

COBOL誕生50周年記念セミナー

日立マクセル殿における メインフレーム資産を再利用した 基幹系システム再構築事例

2010/04/16

日立公共システムエンジニアリング株式会社
システム再生ソリューション部

主任 芝山 敏満

日立マクセル殿における メインフレーム資産を再利用した 基幹系システム再構築事例

Contents

1. 顧客概要
2. プロジェクトの目的
3. マイグレーション作業内容
4. 成功の要因
5. まとめ

日立マクセル殿における
メインフレーム資産を再利用した
基幹系システム再構築事例

1. 顧客概要

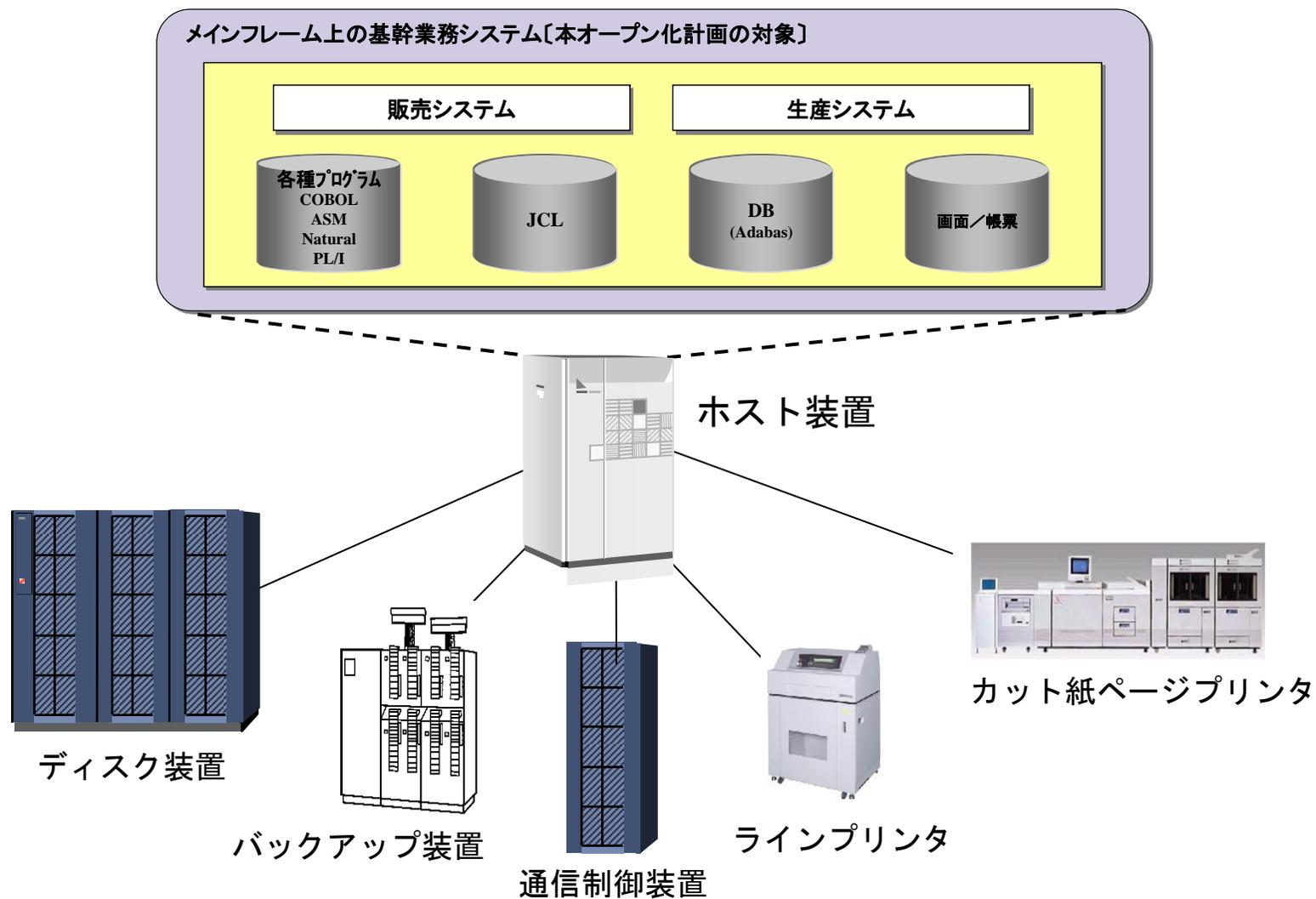
1. 顧客概要

1.1 顧客概要

社名	日立マクセル株式会社 
設立	1960年9月 
従業員数	連結4,451名、単独1,987名
国内製造拠点	5（茨城、京都2、大阪、兵庫）
国内販売拠点	4（東京、名古屋、大阪、福岡）
主要製品	情報メディア部門 （コンピュータテープ、放送用ビデオテープ、ブルーレイディスク、DVD、CD、オーディオテープ、ビデオテープ） 電池部門 （リチウムイオン電池、コイン形リチウム二次電池、酸化銀電池、リチウム一次電池、アルカリ乾電池） 材料・デバイス・電器部門 （光学部品、機能性材料、粘着テープ、RFIDシステム、小型電気機器、電鍍・精密部品、金型・合成樹脂成形品）

1. 顧客概要

1.2 既存システム概要



1. 顧客概要

1.2 既存システム概要

顧客資産概要

No.	区分	種別	販売システム		生産システム		合計	
			本数	規模(Ks)	本数	規模(Ks)	本数	規模(Ks)
1	プログラム	COBOL	2173	1214.6	1162	688.2	3335	1902.8
2		アセンブラ	619	273.3	62	33.9	681	307.2
3		Natural	1065	365.3	287	98.2	1352	463.5
4		PL/I	16	6.4	0	0	16	6.4
5	JCL		2694	—	1232	—	3926	—
6	DB	Adabas	136	—	25	—	161	—
7	画面・帳票	KMAP	400	—	177	—	577	—
合計			7103	1859.6	2945	820.3	10048	2679.9

日立マクセル殿における
メインフレーム資産を再利用した
基幹系システム再構築事例

2.プロジェクトの目的

2. プロジェクトの目的

2-1 基幹系システムの問題点

基幹系システムの経緯

1970年 メインフレーム上に構築

以来、安定性を重視し一貫してメインフレームの利用を堅持

環境の変化

周辺にオープンシステムが増加

問題の顕在化

- メインフレームの高額な運用コスト
- 柔軟性の欠如

2. プロジェクトの目的

2-2. プロジェクトの目的

課題と背景

高額な運用コスト

安定稼働の重視

メインフレーム専門
開発・保守要員

日々のプログラムメンテナンス

メインフレーム上稼働

複数の開発言語

システム規模が大きい

他システムとの連携が困難

特有業務ロジックの
ブラックボックス化

システムの柔軟性欠如

解決策

業務仕様の踏襲

総合運用管理ツール

オープンシステムの技術習得

オープンシステムの導入(切替)

開発言語の統一

資産棚卸し

RDBMS化

ブラックボックス化ロジックの明確化

2. プロジェクトの目的

2-2. プロジェクトの目的

課題と背景

高額な運用コスト

安定稼働の重視

メインフレーム専門
開発・保守要員

日々のプログラムメンテナンス

メインフレーム上稼働

複数の開発言語

システム規模が大きい

他システムとの連携が困難

特有業務ロジックの
ブラックボックス化

システムの柔軟性欠如

解決策

業務仕様の踏襲

総合運用管理ツール

オープンシステムの技術習得

オープンシステムの導入(切替)

