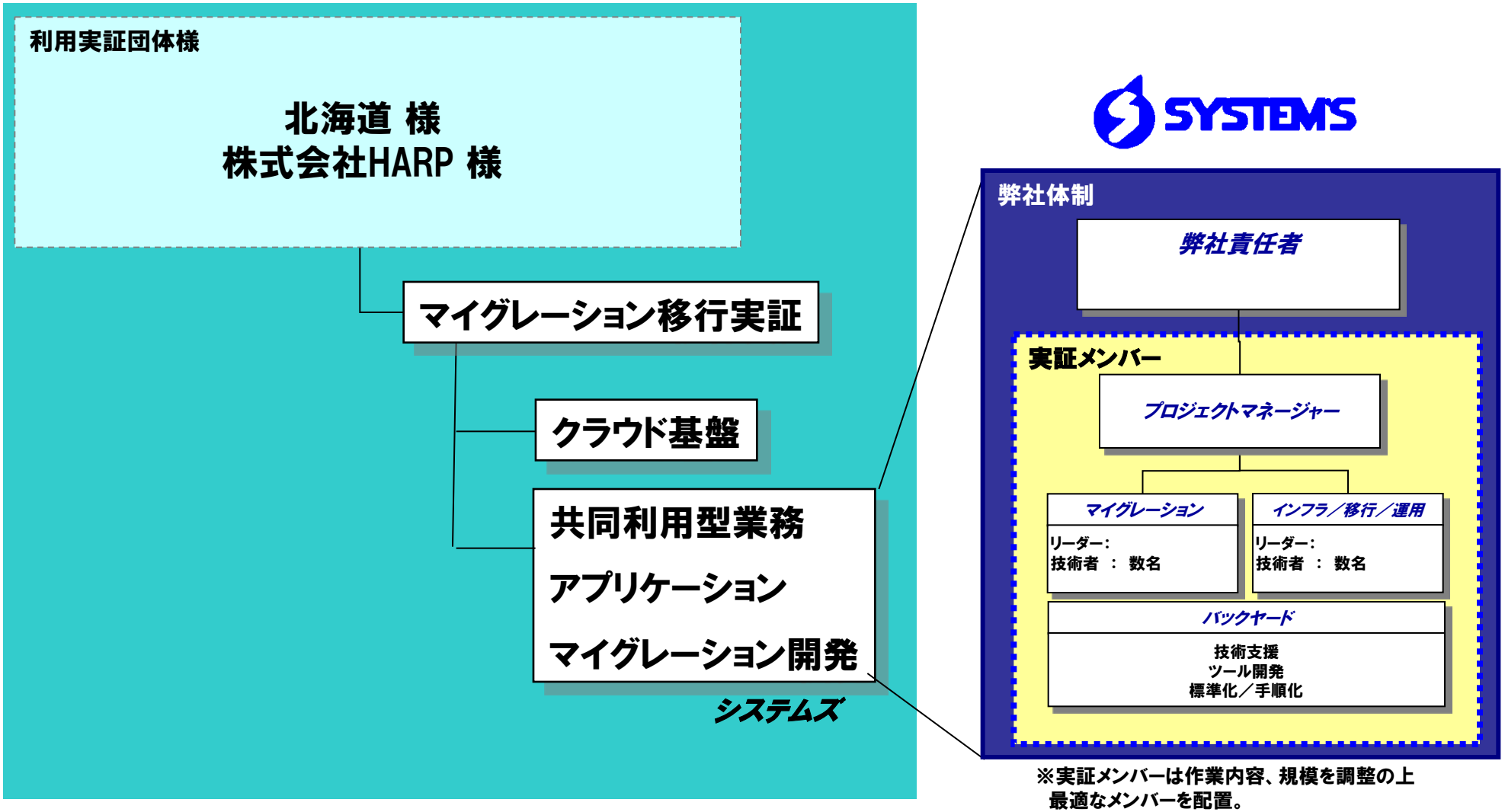
■ 作業概要（実証すべき範囲）・処理イメージ



5.　推進体制

■ 推進体制

マイグレーション移行実証の推進体制を以下に示します。

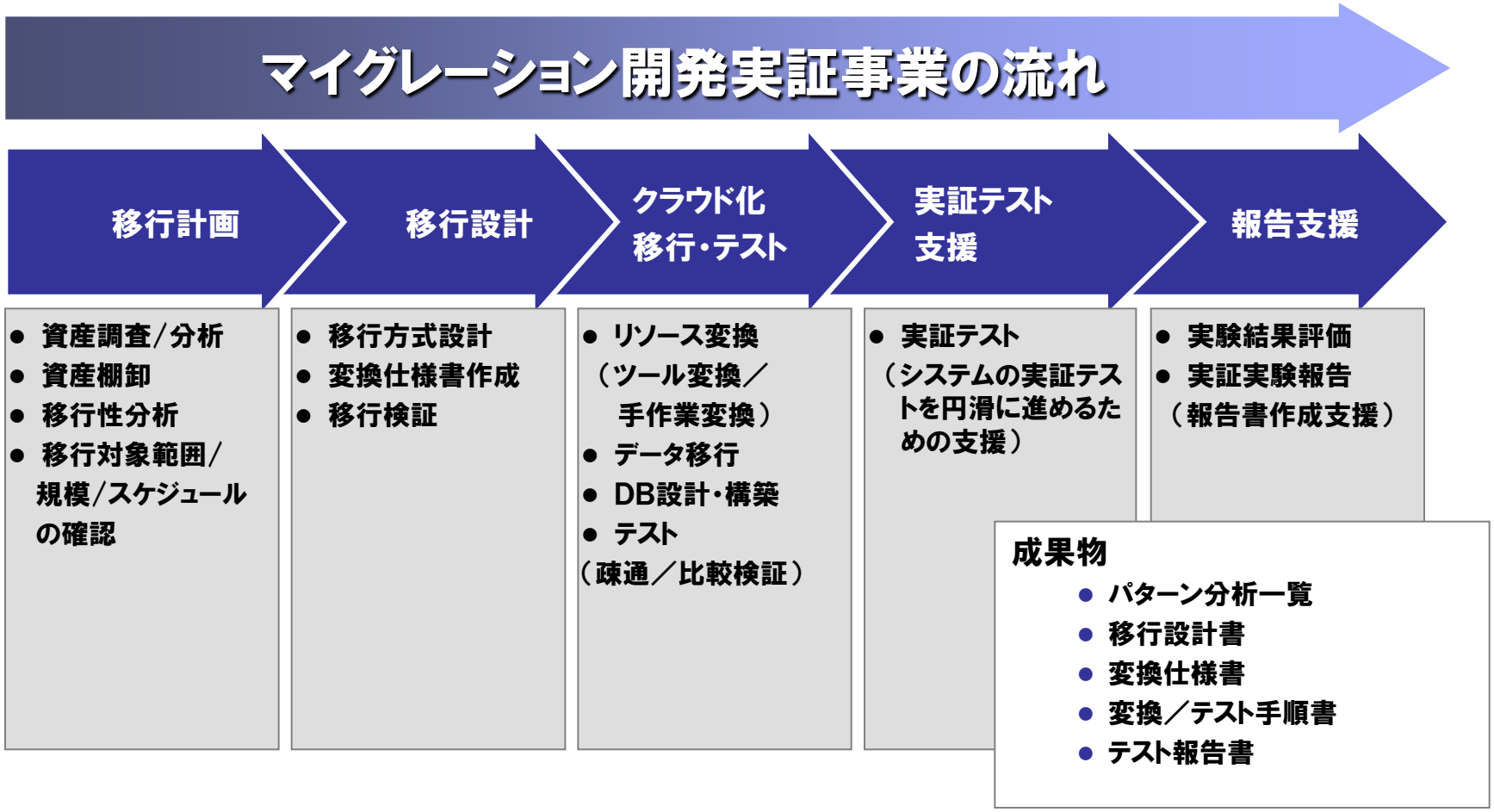


※実証メンバーは作業内容、規模を調整の上
最適なメンバーを配置。

6. マイグレーション開発実証事業・タスクフロー

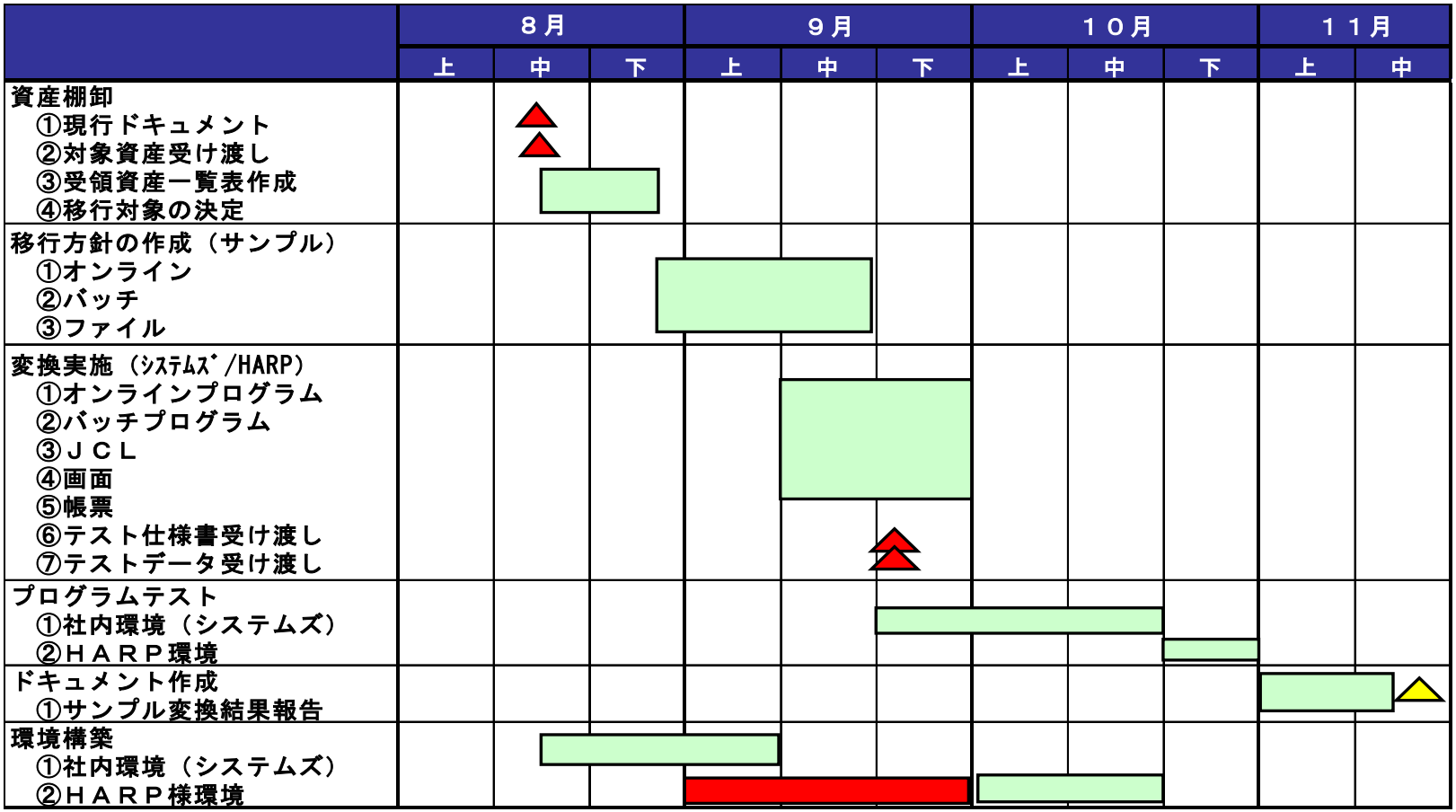
■ マイグレーション・タスクフローの適用

対象システム：人事給与システムの北海道・自治体クラウド基盤へのマイグレーション
大型汎用機の環境からクラウド環境へ移行する際の方法論や課題抽出、自治体クラウド基盤へのマイグレーションおよび動作・検証、報告までを実施。



７．マイグレーション実施スケジュール

■ プロジェクト全体スケジュール



2．プロジェクト詳細

1. 実証事業 対象システム概要

本システムは、教育庁人事情報処理システムの通常処理、人事発令入力処理（新採、転任、退職）の機能を有する。本プロジェクトにおいて、大型汎用機環境のシステムを自治体クラウド環境上にマイグレーション可能であることを検証。

ソフトウエア構成（前提条件）

#	分類	移行元	移行先
1	OS	NEC ACOS-4	Windows2008/VmWare
2	ファイル	VSAS、SEQ	ISAM、SAM
3	オンライン制御	VIS	WebsphereApplicationServer
4	言語	COBOL85	MF-COBOL
5	画面定義	MFDL	JSP/Servlet
6	書式	FORMEX	SVF
7	ＪＣＬ	JCL	DOS-Batch

