

物流業・基幹システムのオープン化事例

～メインフレーム資産をすべて移行

2010年9月7日

株式会社メイテツコム

物流ソリューション部

柄澤 靖博

経済産業省SI・SO認定企業



JQA-IT0035
共通利用型グループウェア
システムの運用、保守及び
サービスデスク



JQA-IM0008
WebソリューションIDC、
放送通信連携、文教、
公共ソリューションサービス

株式会社メイテツコム

MEITETSU COM Co.,Ltd.



設立年月日	1976年9月27日 名古屋鉄道をはじめグループ3社の電算部門を統合して設立
資本金	1億円
代表者	代表取締役社長 浅井保夫
従業員	258名
売上高	69.3億円
事業所	本社 名古屋市中村区名駅南 名鉄協商コンピュータビル 東京支店 東京都港区浜松町
関連会社	株式会社名鉄情報システム 株式会社ホクリクコム
取得認証	ISO/IEC27001 (情報セキュリティマネジメント認証) ISO/IEC20000 (ITサービスマネジメント認証) SI登録 (システムインテグレータ企業登録) S0認定 (特定システムオペレータ企業認定)
URL	http://www.meitetsucom.co.jp/

◆ ブランディングステートメント（お客様へのお約束）



スコープ イノベーション

視界革新

◆ 業務内容

開発・運用・利用の総合力でITのワンストップサービスを提供

～情報化コンサルティングから企画、開発、

そして情報システムのアウトソーシングまでを統括～

- ・業種別ソリューションの提案およびシステム開発・運用
- ・情報システムアウトソーシングの受託
- ・インターネットデータセンターサービス
- ・ASPサービス、パッケージソフトの開発・販売
- ・ISMSコンサルティング、IT教育の受託
- ・Webコンテンツの企画・制作
- ・ネットワークコンサルティング

◆ 業種別システム構築事例

業 種	事 例
交 通	時刻表Web検索、特急列車座席予約、ダイヤ作成、お客様センターシステム 観光バス運行管理パッケージ「楽々道中」、乗合バスICカード乗車券システム バスロケーションシステム、高速バス乗車券発券システム フェリー予約システム
物 流	貨物業基幹業務システム、送り状・荷札発行パッケージ「こぐまくん」 流通倉庫管理システム、各分野配送センターシステム、流通倉庫管理パッケージ
流 通	百貨店システム（POS、友の会） 専門店チェーン基幹業務システム「楽々専門店チェーン」 クリーニング業基幹業務システム、空港免税店システム ホームセンターチェーン発注・特売・店舗システム
サービス	ホテル総合システム、旅行代理店予約管理システム 不動産業会計システム、設備工事業積算・資材システム 病院オーダリングシステム
文 教	大学事務システム（入試・入学・履修・成績・学生情報） 私大オープンカレッジシステム
メディア	TV局地上波デジタル放送システム、番組表システム、ワンセグシステム
行政、自治体	自治体施設予約、官庁出先機関向け給与計算システム
共 通	勤怠・給与・確定拠出年金、固定資産システム、グループシェアード経理システム、 フレックス勤務システム、ERPパッケージ導入

名鉄運輸株式会社

Meitetsu Transport Co., Ltd.

～安全・確実・迅速なトータル物流サービス～



設立年月日	1943年6月1日
資本金	20.6億円
代表者	代表取締役社長 米原 浩一
従業員	3,923名(名鉄運輸グループ全体：8,425名)
売上高	581.63億円(名鉄運輸グループ全体：895.84億円)
所在地	本社 愛知県名古屋市東区葵2丁目12番8号 名鉄運輸ビル
関連会社	名鉄ゴールデン航空(株)をはじめ、名鉄運輸グループ20社
支店・営業所	全国約300か所
取得認証	ISO14001 (環境マネジメントシステム認証) ISO9001-2000 (品質マネジメントシステム認証)
事業内容 ※主な内容	一般貨物／引越等の輸送業務 貨物保管／倉庫管理／物流加工なども含めた総合物流サービス
URL	http://www.meitetsuunyu.co.jp/

信州名鉄運輸株式会社

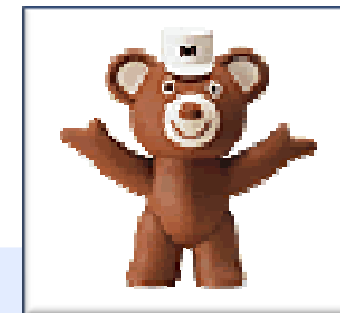
Shinsyu Meitetsu Transport Co., Ltd.



設立年月日	1946年8月21日
資本金	4.7億円
代表者	代表取締役社長 吉田 文太
従業員	506名
売上高	185.8億円(信州名鉄運輸グループ全体：347.3億円)
所在地	本社 長野県松本市鎌田2丁目8番10号
関連会社	信州名鉄運送(株)をはじめ、信州名鉄運輸グループ9社
支店・営業所	全国約44か所
取得認証	安全性優良事業所認定 15事業所 グリーン経営認証 6事業所 I S M S 1事業所
事業内容 ※主な内容	一般貨物自動車運送業（特積みを含む） 倉庫業
URL	http://www.shinmei-net.co.jp/