開発言語の統一

資産棚卸し

RDBMS化

ブラックボックス化ロジックの明確化

2. プロジェクトの目的

2-2. プロジェクトの目的

課題と背景

高額な運用コスト

安定稼働の重視

メインフレーム専門
開発・保守要員

日々のプログラムメンテナンス

メインフレーム上稼働

複数の開発言語

システム規模が大きい

他システムとの連携が困難

特有業務ロジックの
ブラックボックス化

システムの柔軟性欠如

解決策

業務仕様の踏襲

総合運用管理ツール

オープンシステムの技術習得

オープンシステムの導入(切替)

開発言語の統一

資産棚卸し

RDBMS化

ブラックボックス化ロジックの明確化

2. プロジェクトの目的

2-2. プロジェクトの目的

課題と背景

高額な運用コスト

安定稼働の重視

メインフレーム専門
開発・保守要員

日々のプログラムメンテナンス

メインフレーム上稼働

複数の開発言語

システム規模が大きい

他システムとの連携が困難

特有業務ロジックの
ブラックボックス化

システムの柔軟性欠如

解決策

業務仕様の踏襲

総合運用管理ツール

オープンシステムの技術習得

オープンシステムの導入(切替)

開発言語の統一

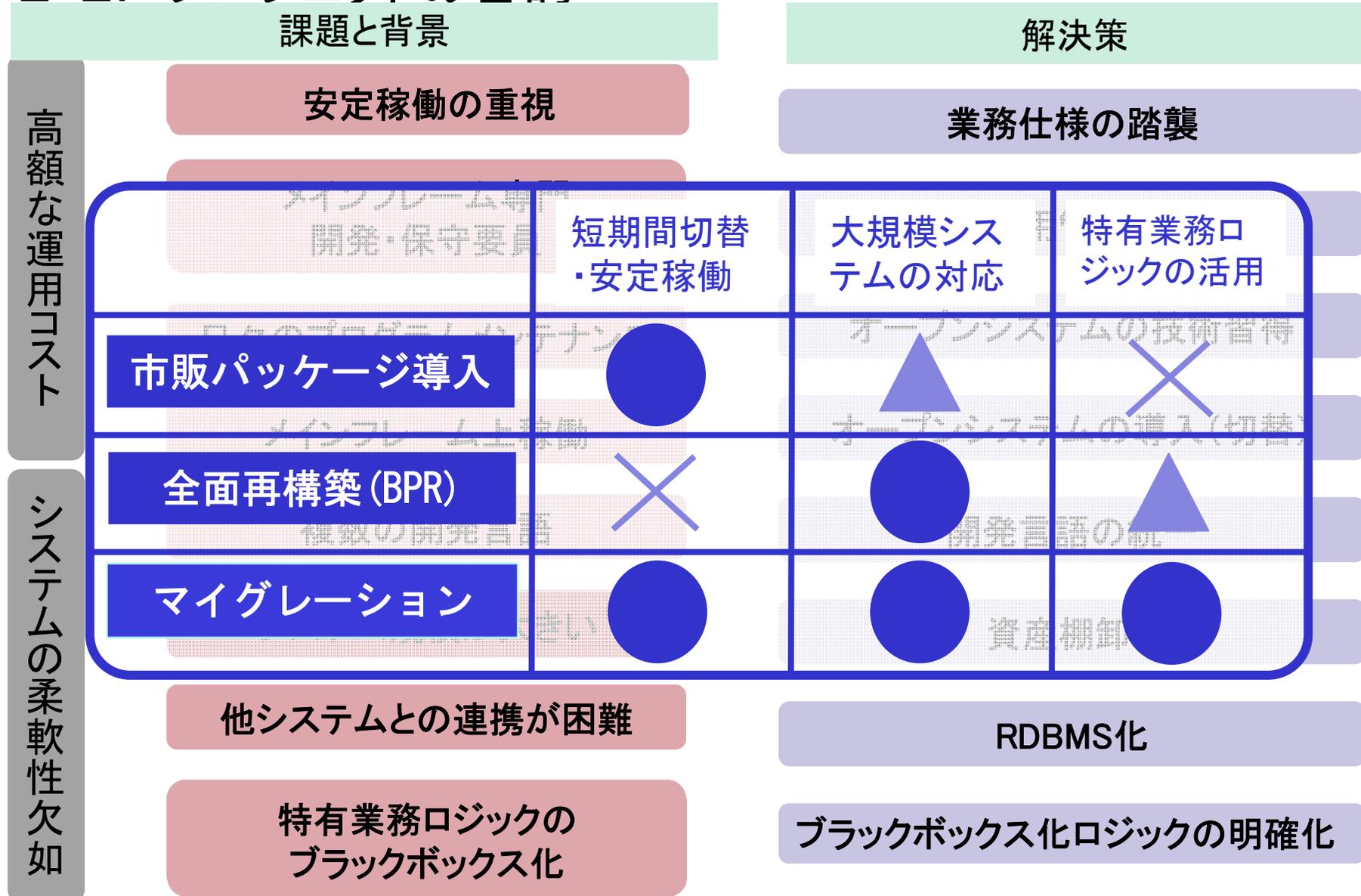
資産棚卸し

RDBMS化

ブラックボックス化ロジックの明確化

2. プロジェクトの目的

2-2. プロジェクトの目的



2. プロジェクトの目的

2-2. プロジェクトの目的

課題と背景

安定稼働の重視

メインフレーム専門
開発・保守要員

日々の

メ

システム規模が大きい

他システムとの連携が困難

特有業務ロジックの
ブラックボックス化

解決策

業務仕様の踏襲

総合運用管理ツール

技術習得

導入(切替)