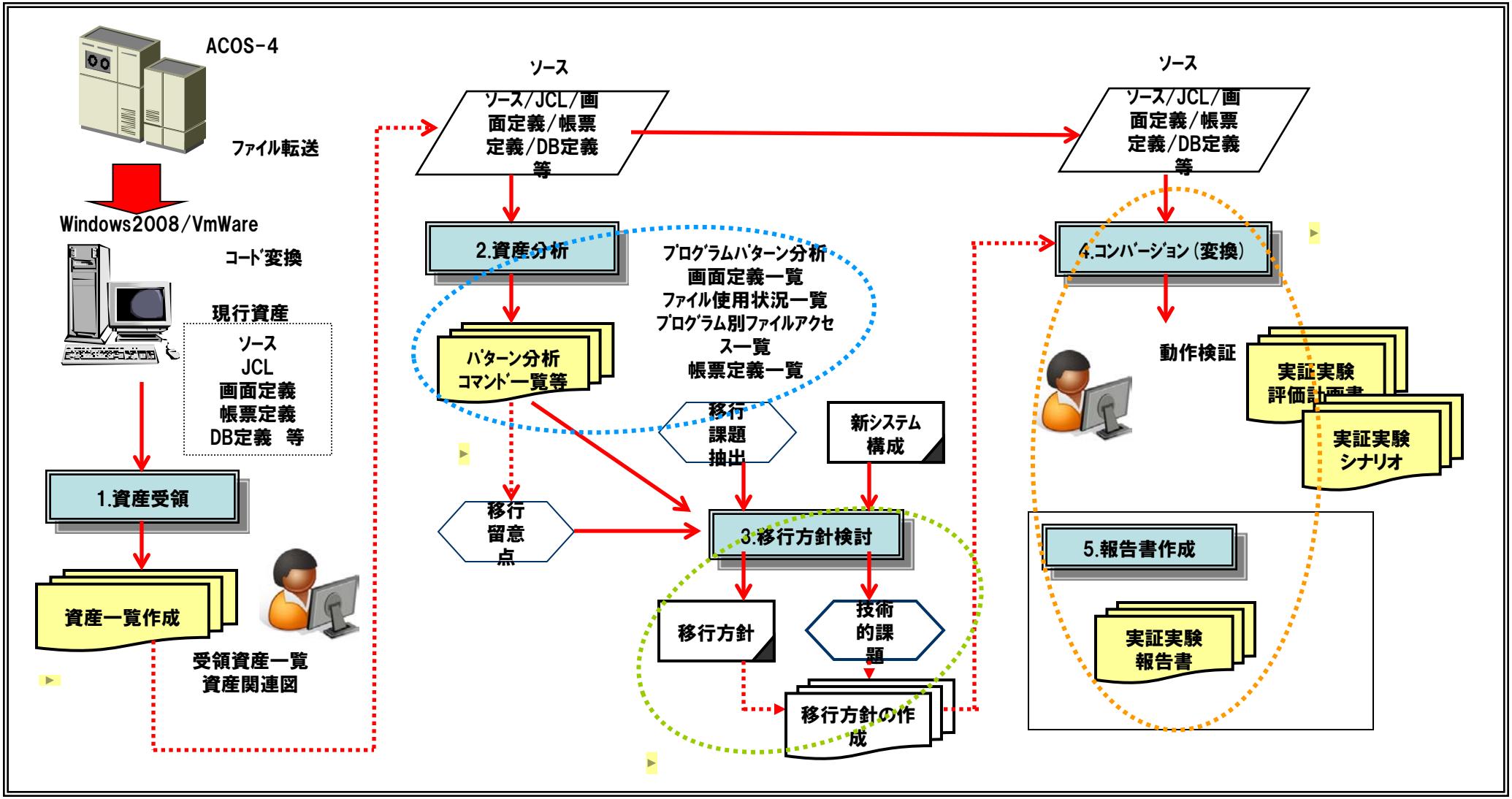
●移行先は実証実験環境上で稼動

#	分類	実証実験環境
1	仮想環境	VMWare　ESXi4.0
2	ＯＳ	Windows2008
3	ＣＰＵ	ｲﾝﾃﾙXeonﾌﾟﾛｾｯｻｰE5530（2.40GHz）
4	ＨＤＤ	10GB
5	メモリ	4GB

[想定容量]　合計　50GB
・WAS　1 GB
・SVF　1 GB
・RAD　3.5GB
・Windows2008　40GB

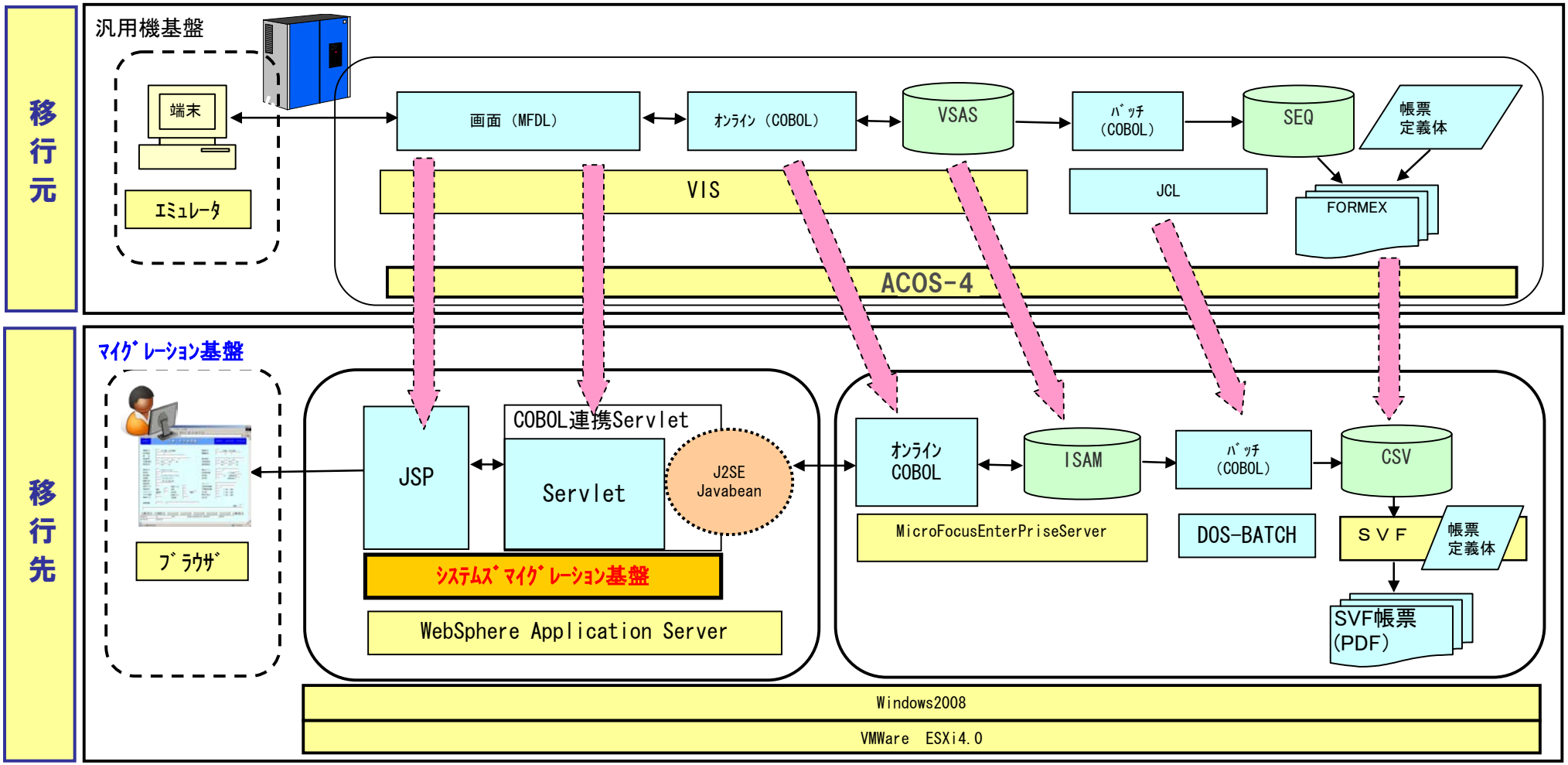
2. 実証事業 システム移行シナリオ

■当社独自の移行手順に基づくシナリオ化



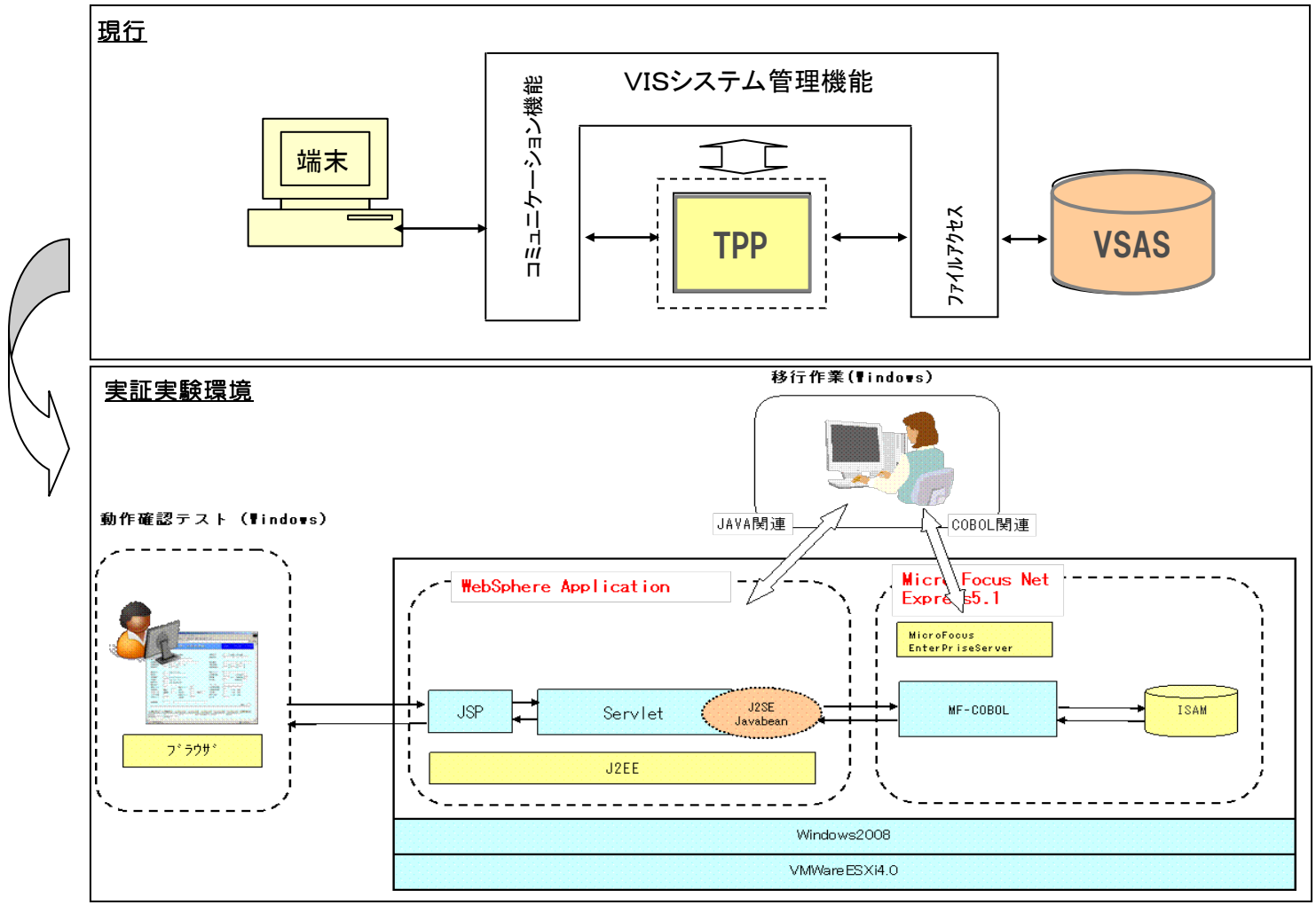