四国名鉄運輸株式会社

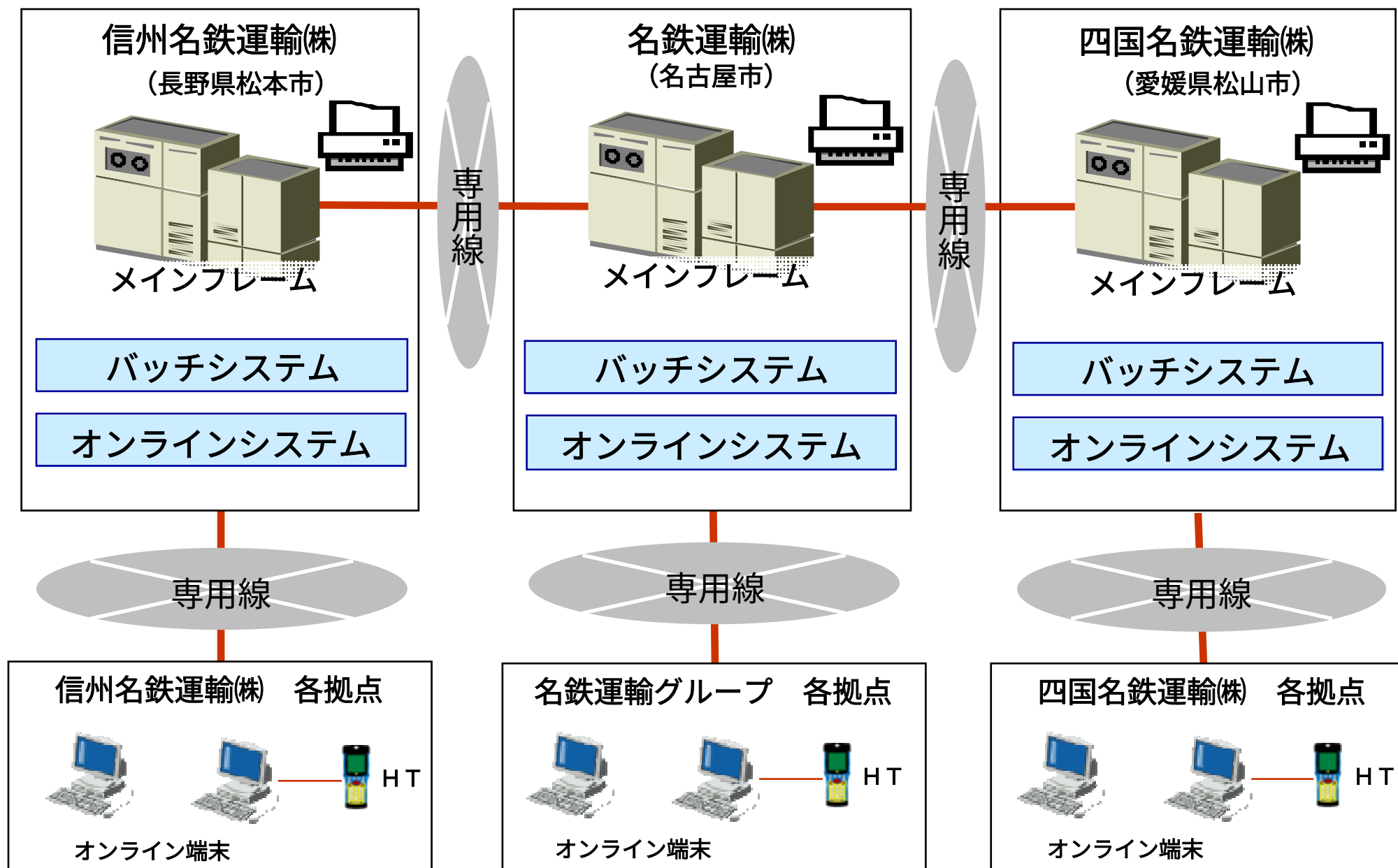
Shikoku Meitetsu Transport Co., Ltd.



設立年月日	1942年3月3日
資本金	2.4億円
代表者	代表取締役社長 中島 慎太郎
従業員	1,050名
売上高	146億円
所在地	本社 愛媛県松山市空港通4丁目5番5号
関連会社	四国名鉄運送(株)をはじめ、四国名鉄運輸グループ5社
支店・営業所	全国約28か所
取得認証	安全性優良事業所認定 14事業所 グリーン経営認証 11事業所
事業内容 ※主な内容	一般貨物自動車運送業（特積みを含む） 倉庫業・航空貨物取扱事業・流通加工業務
URL	http://koguma.shikokumeitetsu.co.jp/

1. 基幹システムオープン化の背景
2. オンラインシステムのオープン化
3. バッチシステムのオープン化
4. Webシステムとの連携
5. まとめ

1-1.基幹システムオープン化の背景：従来の構成



1-2.基幹システムオープン化の背景：業務構成

輸送オンラインシステム

輸 送 管 理

未 収 管 理

簡易端末システム

収 支 管 理

精 算 管 理

輸 送 管 理

営 業 支 援

マスタ 管 理

委託会社サービス

日次・月次バッチシステム

輸 送 管 理

未 収 管 理

収 支 管 理

精 算 管 理

営 業 支 援

マスタ 管 理

車両稼働管理

人事・給与管理

VANシステム

輸送情報送受信

1-3.基幹システムオープン化の背景：背景

■1998年より業務分析を行い、従来システムの課題整理、システム再構築のコンセプトを取りまとめた。

◆システムへの要求：効率化と低コスト運用への見直し

- ・メインフレームによる基幹システムは、稼働から17年を経過し、システムの柔軟性に問題！
- ・時代のニーズにマッチした業務の最適化によるシステムの軽量化とダウンサイジングによる運用コストの削減！