統一

資産棚卸し

RDBMS化

ブラックボックス化ロジックの明確化

マイグレーションを選択

- ①ランニングコストの削減
- ②スキルのマイグレーション
- ③柔軟なシステムの実現

高額な運用コスト

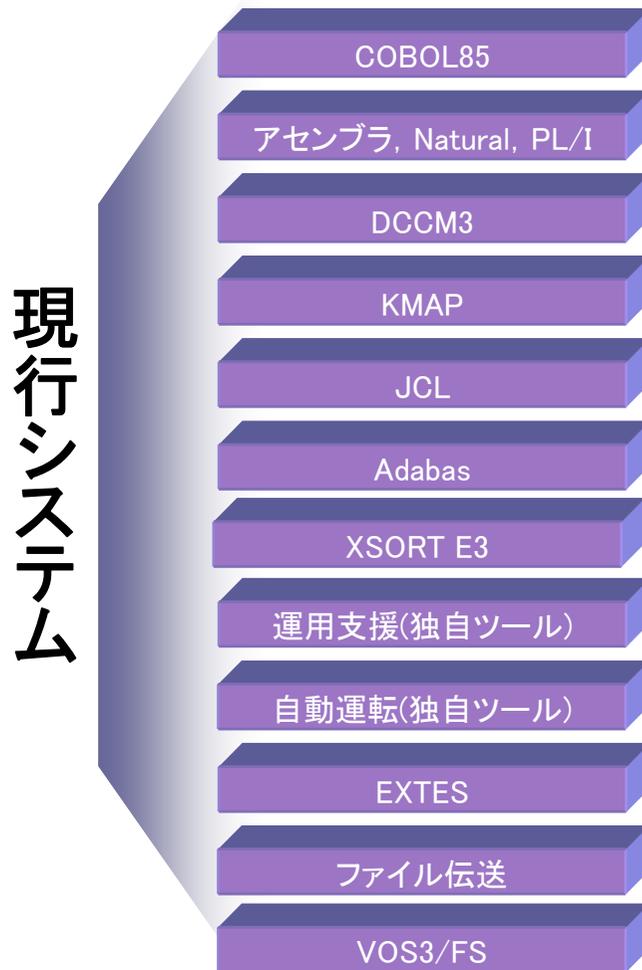
システムの柔軟性欠如

日立マクセル殿における
メインフレーム資産を再利用した
基幹系システム再構築事例

3. マイグレーション作業内容

3. マイグレーション作業内容

3-1. マイグレーション概要



<開発言語の統一>

COBOLの特徴

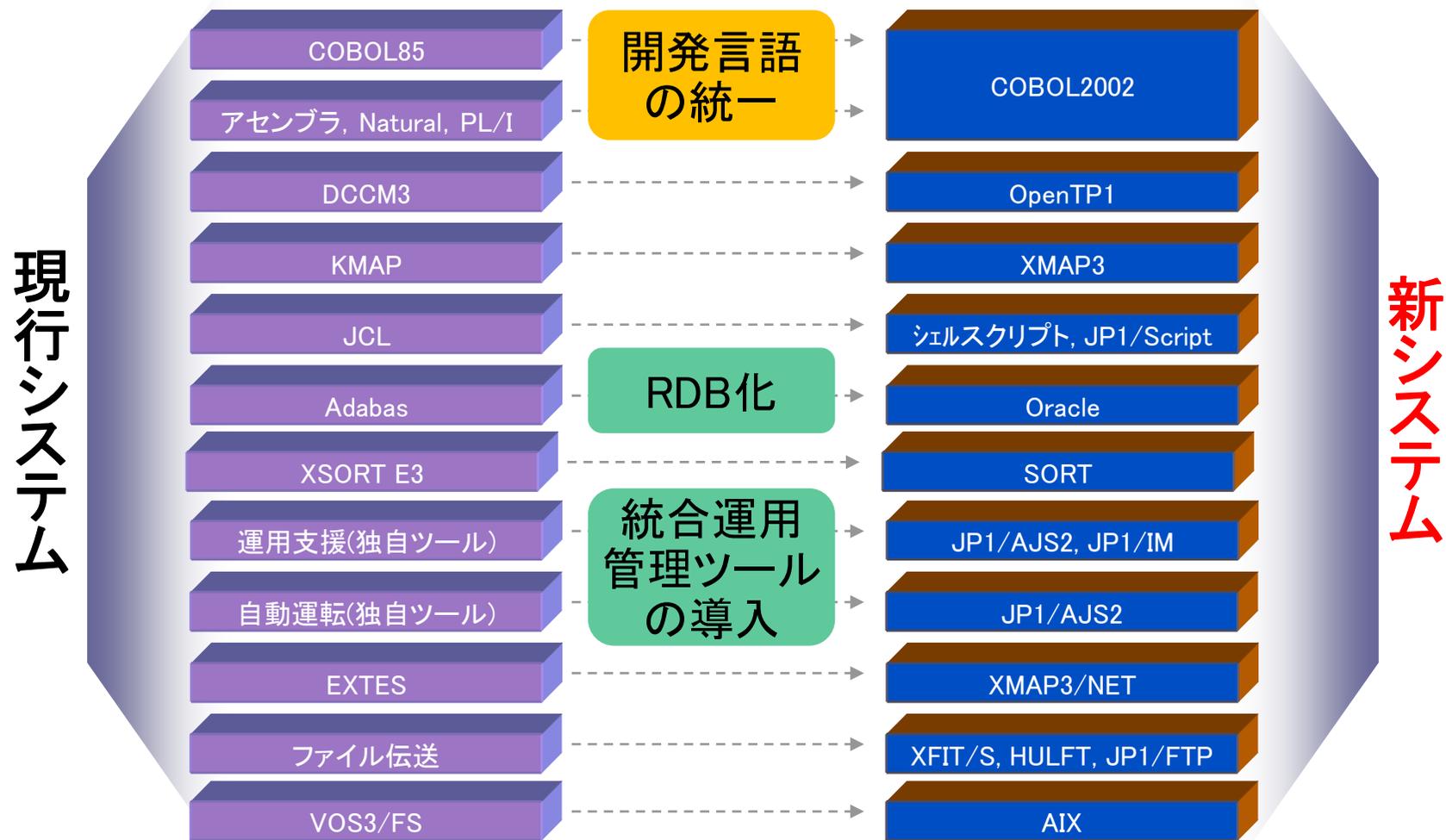
- ① ホスト系開発者の最大公約数的な言語である。
- ② 世界的に言語仕様が統一されている。
- ③ サーブレット環境でCOBOLラッパ等も利用可能。