3. 実証事業 システム構成

■実証事業における、移行前／移行後でのシステム構成



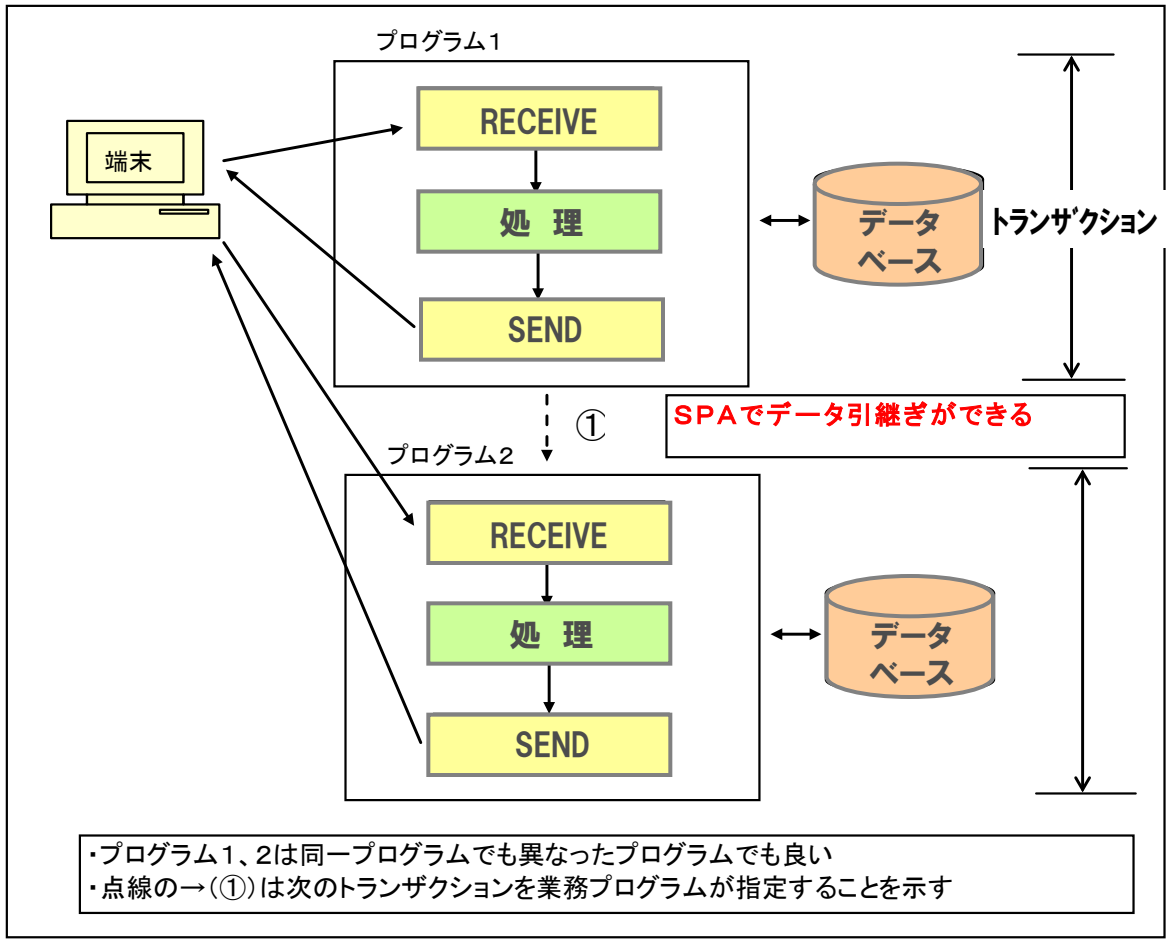
4－1. 実証事業　システム移行方針～オンライン

■移行方針 ～ オンラインシステム構成
・今回のマイグレーション実証環境のシステム構成は、以下の通り。



4－2. 実証事業　システム移行方針～オンライン

■移行方針 ～ 現行オンライン方式
形態：会話型のメッセージ処理形態
会話型のメッセージ処理形態とは、端末と業務処理プログラムがメッセージの送受信を
複数回繰り返すことによって、一つのまとまった業務処理を行なうメッセージ処理形態となります。



＜現行方式の特徴＞