◆市場のニーズ

- ・景気低迷による低価格な輸配送を要求！
- ・高品質／高サービスの高まり！
- ・荷物のリアルタイム情報の要求が加速！

◆時代の変化

- ・ITインフラの急速な整備！
 - ・ハードウェアとOSの成長！
 - ・機器が小型化され使い勝手が向上！
- ⇒勢いを増すオープン化への流れ！

基幹システムのオープン化

1-4.基幹システムオープン化の背景:オープン化の目的

■運送業界の業務の高度化、多様化に対応するには基本的なインフラを再構築し情報システムの戦略的な活用で競争力を強化！

時代の変化と
顧客ニーズへの対応

システムの効率化と
低コスト運用への見直し

新世代の物流サービスを
オープンシステムで再構築

「メインフレームからサーバへの基幹システムマイグレーション」

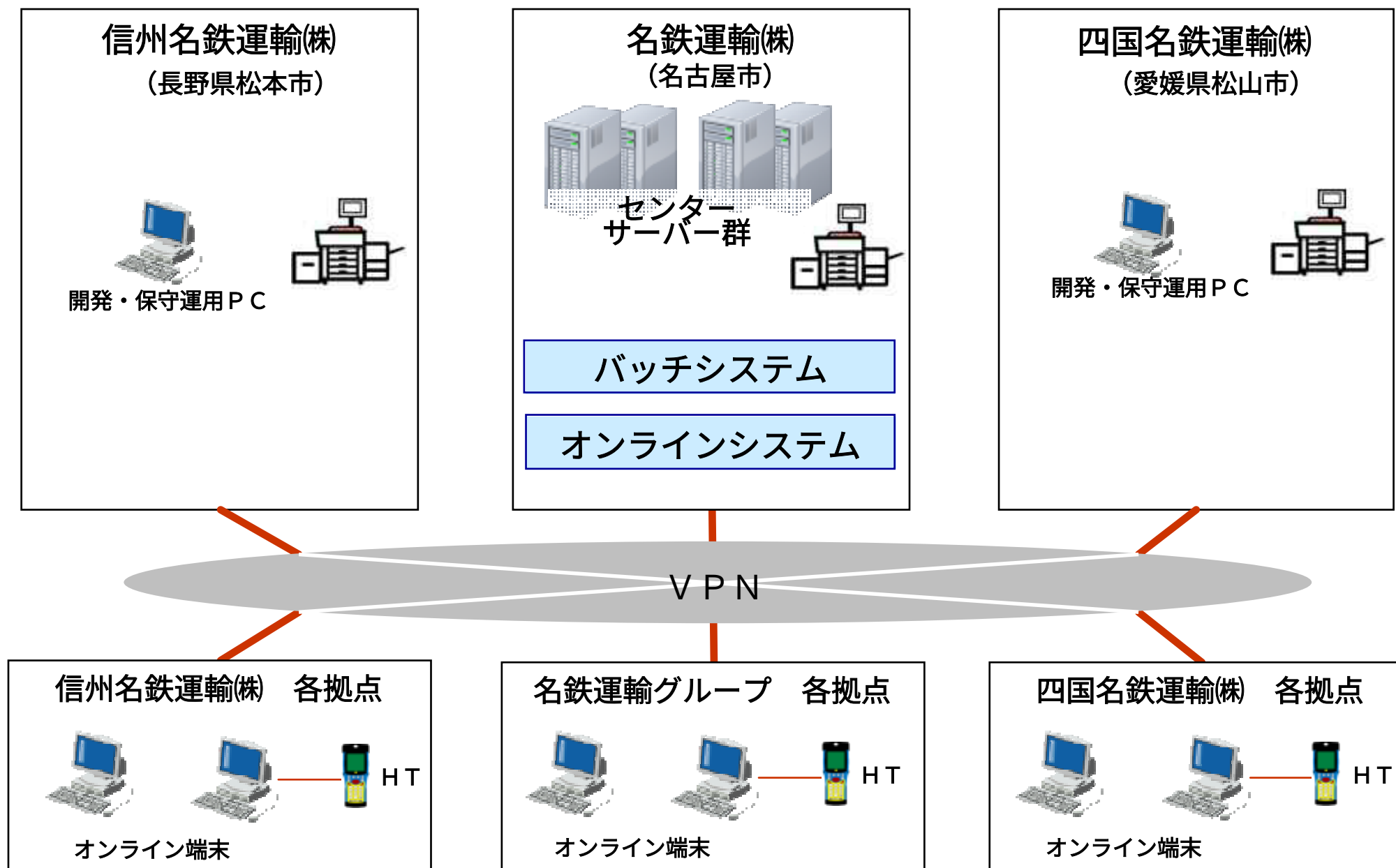
輸送品質と
顧客サービスの向上

業務の効率化

信頼性と即時性の高い
輸送情報を提供

競争力を強化

1-5.基幹システムオープン化の背景:オープン化後の構成



1. 基幹システムオープン化の背景
2. オンラインシステムのオープン化
3. バッチシステムのオープン化
4. Webシステムとの連携
5. まとめ

■オープン化の課題

- ・グループ各社の旧端末機器のリース残の調整！
- ・限られた要員の中で全国一斉の端末機器の切り替えが困難！
- ・現場運用の変更に對し段階的な導入を！
- ・各社のメインフレームで蓄積された移行対象データが点在かつ膨大！