COBOLへの統一を決断！

3. マイグレーション作業内容

3-1. マイグレーション概要

移行前後の資産構成

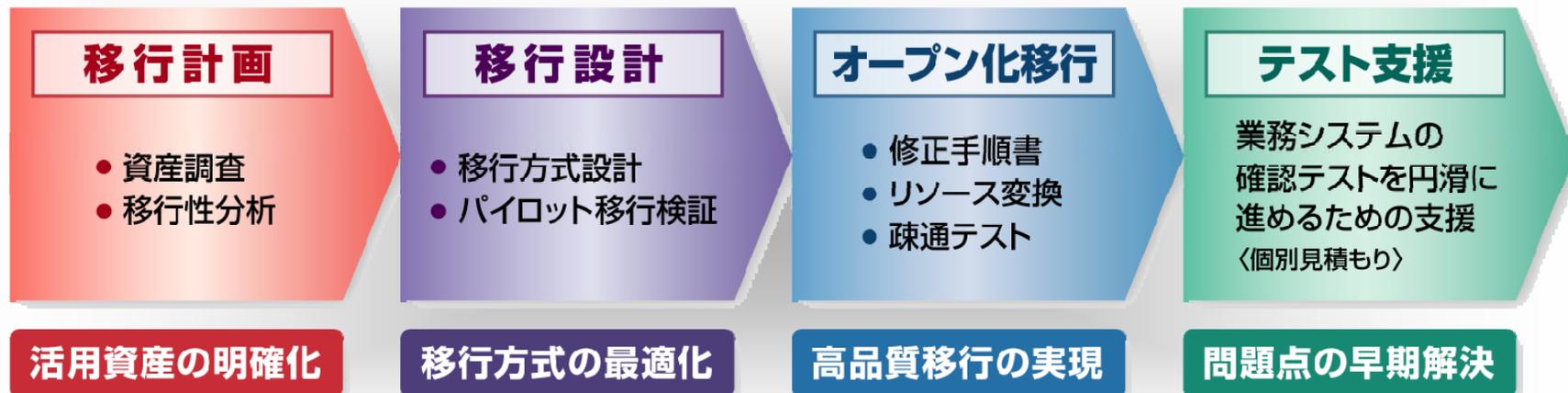


3. マイグレーション作業内容

3-2. マイグレーションの標準プロセス

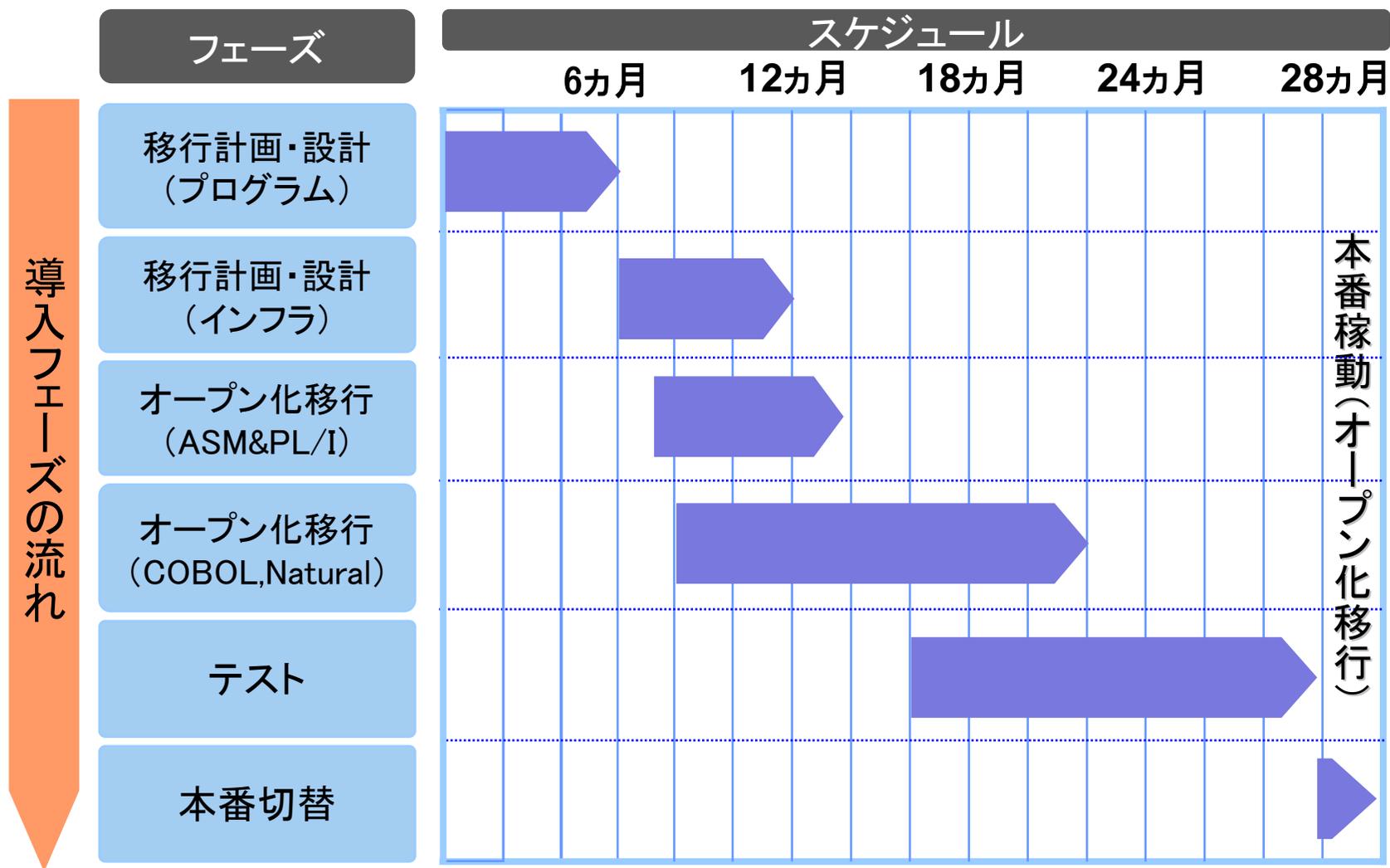
マイグレーションプロジェクトの標準プロセスを適用

運用支援ベンダがマイグレーションプロジェクトの進め方についてイメージすることができ、
工程毎にどのようなチェックをすべきかを明確にできる



3. マイグレーション作業内容

3-3. スケジュール



日立マクセル殿における
メインフレーム資産を再利用した
基幹系システム再構築事例

4. 成功要因

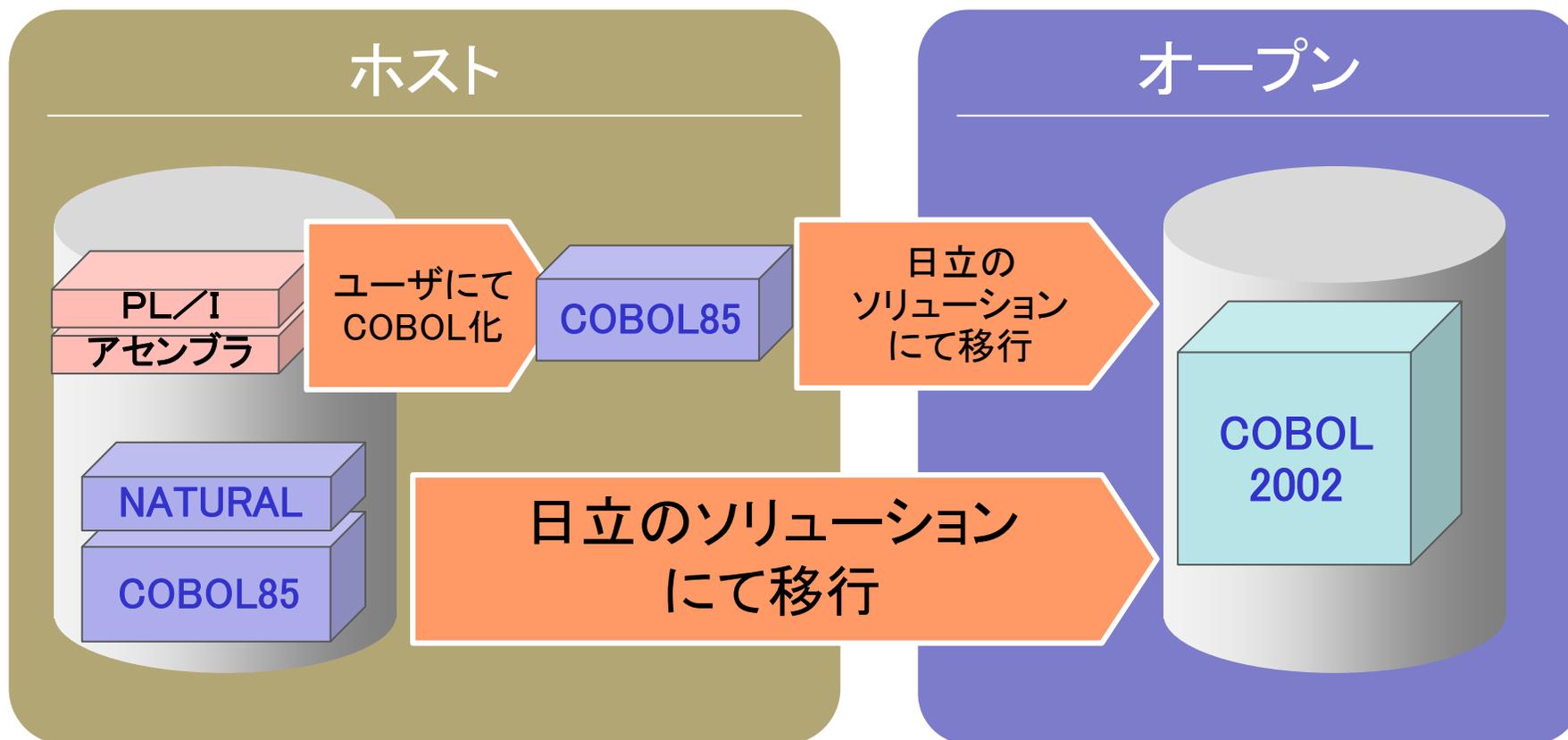
4. 成功要因

成功要因

- ① COBOLへの統一
- ② 顧客と一体となった作業分担
- ③ 製品設計製造部門と一体となった技術支援
- ④ 実績あるマイグレーションの適用

4. 成功要因

4.1 COBOLへの統一



4. 成功要因

4.1 COBOLへの統一

現行の開発・運用要員が持つ知識(言語、業務ノウハウ)



移行後も活用が可能となった!