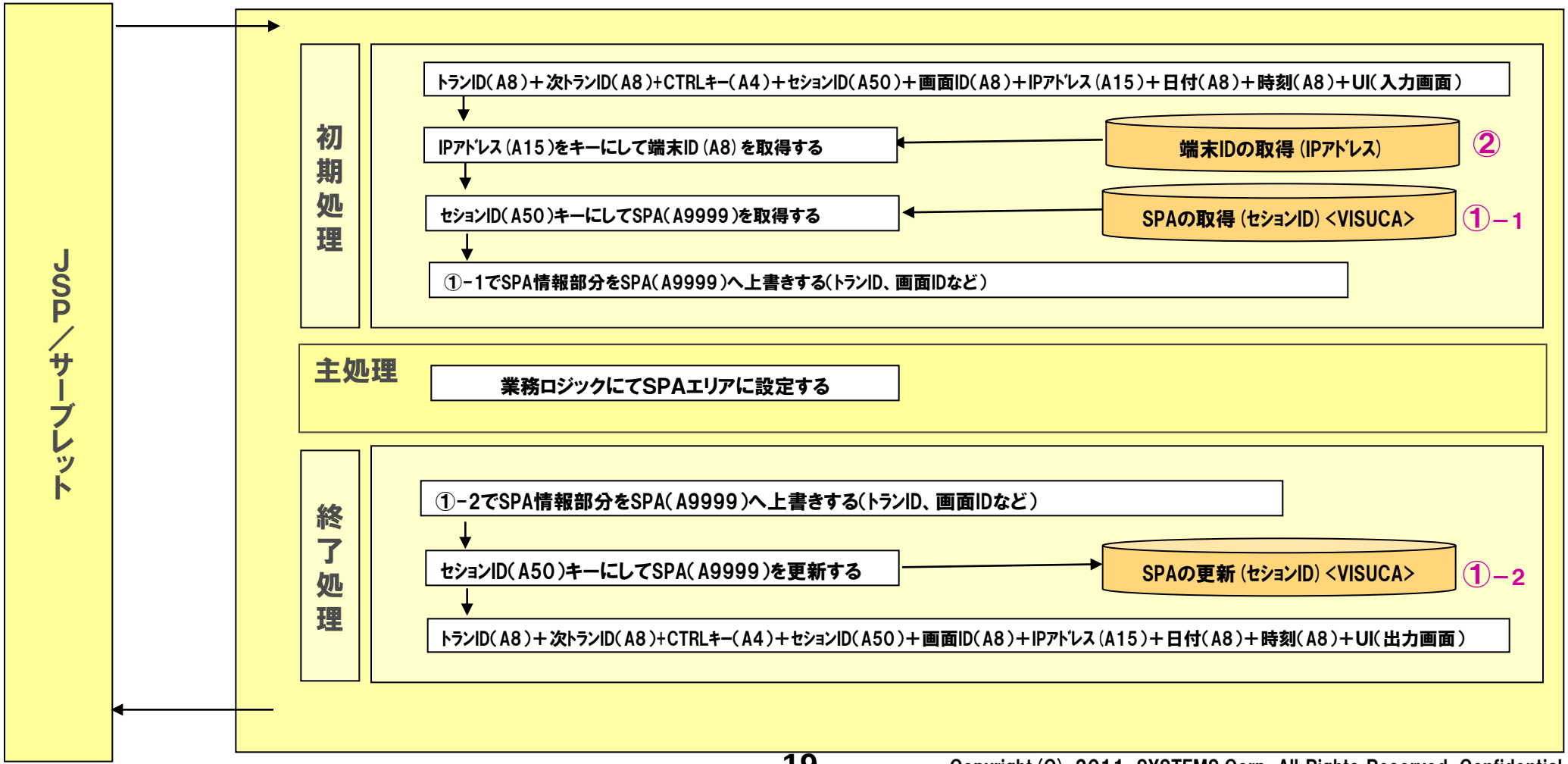
①SPA（一時記憶領域）の使用あり
②端末IDでアクセス制御有り
③トランザクションコードによる起動
④1プログラム複数画面有り（再定義有り）
⑤画面の動的変更処理有り
⑥PFキー操作有り
⑦マクロの使用有り
⑧画面のWeb化

4－3. 実証事業　システム移行方針～オンライン

■移行方針 ～ オンライン方式

①ＳＰＡ（一時記録エリア）は、セションＩＤをキーとしてＩＳＡＭデータとして保持。起動時に作成、終了時に削除します

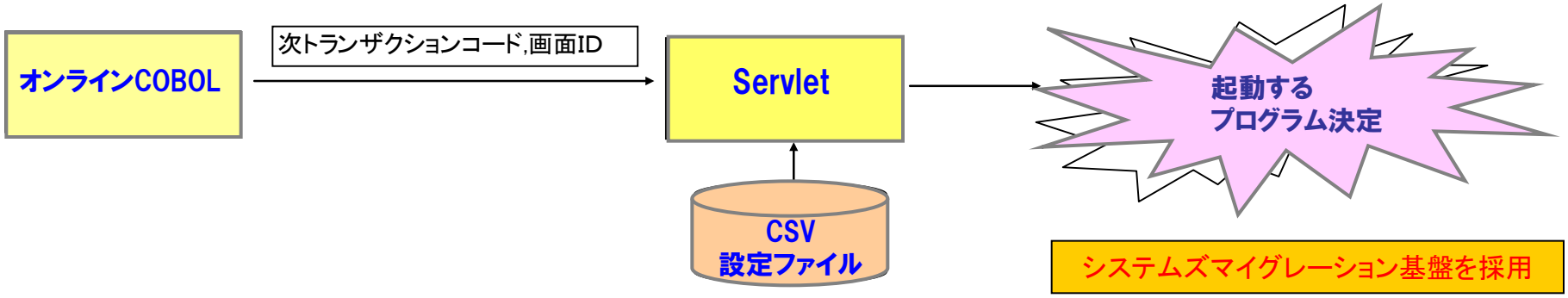
②端末ＩＤは「ＩＰアドレス」、「コンピュータ名」などローカルＰＣ情報を取得し、サーバ側にて「端末ＩＤ」に変換します