新／旧オンラインシステムを並行稼動

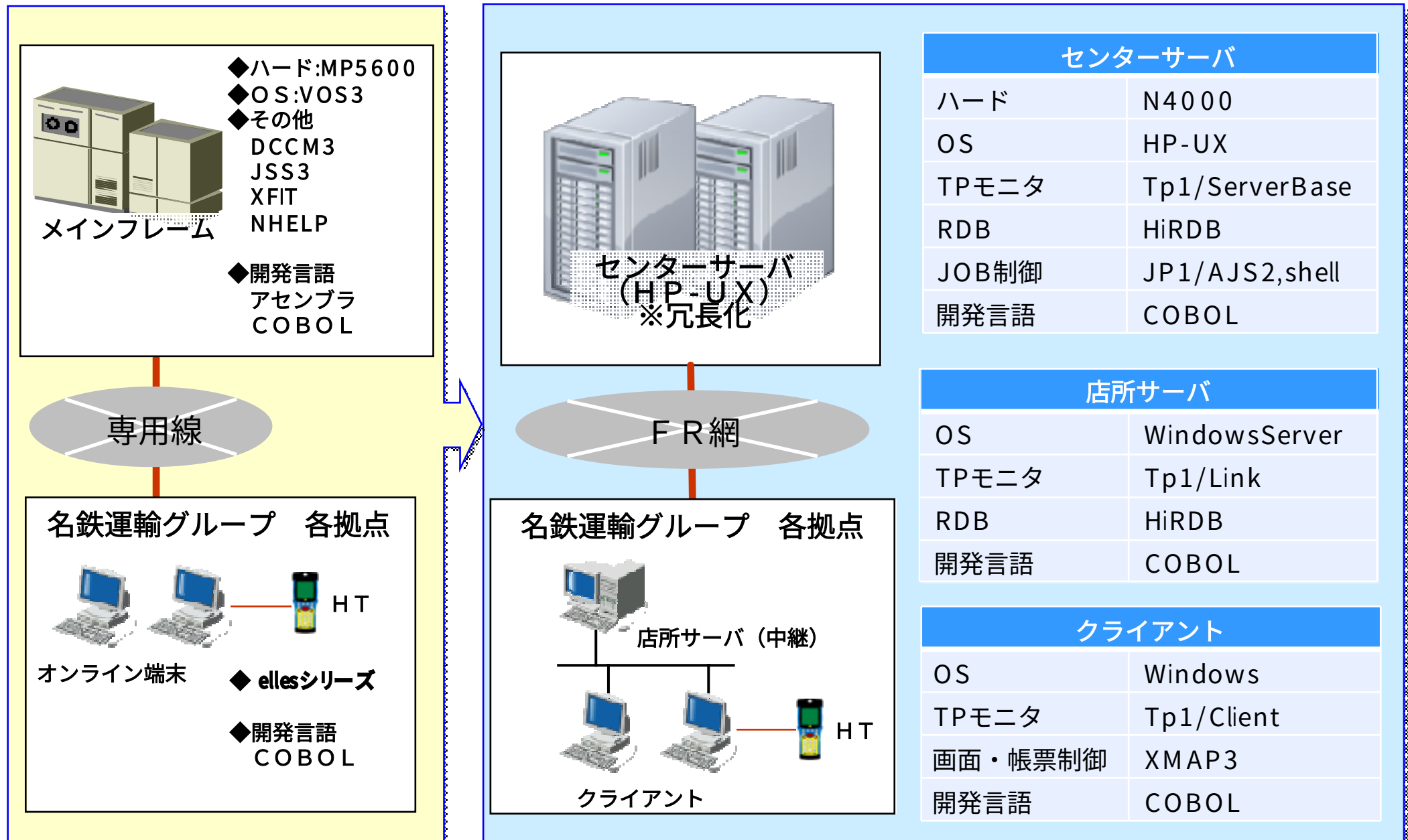
並行稼動のポイント

輸送情報の連続性

リアルタイムなデータ交換

データの整合性

2-2.オンラインシステムのオープン化:移行パス



2-3.オンラインシステムのオープン化:環境選定

■ TPモニタ

- メインフレームと同等の信頼性
- OLTP環境のシステム構築
- ミッションクリティカルなシステムに対応可能

OpenTP1

■ データベース

- 大量のデータの高速処理
- 高トラフィックに対応
- 高性能・高信頼性・高可用性

HiRDB

■ 開発言語

- 長期のシステム利用に耐え得る言語
- 長年蓄積された業務知識と開発技術を持つ要員を活用
- メインフレームの資産を有効活用

COBOL85

OLTP:OnLine Transaction Processing

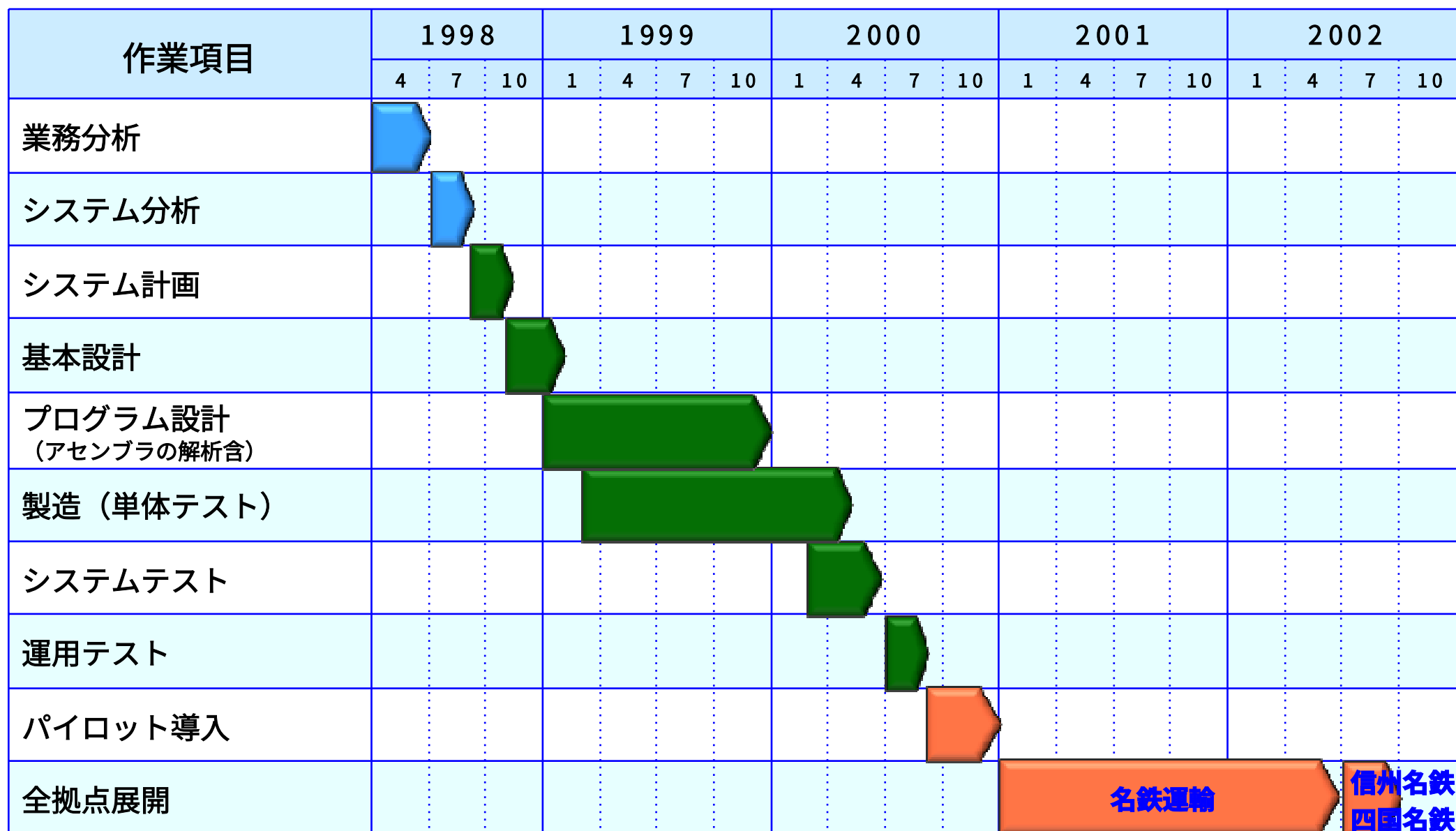
2-4.オンラインシステムのオープン化:規模

作業項目	数量
1 センタサーバ	
アセンブラ ⇒ COBOL化	3 9 8本
COBOL開発	2 4本
2 店所サーバ	
COBOL開発	4 4 1本
3 クライアント	
COBOL開発	9 7本
画面作成	1 3 9画面
帳票作成	1 3 2帳票
4 デイリーバッチ処理	
アセンブラ ⇒ COBOL化	7 8 8本
COBOL開発	1, 7 7 6本
5 データ移行	
センタサーバ	2 0 0, 0 0 0件
店所サーバ	1 0 0, 0 0 0件