開発要員の外部調達

↓(プログラマ人口が多い)

これまで同様に調達が容易となった!

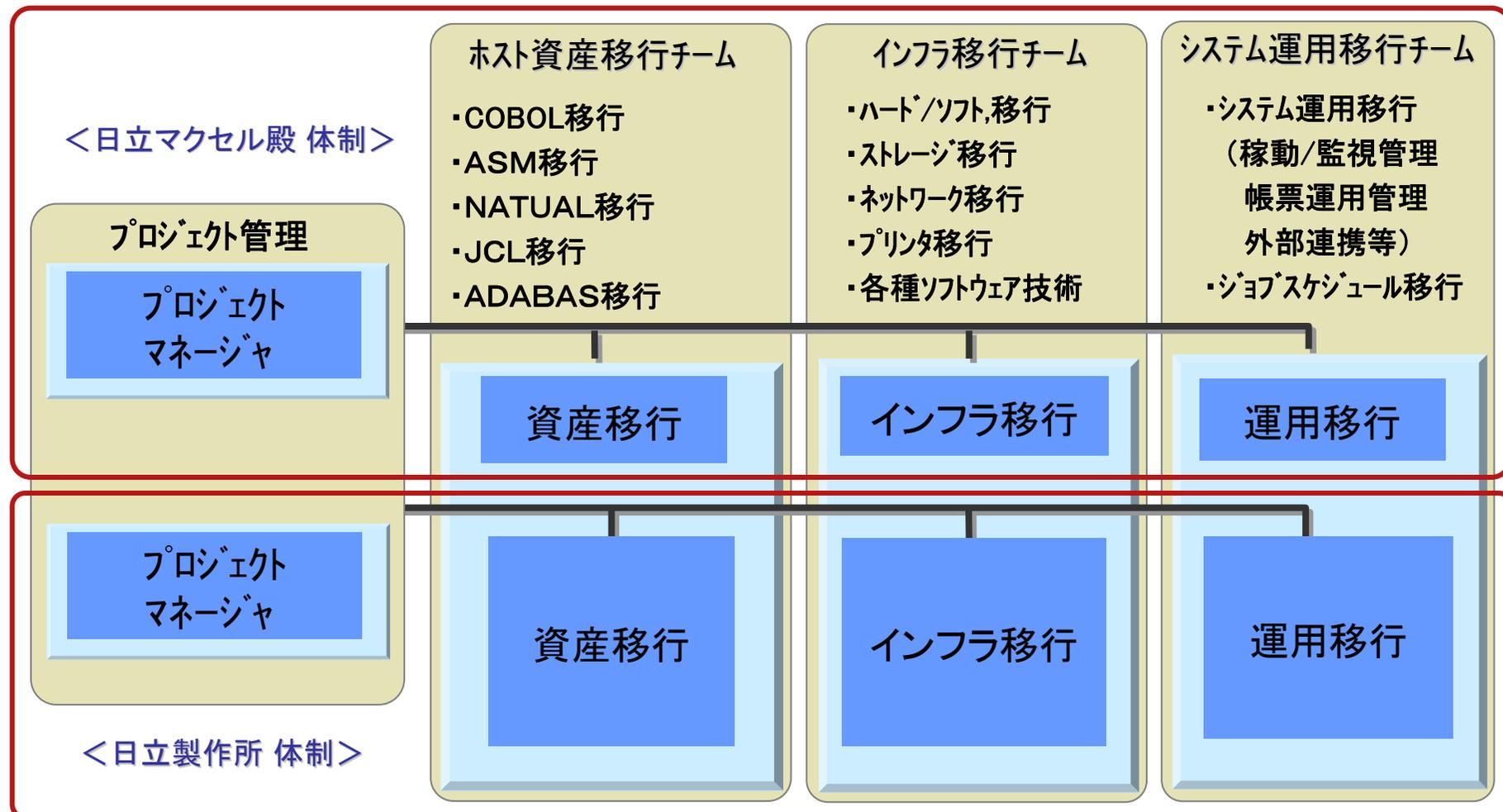
日立で標準仕様をベースとした製品を製造・販売



これまで同様のサポートが享受可能となった!