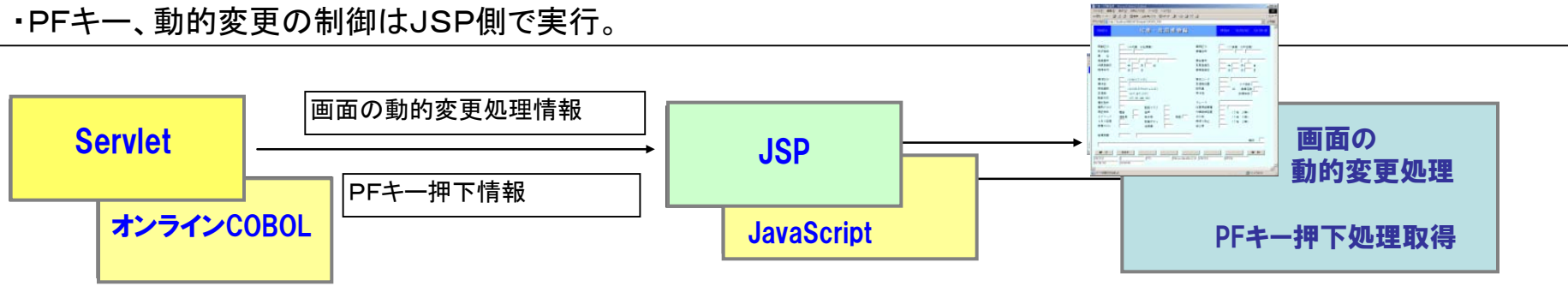
4－3. 実証事業　システム移行方針～オンライン

■移行方針 ～ オンライン方式　COBOL・JAVA連携方法

トランザクションコードによる起動
1プログラム 複数画面有り（再定義有り）
・トランザクションコードと画面IDにて、次に起動するプログラムを決定する。



画面動的変更処理有り
PFキー操作有り
・PFキー、動的変更の制御はJSP側で実行。



※対象ソースでは、画面動的変更は、2種類で実行。
a.768 ・・・ エラー（赤／ブリンク）※分散環境ではブリンクができず。（リバースへ変更する。）
b.7168 ・・・ 入力制御（保護）

システムズマイグレーション基盤を採用

4－4. 実証事業　システム移行方針～オンライン

■移行方針 ～ オンライン方式　COBOL・JAVA連携方法

【前提】
COBOLのリンケージセクションで宣言されている情報から、IF部分を自動生成する。

【修正前】

現行COBOLで

```
04 WK-TOROKU-NO              PIC  X(11).
04 REDEFINES WK-TOROKU-NO.
  05 WK-TOROKU-NO1           PIC  X(03).
  05 WK-TOROKU-NO2           PIC  X(03).
  05 WK-TOROKU-NO3           PIC  X(01).
  05 WK-TOROKU-NO4           PIC  X(04).
```

となっています

JAVAでのIF項目は、

```
private java.lang.String wk_toroku_no_io = EMPTY_STRING;
```

と、COBOLのRedefine宣言は無視され、1つの変数になってしまいます。

→画面に設定する場合（COBOLから戻ってきた場合）や、
画面からCOBOLに引き渡す場合に、分解・結合をJAVAで行う必要性が生じる

【修正後】

なので

```
04 WK-TOROKU-NO-R.
  05 WK-TOROKU-NO1           PIC  X(03).
  05 WK-TOROKU-NO2           PIC  X(03).
  05 WK-TOROKU-NO3           PIC  X(01).
  05 WK-TOROKU-NO4           PIC  X(04).
04 REDEFINES WK-TOROKU-NO-R.
  05 WK-TOROKU-NO            PIC  X(11).
```

とすることで

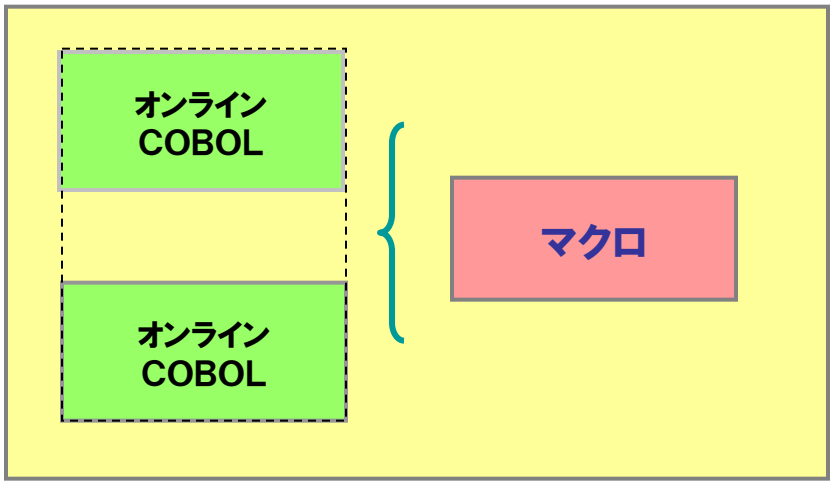
```
public class Wk_toroku_no_r_io implements com.microfocus.cobol.lang.CustomRecord {

    private java.lang.String wk_toroku_no1_io = EMPTY_STRING;
    private java.lang.String wk_toroku_no2_io = EMPTY_STRING;
    private java.lang.String wk_toroku_no3_io = EMPTY_STRING;
    private java.lang.String wk_toroku_no4_io = EMPTY_STRING;

        ～

}
```

となり、画面項目と1対1の関係になるため、JAVA側での作りこみが不要になる。

4－4. 実証事業　システム移行方針～オンライン

■移行方針 ～ オンライン方式

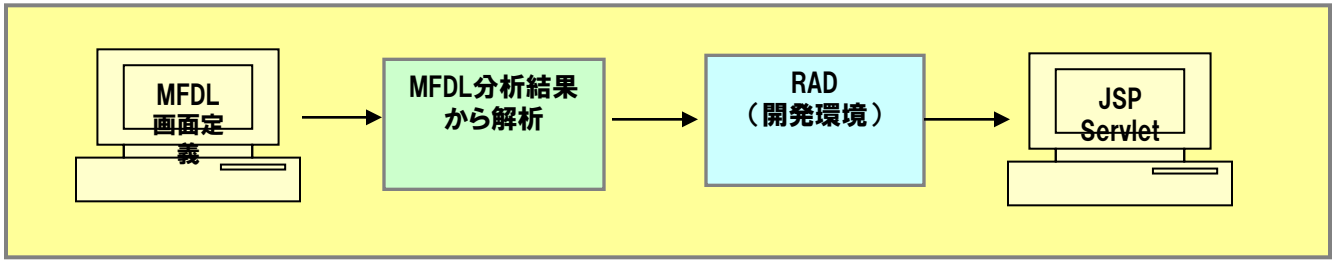
① マクロの使用 有り
・マクロをパターン別に展開して、オンラインCOBOLに挿入します。