全体作業工数

1, 1 2 1人月

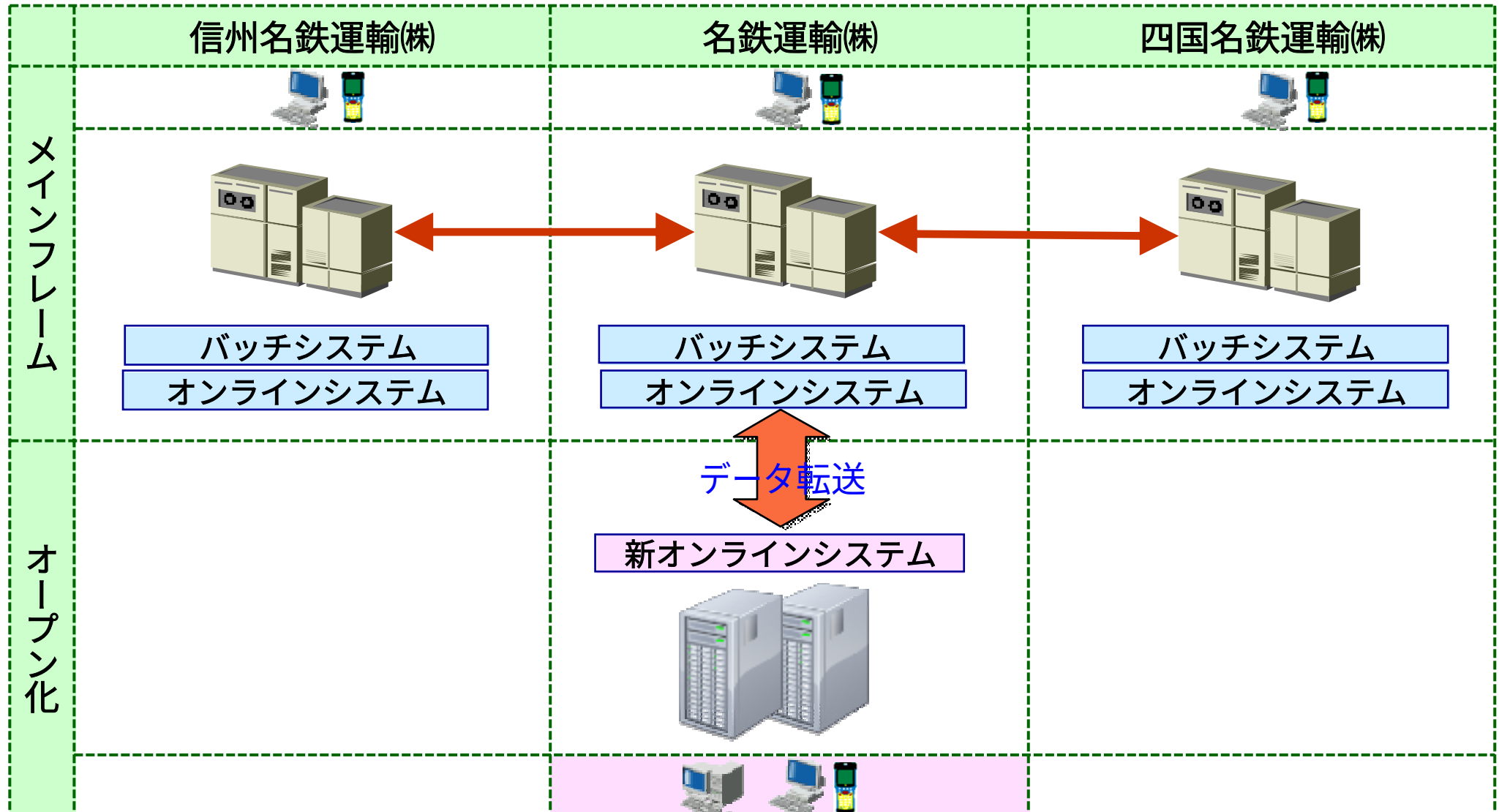
2-5.オンラインシステムのオープン化:スケジュール



2-6.オンラインシステムのオープン化:ポイント

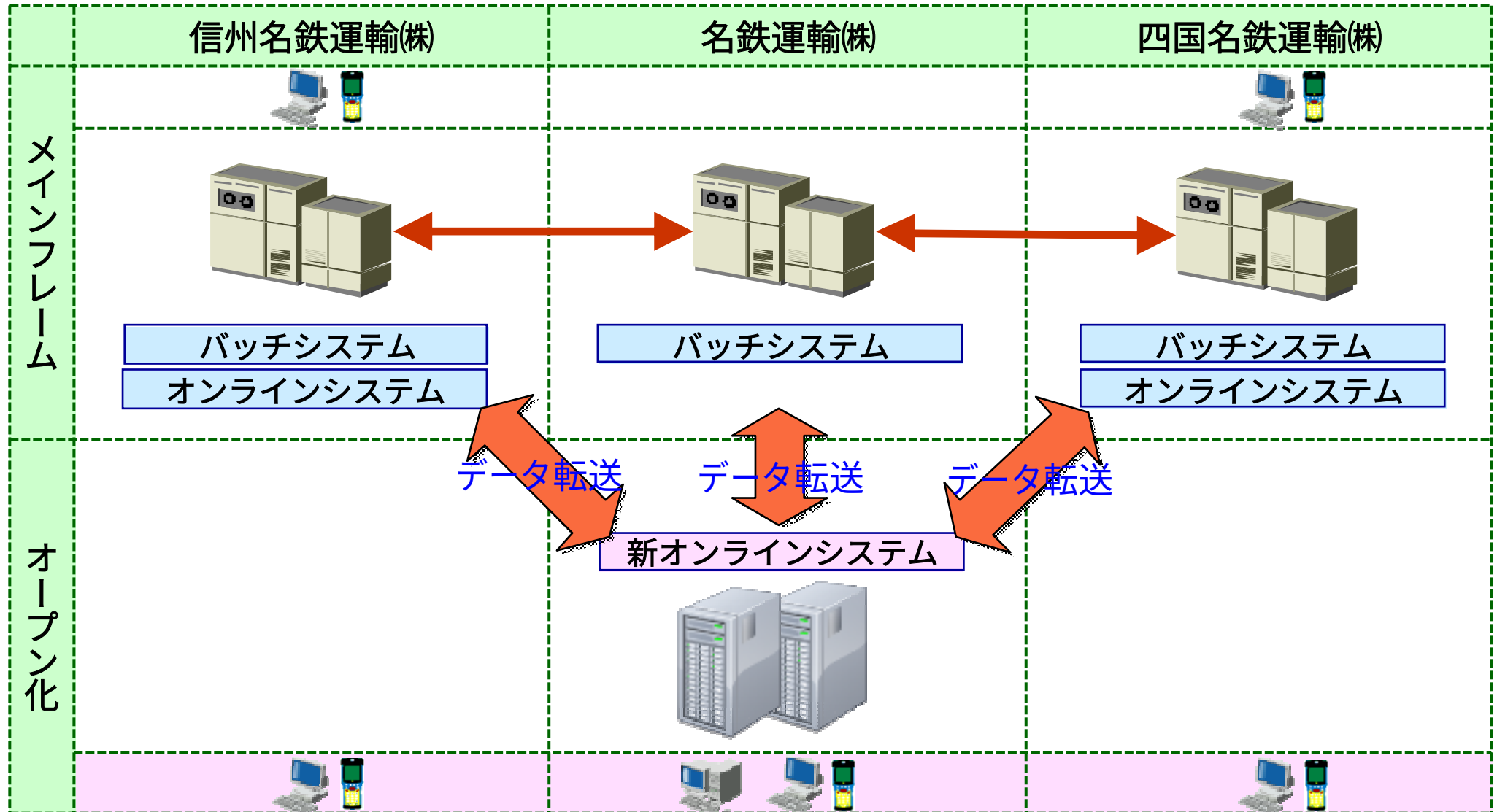
■第1段階・名鉄運輸(株)の移行

・センタサーバは名鉄運輸(株)メインフレームとデータ交換！



2-6.オンラインシステムのオープン化:ポイント

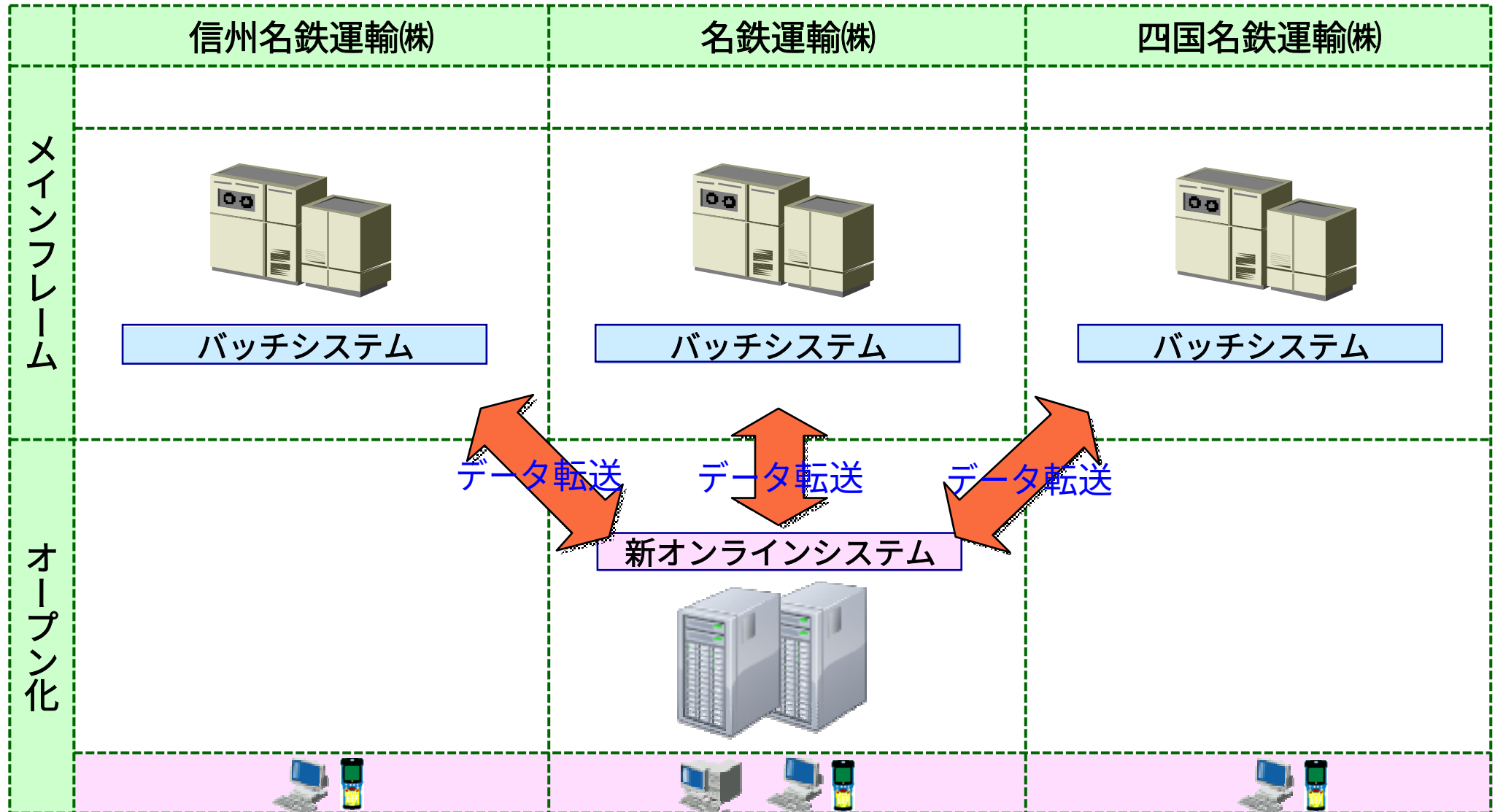
- 第2段階・信州名鉄運輸(株)、四国名鉄運輸(株)を【同時】に移行
- ・センタサーバは3社のメインフレームとデータ交換！



2-6.オンラインシステムのオープン化:ポイント

■第3段階・全社の移行がすべて完了

- ・センタサーバは3社のメインフレームのバッチシステムとデータ交換！



2-7.オンラインシステムのオープン化:まとめ

■当時を振り返って

- ・オンラインプログラムは、「アセンブラ」であったが、COBOLで再構築（リビルド）を行った。
- ・オンラインプログラムはビット操作が多用され、独自のデータ保持方式であったため、COBOLの言語仕様に適用させるのに苦労した。
- ・当時のSE、PGはCOBOLを知る要員も多く、人的資産（ノウハウ）を有効に活用することができた。
- ・オープン化後10年が経過したが、いまだ現役であり、安定稼働をしている。
- ・3台のメインフレームで稼働していたオンラインシステムを、1台のサーバに集約したが、一番の課題であったデータの整合性、連続性を保った移行がスムーズに行えた。
- ・再構築前と比較し、端末の操作方法や、現場（ドライバーなど）の運用が変更になったが、一部機能をパイロット版として先行導入したことで、スムーズに全体移行を行うことができた。