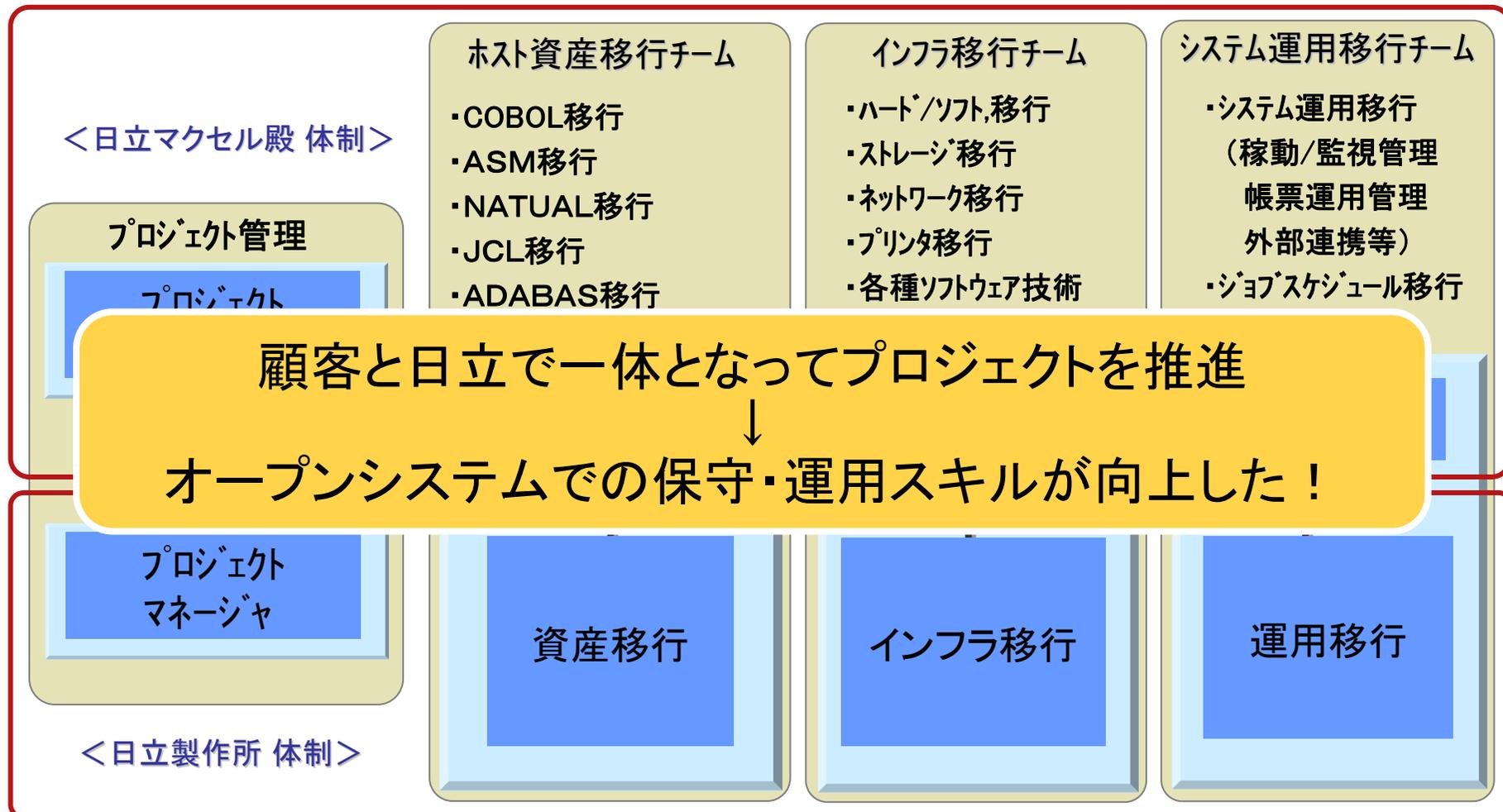
4. 成功要因

4.2 顧客と一体となった作業分担



4. 成功要因

4.2 顧客と一体となった作業分担



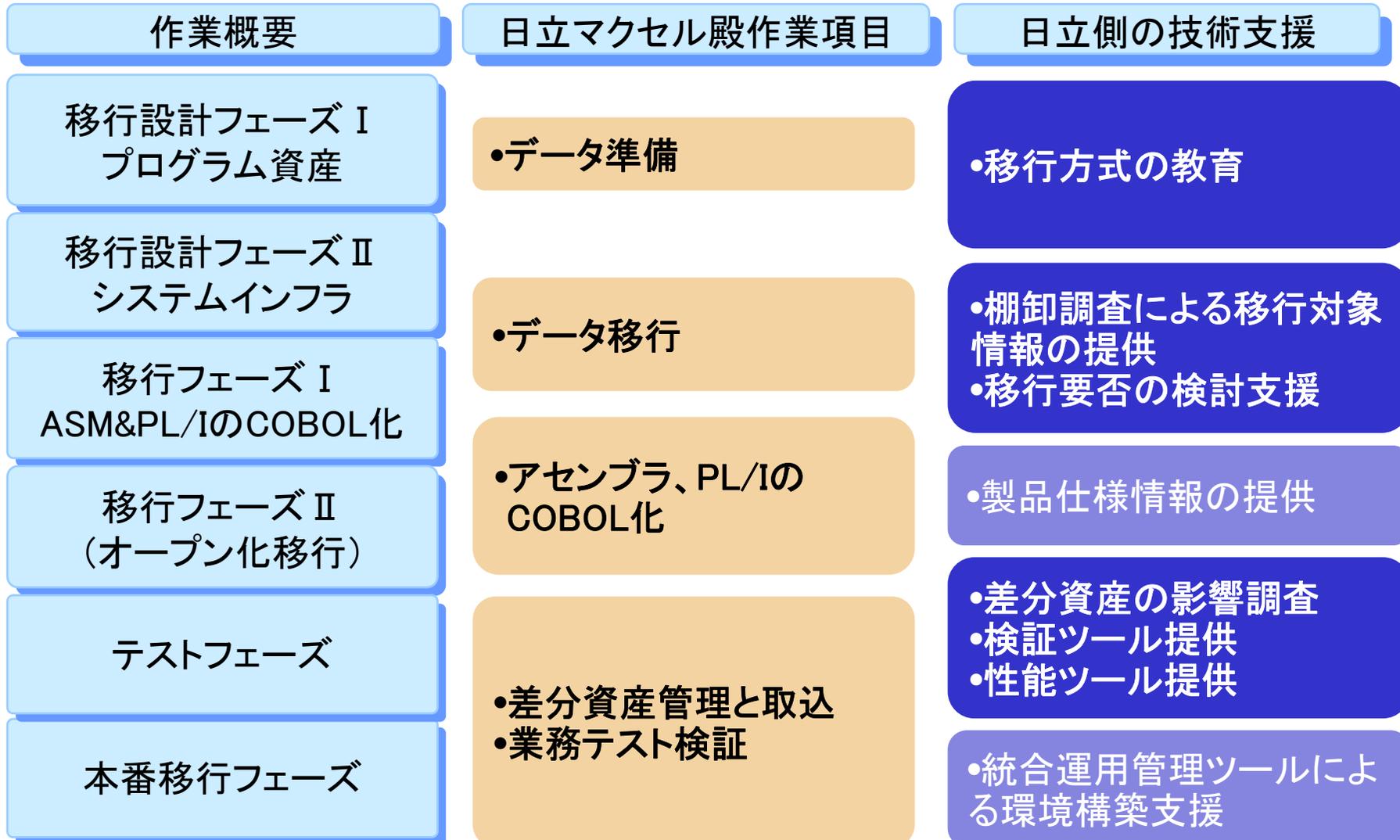
4. 成功要因

4.3 製品設計製造部門と一体となった技術支援



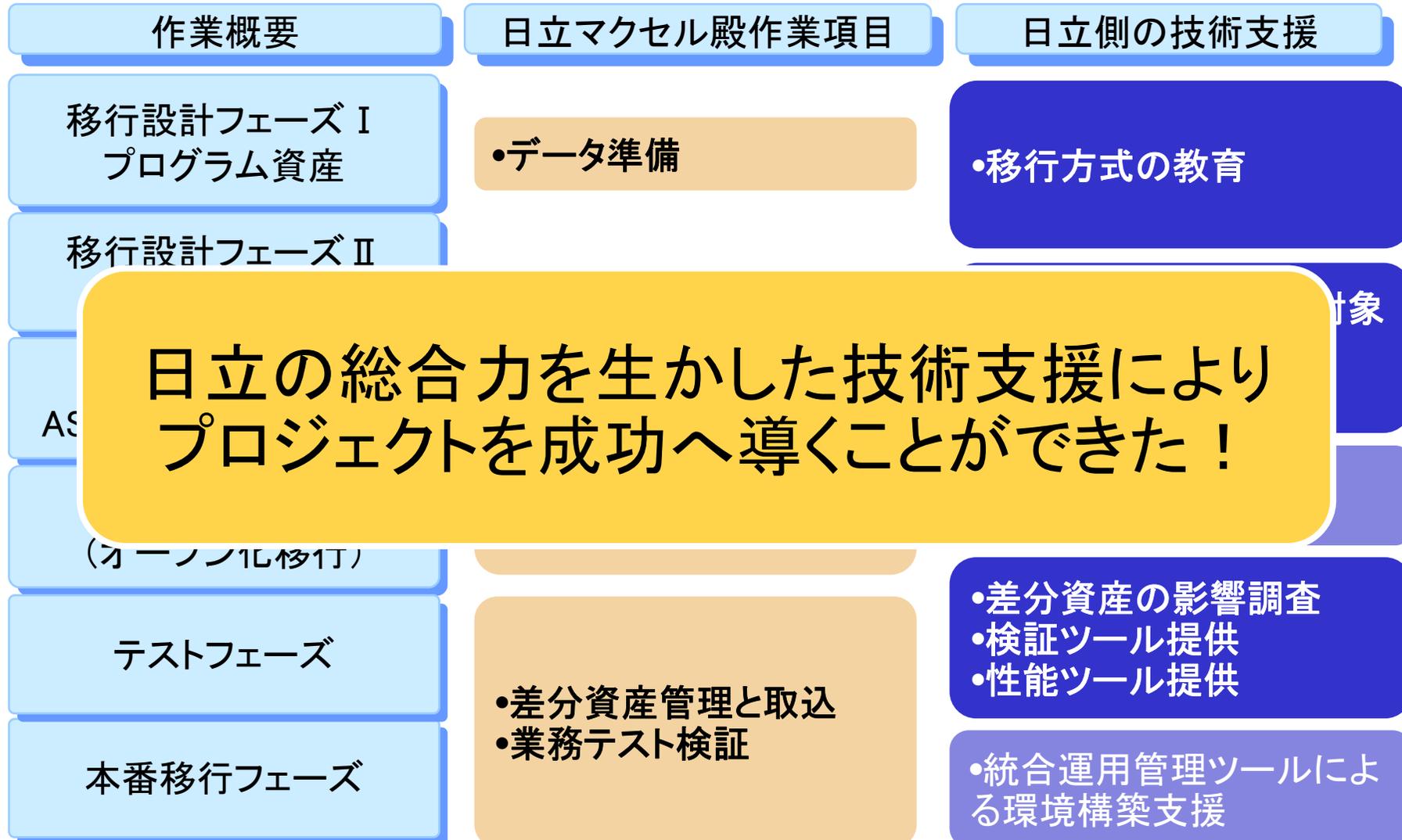
4. 成功要因

4.3 製品設計製造部門と一体となった技術支援



4. 成功要因

4.3 製品設計製造部門と一体となった技術支援



4. 成功要因

4.4 実績あるマイグレーションの適用

マイグレーションの特徴

- ◆システム資産を棚卸しすることにより、母体規模の縮減可能
- ◆既存のシステム資産を流用することにより短期間で移行可能
- ◆今までに蓄積された既存システムの業務機能をそのまま移行可能
- ◆日立の確立されたマイグレーション技術により、高品質を確保
(機械変換率:約90%)