＜マクロタイプ別＞

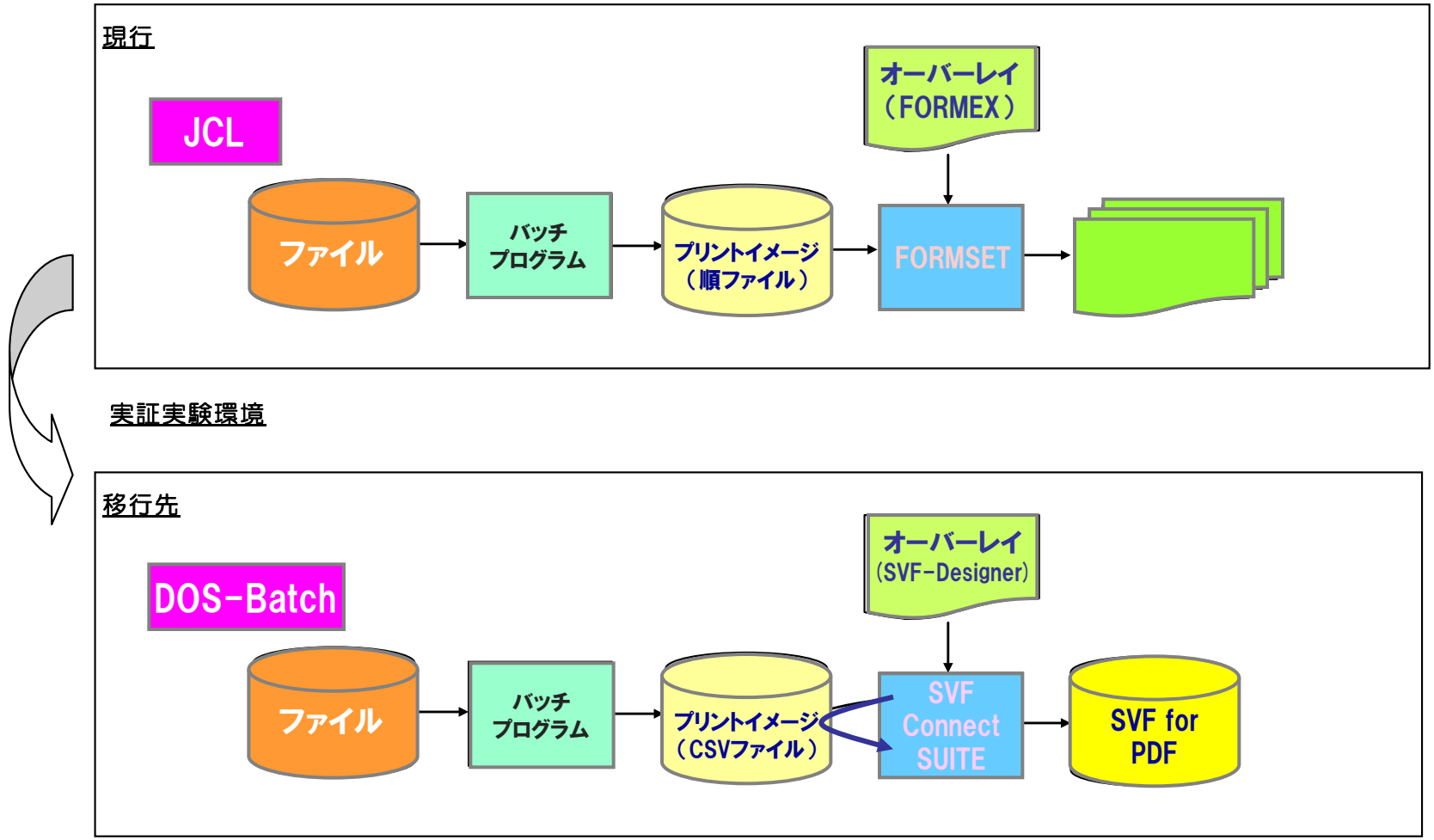
a. 引数無し
b. 引数有り（固定）
c. 引数有り（自マクロ内のみで使用）
d. 引数有り（呼出し元で内容変化パターン）

② 画面のWeb化
・MFDL画面を分析し、JSP/Servletを新規開発します。



4－5. 実証事業　システム移行方針 ～ バッチ

■移行方針 ～ バッチシステムより
帳票システムを、電子帳票化（PDF出力）。今回の実証環境のシステムは以下の通り。

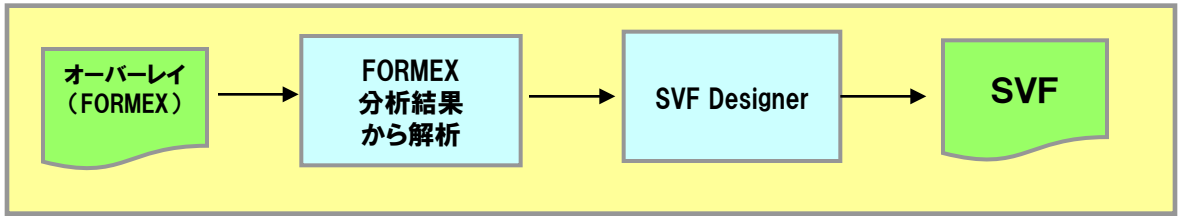


4－5. 実証事業　システム移行方針 ～ バッチ

■移行方針 ～ バッチシステムより帳票システムを、電子帳票化（PDF出力）。その移行方法。

① 帳票定義体の移行

・FORMEX定義体の書式フォーマットを分析した結果をもとに、SVF Designerにて新規作成します。



② プログラムの移行

・SVF Designerで定義したフォーマットに併せたＣＳＶファイルを出力するように修正します。

1.CSVファイルの出力方法（カンマ区切りで出力）
　1レコード目　　・・・ 項目名称を出力します。
　2レコード目以降 ・・・ 項目名称に合わせたデータ項目をセットします。
2.CSVファイルの出力形式
　「見出し行」＋「明細行」を1レコードとして出力します。
3.SVF定義体とCSVファイルの結合は、SVF Designerで定義します。
　CSVファイルの1レコード目の項目名称をSVF定義の項目と連携させる方式とします。

③ CSVファイルと帳票のマッピング

・プログラムで出力したＣＳＶファイルを「SVF-Connect　SUITE」でポーリング受信しＳＶＦ帳票定義体とマッピングを行ないます。また、マッピングを行なった帳票は、「SVF　for　PDF」にてＰＤＦ化して特定のディレクトリ配下に格納します。

4－6. 実証事業 システム移行方針～プログラム言語

■移行方針 ～ プログラム言語

COBOL85を、MF－COBOLへ移行（COBOL to COBOL）

・パターン分析による調査、分析、評価。

項番	命令語	タイプ	ﾊﾟﾀｰﾝ数	使用回数	変換観点
1	IDENTIFICATION	ID	1	21	－
2	PROGRAM-ID	ID	1	21	－
3	AUTHOR	ID	1	7	なし。コメントアウト
4	ENVIRONMENT	EN	1	21	－
5	CONFIGURATION	EN	1	21	－
6	SOURCE-COMPUTER	EN	1	21	コメントアウト
7	OBJECT-COMPUTER	EN	1	21	コメントアウト
8	INPUT-OUTPUT	EN	1	12	－
9	FILE-CONTROL	EN	1	12	－
10	SELECT	EN	8	17	SVAS→ISAM、帳票ファイル→CSVに変更
11	I-O-CONTROL	EN	1	1	APPLYはコメントアウト。CSVﾌｧｲﾙのため
12	DATA	DA	1	21	－
13	FILE	DA	1	12	－
14	WORKING-STORAGE	DA	1	21	－
15	LINKAGE	DA	1	20	－
16	FD	DA	4	16	DEPENDING ONはコメントアウト
17	SD	DA	1	1	－
18	01	DA	4	267	VALUEのNC"漢字"のCは削除.
19	77	DA	4	24	－
20	88	DA	1	1	－
	CD	DA	1	1	コメントアウト、PROCEDU
	OPY	DA	1	323	－
	UNICATION	DA	1	7	通信はコメントアウト
		DA	1	11	なし。コメントアウ
		PR	8	21	引数を

項番	命令語	タイプ	ﾊﾟﾀｰﾝ数	使用回数	変換観点
26	OPEN	PR	3	4	－
27	CLOSE	PR	2	4	－
28	READ	PR	10	19	－
29	END-READ	PR	1	13	－
30	WRITE	PR	6	10	帳票→CSVに変更
31	END-WRITE	PR	1	1	－
32	REWRITE	PR	2	4	－
33	END-REWRITE	PR	1	2	－
34	START	PR	4	9	－
35	END-START	PR	1	3	－
36	DELETE	PR	1	2	－
37	ACCEPT	PR	1	2	－
38	DISPLAY	PR	11	70	CONSOLE→SYSLST（SYSOUT）、NC"漢字"のNCは削除
39	COMPUTE	PR	9	13	－
40	ADD	PR	1	79	－
41	SUBTRACT	PR	1	4	－
42	DIVIDE	PR	1	3	－
43	MOVE	PR	113	7405	NC"漢字"のNCは削除
44	INSPECT	PR	1	6	－
45	INITIALIZE	PR	9	266	－
46	SET	PR	1	16	－
47	SORT	PR	1	1	－
48	IF	PR	205	1560	－
49	ELSE	PR	2	501	－
50	END-IF	PR	1	368	－

5. 実証事業 プロジェクト進捗時課題と対応例

■ 課題

□ オンライン
①画面のSPA(一時記憶領域)の使用
②端末IDでアクセス制御している
③PFキー操作
④エミュレータに処理の異常終了時に「異常終了」と表示される
□データ
①資産提供がCGMT
□ クラウド化
①操作性(ブラウザ機能の実現)

■ 対策

□ オンライン
①画面のSPA(一時記憶領域)をISAMデータとして保持するよう代替機能追加
②端末IDに代わる「IPアドレス」・「コンピュータ名」等を端末IDに変換する処理を追加
③ファンクションキーの制御を処理を追加
④異常終了画面を作成
□ データ
①FDでの提供。(セキュリティ上、FTP等データの外部通信は未認可)
□ クラウド化
①ブラウザの種類、バージョン対応、制限が必要