1. 基幹システムオープン化の背景
2. オンラインシステムのオープン化
3. バッチシステムのオープン化
4. Webシステムとの連携
5. まとめ

3-1. バッチシステムのオープン化：課題

■ オープン化への課題

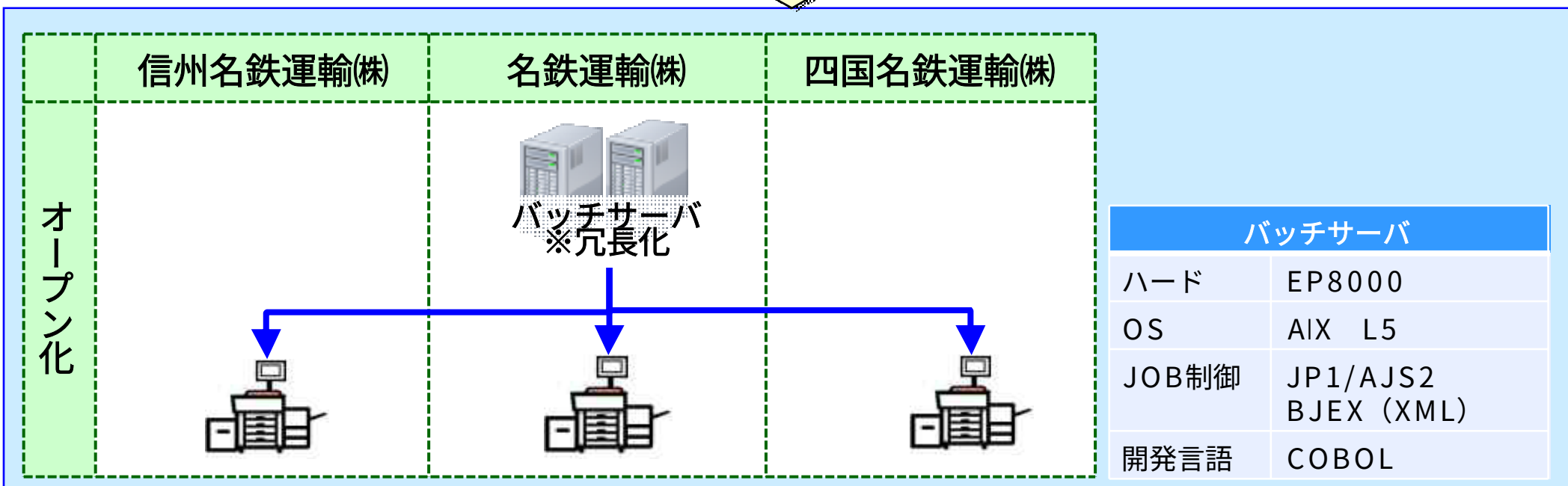
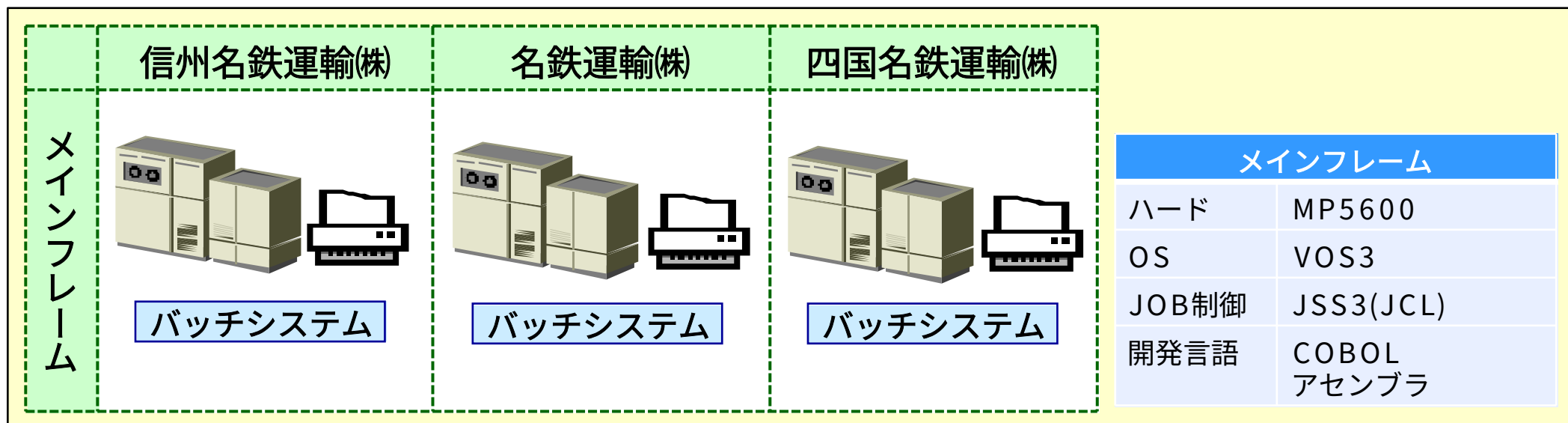
- ・ 処理結果のタイムリーな把握やデータ化、ペーパーレス化
- ・ 請求書や管理資料等を営業所、支社へ出力
- ・ 最新IT技術の連携が困難
- ・ 運用・保守管理の硬直化、コストの肥大化

操作性・運用性・設計思想は維持したい！
オープン化手法として【リホスト】を採用！

【活かす開発】

メインフレーム資産を活かす
業務ノウハウを活かす
設計思想を活かす

3-2. バッチシステムのオープン化：移行パス



3-3. バッチシステムのオープン化：環境選定

■ バッチジョブ実行基盤

- 使い慣れたJCLと同様の定義で移行が可能
- shellと比較しジョブ定義の平準化が可能
- コンバートにより作業負担が大幅に軽減

BJEX

※uCosminexus
Batch Job Execution Server

■ ジョブ管理

- 大規模処理のジョブ制御（JOBNET）が可能
- スケジュール管理が可能

JP1/AJS2

■ ホスト固有言語 (NHELP)