大規模基幹系システムを早期に
安定稼働させることができた！

母体規模の縮減により、
将来のシステム拡張のリスク低減へ繋がった！

日立マクセル殿における
メインフレーム資産を再利用した
基幹系システム再構築事例

5. まとめ

5. まとめ

5.1 プロジェクトの成果

スキルのマイグレーション

メインフレーム
開発・運用担当要員



メインフレーム知識

COBOL言語知識

業務機能知識

[基礎技術の習得]オープンシステムの技術教育

[スキルの向上]共同作業体制の中、移行作業を分担・経験

オープン
開発・運用担当要員



オープン知識

COBOL言語知識

業務機能知識

5. まとめ

5.1 プロジェクトの成果

スキルのマイグレーション

新システムを自分達のものとする事ができた！

スキルの観点から、
将来的な取り組みの基盤ができた！

オープン
開発・運用担当要員



オープン知識

COBOL言語知識

業務機能知識

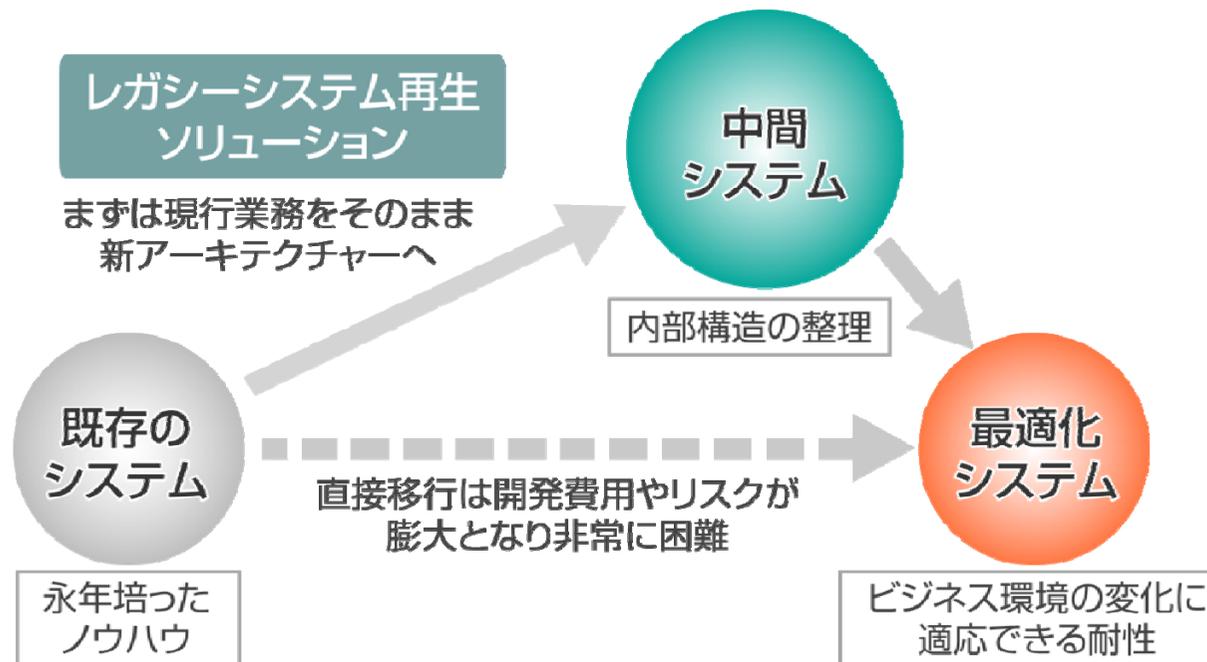
5. まとめ

5.1 プロジェクトの成果

柔軟なシステムの実現

移行には約2年間の期間を要したが、リスクを抑え業務を継承する形でオープンシステム化を実現し、将来への投資が可能となった。

- ① 今後のWeb開発やJAVA化といった環境の整備