3．まとめ

1. 実証事業 検証内容

マイグレーション環境

No	検証項目	マイグレーションの容易性
1	プログラム移行	COBOL to COBOLの移行、文字コードの機械的変換、シフトコードの利用状況から、全体的に移行性が非常に高い状況にあると言える
2	データベース移行	適用外
3	オンラインアクセス移行	画面については、手変換による変換が必要であるが、パターン化が可能であり移行性は高いと言える 帳票については、今回の方式では手変換の箇所が多く、移行性は低くなっている。
4	JCL移行	完全自動変換 移行性は高いと言える
5	データ移行	ツールによる移行が可能であり、移行性は比較的高いと言える

クラウド環境

No	検証項目	マイグレーションの容易性
1	ブラウザ 種類・バージョン	InternetExplorer　ver6以降に限定
2	ブラウザ機能（ファンクション・戻るボタン等）	ファンクション制御の適用
3	応答レスポンス	実証実験環境にて、各画面0.5秒以内の応答速度を確認 。
4	高負荷時レスポンス	適用外
5	セキュリティ	端末IDによる制御の実装

2. 今後に向けた課題、取り組み

本システムの検証において、大型汎用環境システムのクラウド環境への移行について、アプリケーションレベルでの移行では、機能面、性能面ともに問題なく移行が可能であると言えます。
ただし、今回は実証対象以外の分析を行なっていない為、今後は全体を対象にして、移行対象のシステムの資産棚卸し、分析、評価を実施していく必要があります。

自治体クラウド環境は、IaaSとしての側面が強くあります。将来的に各自治体間で共通のアプリケーションを利用するためには、実行基盤／開発基盤の統一を図り、PaaS ／SaaSクラウドへの移行を視野に入れて進める必要があります。
PaaS／SaaS環境への移行を進めるためには、アプリケーションレベルだけではなく、特に「運用面」においても検討を進める必要があると考えます。

特に下記に示す3点についての検討を実施する必要があるといえるでしょう。

①セキュリティの確保

PaaS／SaaS環境への移行を進める上で、セキュリティーの確保は避けては通れない課題となります。
マルチテナントへの対応等、大型汎用環境では専用端末としての利用から、公開利用への変化に対する運用設計の見直しが必要です。

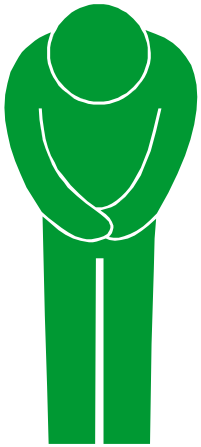
②認証・課金対応

大型汎用環境のシステムの専用端末からのアクセスとは異なり、公開環境へのクライアントからのアクセスとなるため利用者の特定・選別を行う必要があります。
認証処理の実装を行い、利用に応じた課金の対応等を検討する必要も検討要素です。

③データのバックアップ体制

クラウド環境においてデータのバックアップは重要課題のひとつになります。
バックアップはサービス提供側に完全に委託されるため、運用締結時にバックアップ方式についての 検討をし、サービス提供側との締結を結ぶ必要があります。

ご清聴ありがとうございました！



株式会社 システムズ

本資料に記載の内容は、印刷時点のものであり、セミナー当日の内容と一部異なっている場合もございます。予め、ご了承ください。

マイグレーションとは

「マイグレーション」

【英】migration

出典 ：IT用語辞典BINARY
(http://www.sophia-it.com/)

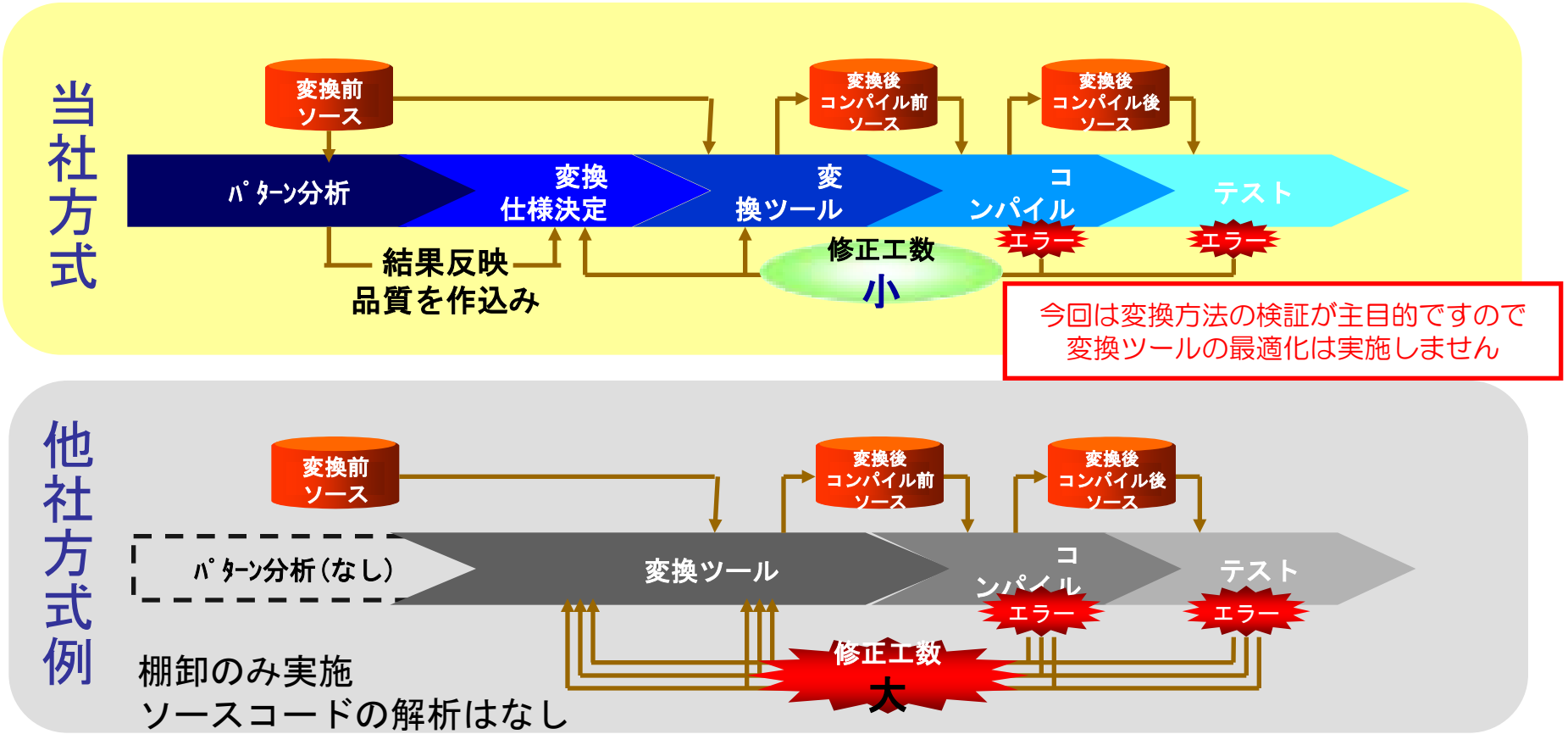
マイグレーションとは、システム、または、データ資産の移行作業のことである。マイグレーションは、基幹システムを新しいプラットフォームへ移行したり、OSやハードウェアなどの環境が異なるシステムへの移行を指す場合が多い。
マイグレーションは、手作業でデータのエクスポートやインポートなどの移行を行う場合もあるが、手作業ではミスが起こる可能性が高い。そのため、専用の移行ツールを用いて自動的に行われる場合もある。この移行ツールも、利用中のシステムや移行先の環境などを考慮して、カスタマイズされた後に使用されることが多い。移行には、異なる環境・機種間の移行、オープン系システムから別のオープン系システムへの移行、レガシーシステム(メインフレームを使った古いシステムのこと)からオープン系システムへの移行などさまざまな種類があるが、特に、レガシーシステムからオープン系システムへの移行はレガシーマイグレーションと呼ばれる。