- ② 他システムとのシームレスな連携



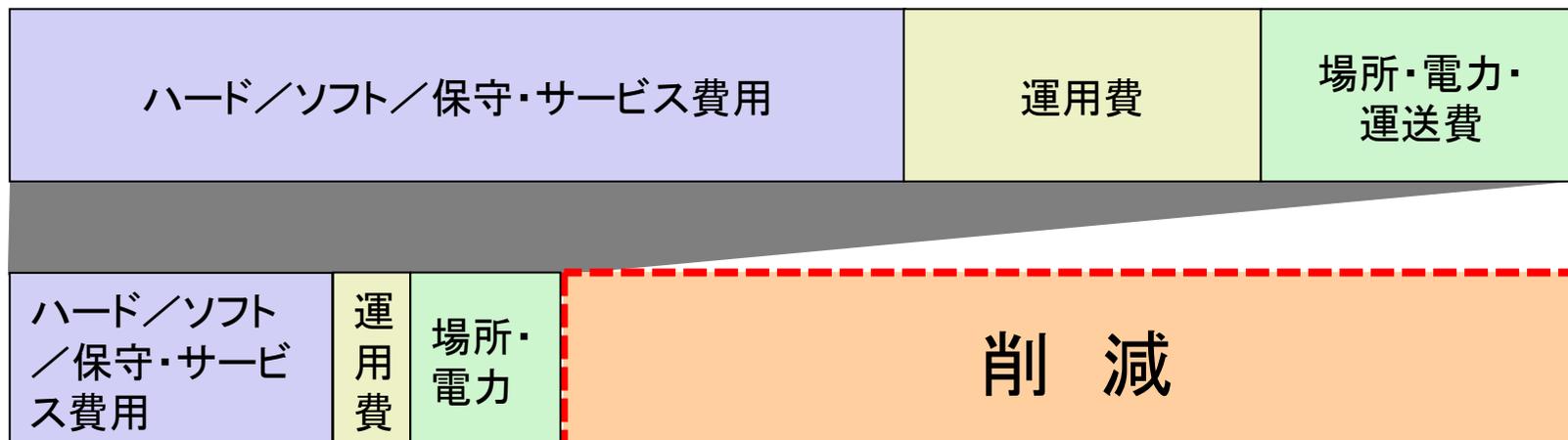
5. まとめ

5.1 プロジェクトの成果

ランニングコストの削減

本プロジェクトにより、メインフレームからオープンシステムへの移行が完了した。これにより、以下の効果が得られた。

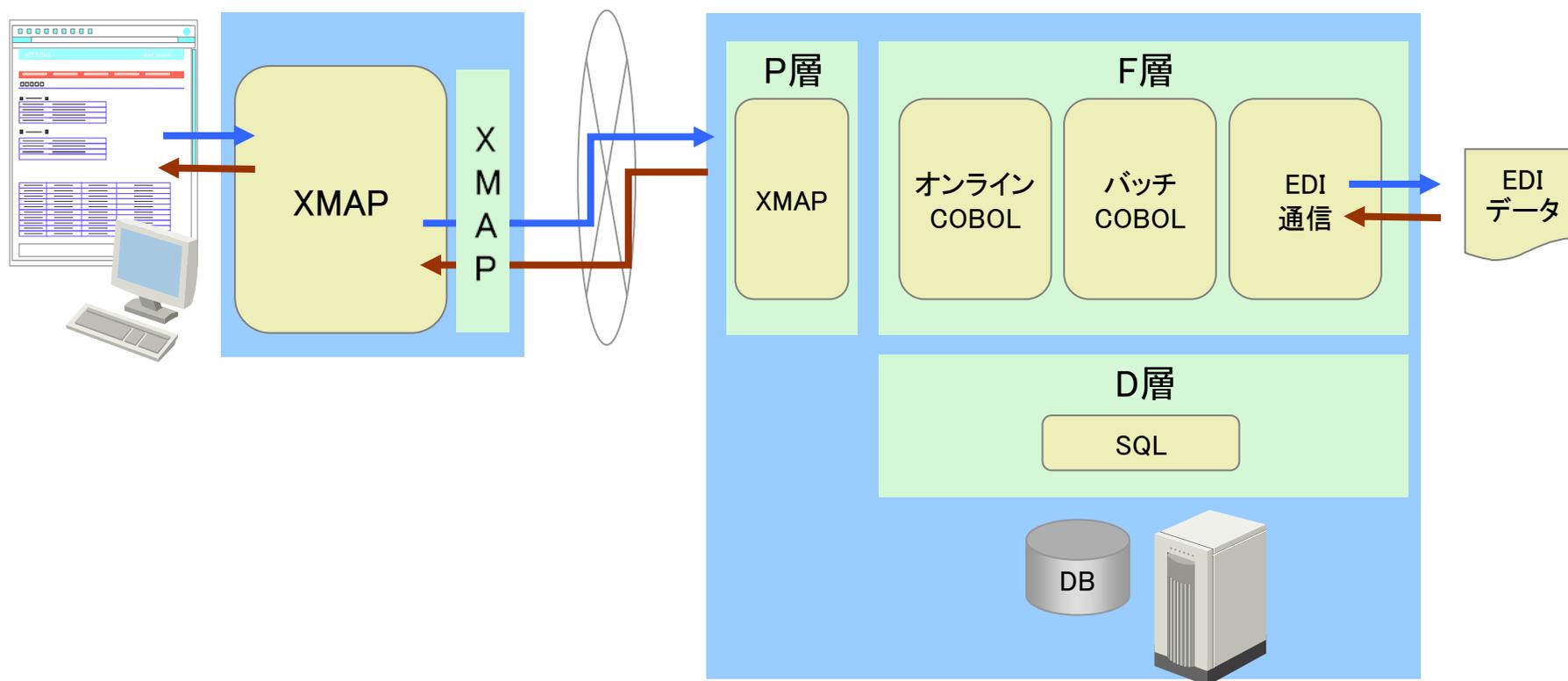
- ①アウトソーシングからインソーシングに切り換えることで、大きくコスト低減を図ることが出来た。
- ②メインフレーム使用時と比較すると、ハードウェアとソフトウェア買取りに伴う投資額を考慮しても、月々のランニングコストは約3分の1になった。



5. まとめ

5.2 システムの将来像

Web化



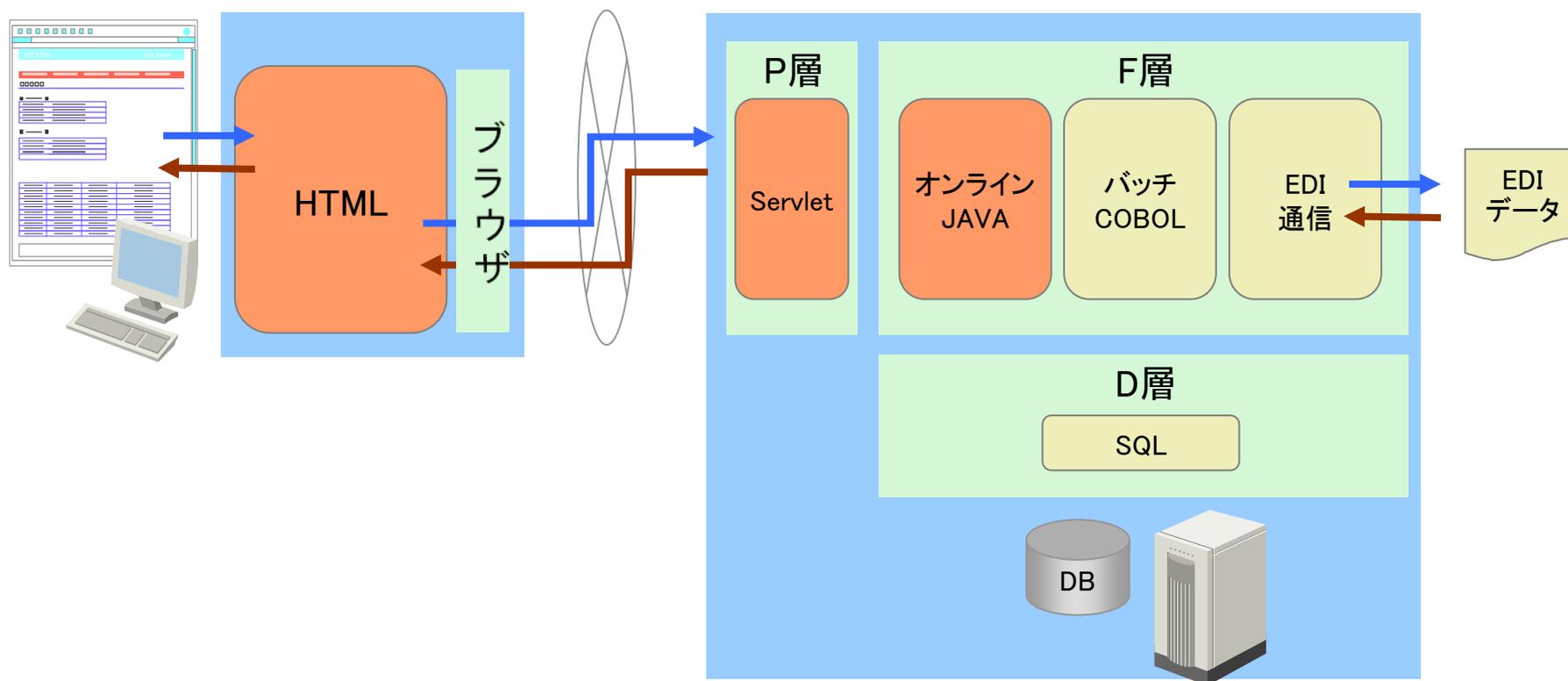
5. まとめ

5.2 システムの将来像

Web化

オンライン処理はWeb化

バッチ系COBOLは継続運用



END

日立マクセル殿における
メインフレーム資産を再利用した
基幹系システム再構築事例

2010/04/16

日立公共システムエンジニアリング株式会社
システム再生ソリューション部

主任 芝山 敏満

HITACHI
Inspire the Next