【 マ イ グ レ ー シ ョ ン　ロ ー ド マ ッ プ 例 】

システムズのマイグレーション特徴

■当社マイグレーションの特徴（コンセプト）

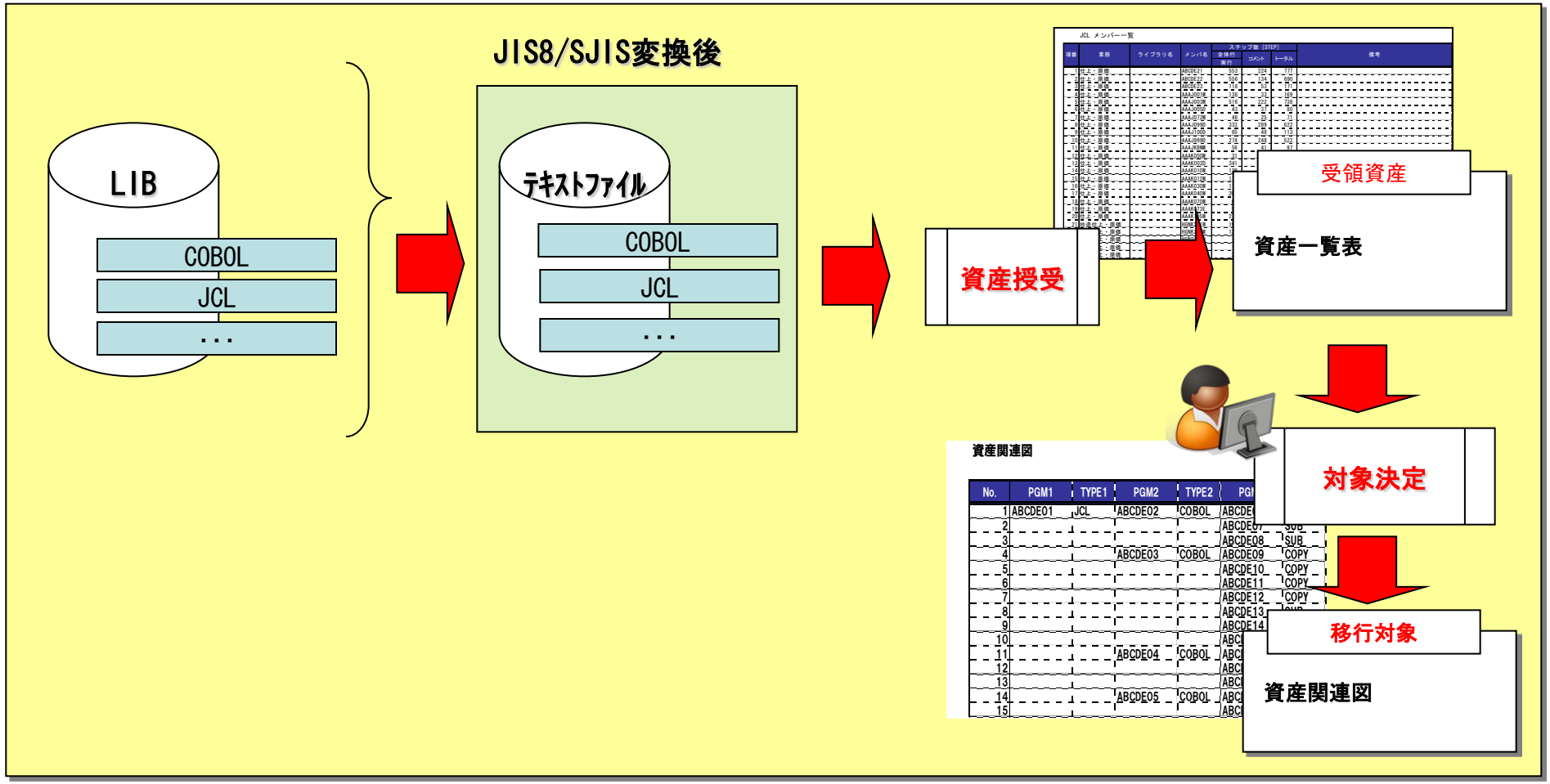
事前にプログラムソースを解析、変換方法を検証した後に、最適化したツールを準備します



２－１．マイグレーション作業内容

（１）資産授受

移行対象となる資産を頂き受領資産一覧表を提供します。



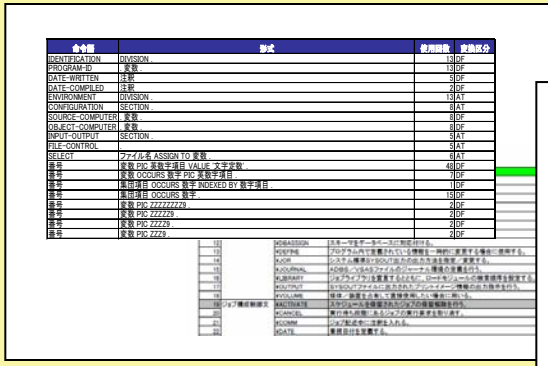
２－２．マイグレーション作業内容

（２）資産分析

変換対象ソースとなる資産を分析します。

対象ソース

資産関連図



資産関連図で使用している対象プログラム

テキストファイル

COBOL

JCL

・・・

資産分析

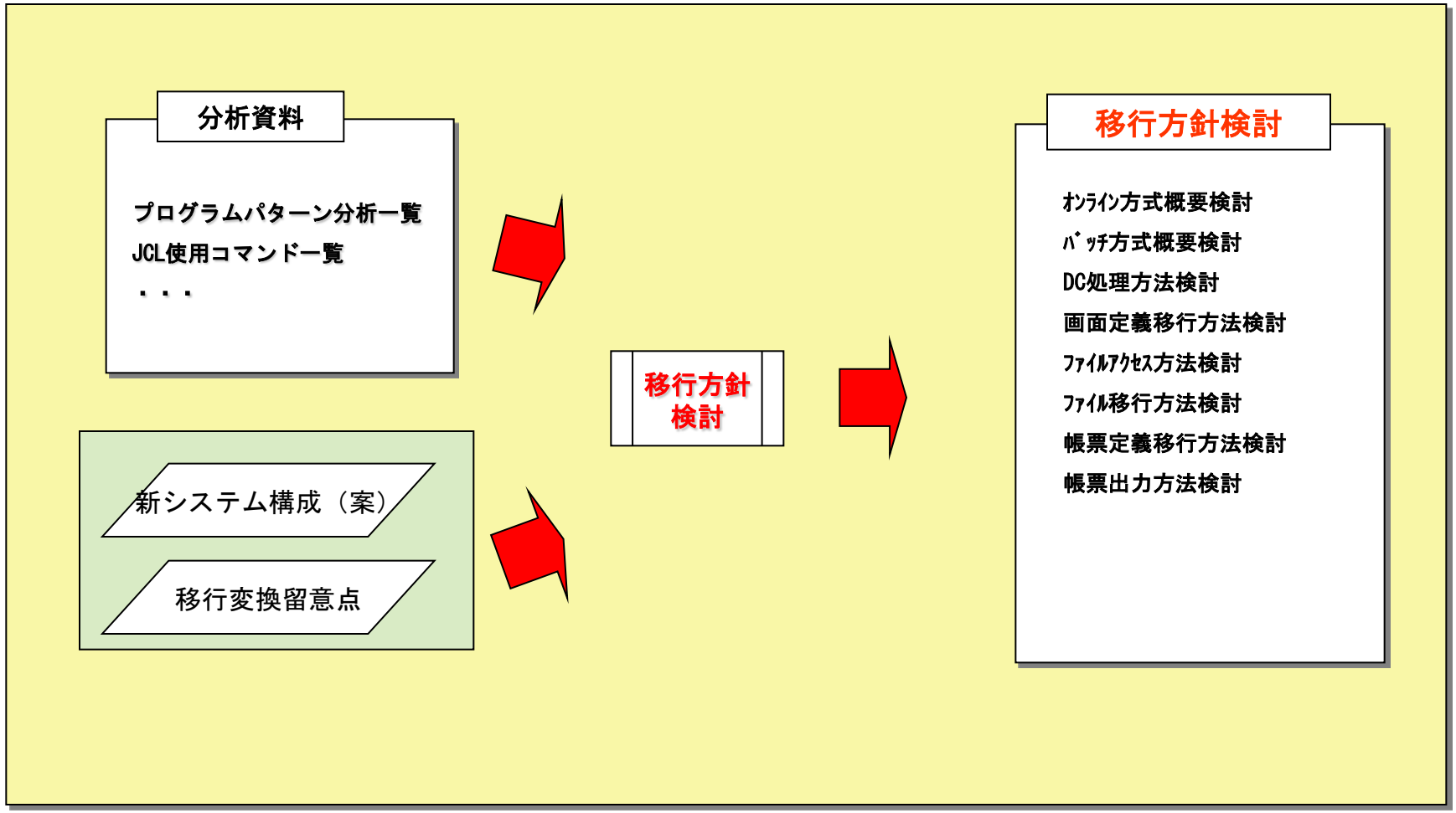
分析資料

- プログラムパターン分析一覧
- JCL使用コマンド一覧
- 画面定義使用パラメタ
- 帳票定義使用パラメタ
- ファイル使用状況
- ファイルのキー・アクセス方法状況

２－３．マイグレーション作業内容

（３）変換対象ソースの移行方針検討

分析結果を基に移行先環境を考慮した移行方針の検討を行ないます。



２－４．マイグレーション作業内容

（４）変換及び報告書作成

移行方針を基に技術的な留意点を中心にコンバージョンを行ないます。