- COBOLへの移行に比べ工数削減が可能
- 使い慣れた定義が今後も利用可能

**NHELP実行支援
ライブラリ**

■ 開発言語

- COBOLは「業務ノウハウ」⇒有効活用
- 基本的にストレートコンバートが可能
- COBOLは進化し続けている

COBOL2002

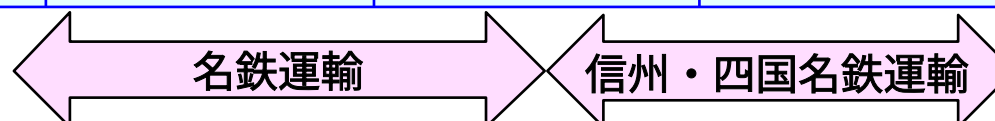
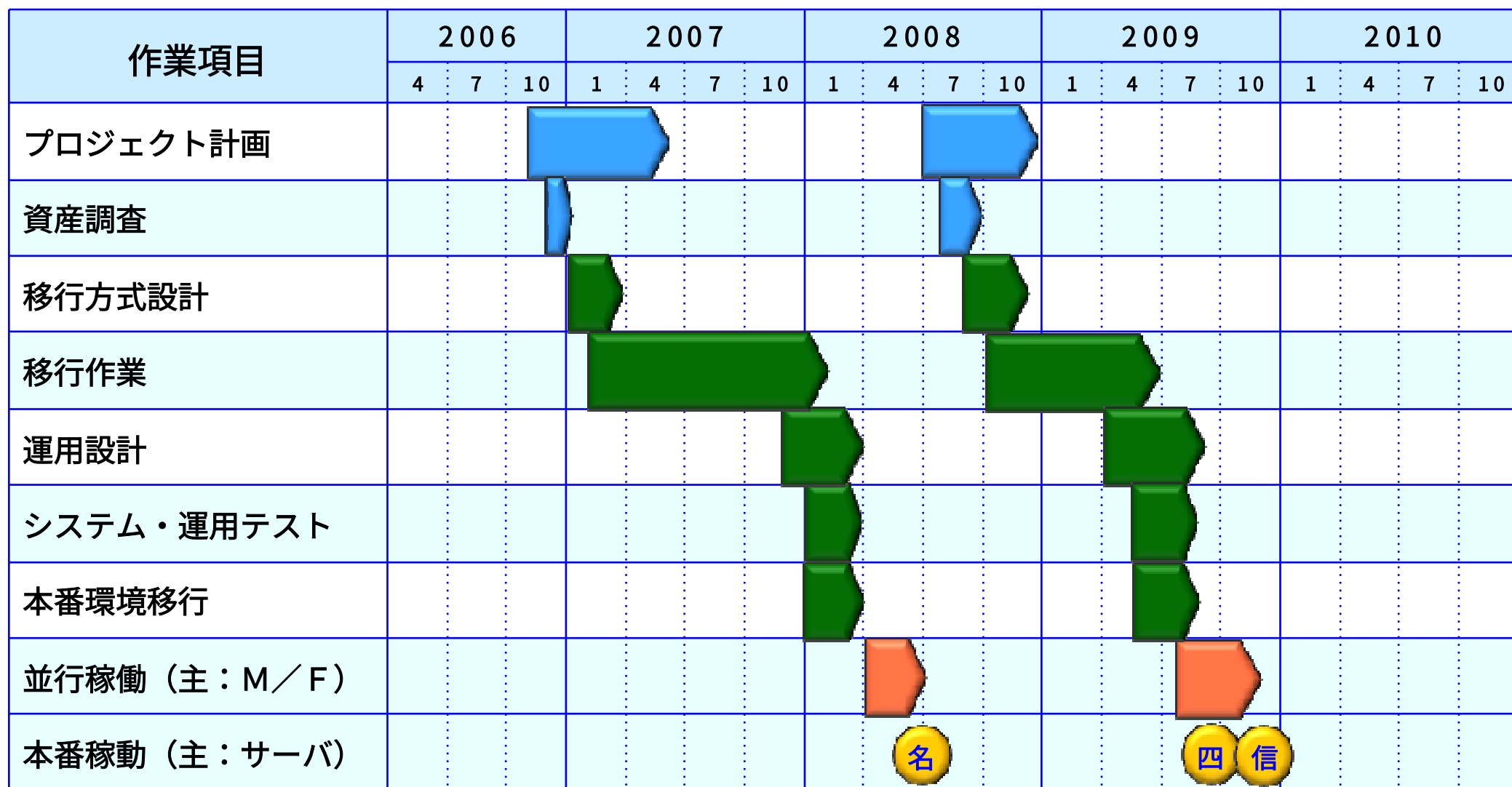
3-4. バッチシステムのオープン化: 規模

作業項目	名鉄運輸	信州名鉄運輸	四国名鉄運輸
アセンブラ ⇒ COBOL化	1 1 9本	8 1本	5本
COBOLの移行	3, 0 9 8本	2, 2 4 1本	2, 1 3 0本
COBOLの移行 (帳票出力系)	7 8 8本	5 0 0本	5 4 2本
NHELP移行	0本	5, 2 0 5本	2, 8 2 1本
JCL移行	1, 7 7 6本	2, 1 0 5本	1, 4 0 7本
データ移行 (ディスク常駐)	7 0 0	1 0 2	2 4
データ移行 (外部媒体[CGMTなど])	1, 6 0 0	1, 1 0 0	1, 5 0 0
VAN伝送設定 (外部システム連携)	1, 0 0 0 ID	3 2 2 ID	2 5 2 ID
書式オーバーレイ	1 2 4 帳票	4 6 帳票	1 0 4 帳票
帳票移行 (設計帳票)	4 0 帳票	1 6 帳票	1 7 帳票

全体作業工数

名鉄運輸	信州名鉄運輸	四国名鉄運輸	合計
376人月	208人月	151人月	735人月

3-5. バッチシステムのオープン化: スケジュール



3-6. バッチシステムのオープン化：移行プロセスの概要①

プロセス	作業項目	作業内容（主な作業）
移行計画	経営戦略とのマッピング	基幹システムの更新は、ユーザにとって重要な経営課題 ⇒将来像、移行方式、移行スケジュールを明確に
	移行総費用の算出	プログラム、JCL、ファイルシステム、帳票等の変換工数、及び本番移行作業工数の算出
	移行後の運用コスト算出	オペレータ、運用要員の役割見直し、ハードウェアの保守費用、ハウジング費用を算出。また削減効果を定義
資産棚卸	資産棚卸	- 漏れなく移行を行うため、システム資産を詳細に分析 - システム全体の規模や複雑さを明確に - 移行の必要性を明確に（移行作業の無駄を防止のため）
	資産の取りまとめ	- プログラム言語・本数、ツールの種類を整理 - 入出力環境・・・ファイル、帳票、外部媒体を整理 - 実行環境・・・JCL、JOBNET、JOBスケジュールを整理
	移行性分析	- 単純に変換できないもの、新規開発対象を調査 - プロトタイプを用いてツールを調整し、変換パターンを定義 - 変換パターンの検証
移行方式設計	移行の要件定義	- データ保有方法 ⇒移行後のデータ形式、世代管理方式、バックアップ方式 - 帳票の作成方式、管理手法 - JOB実行の管理手法 ⇒JOB実行の自動化、スケジューリングの手法 - 外部システムとのデータ連携 ⇒現行メディア見直し、伝送化による通信方式 - 新システムの性能要件 ⇒メインフレーム以上の処理能力確保 - 運用方式 ⇒開発手法、臨時対応、開発環境から本番環境への移行

3-6. バッチシステムのオープン化：移行プロセスの概要②

プロセス	作業項目	作業内容（主な作業）
移行方式設計	移行仕様の策定	- 各移行対象の変換方法を漏れなく明確に - 自動変換及び、自動変換不能ソースの移行方式 - ソースコードの変換方式 制御コード、文字コードの変換仕様 JCL⇒XMLの変換方法、JOBNETの構築方式 - ユーティリティ、NHELPの移行方式
移行作業	作業手順策定	移行作業の手順を策定（自動・手動部分を明確に）
	資産のオープン化	作業手順に沿って、ソースのオープン化実施
	単体テスト	オープン化されたソースが動作するか確認
運用設計	運用方式設計	JOB自動実行とスケジュール変更に柔軟な対応を可能に 運用カレンダーによるスケジュール実行 処理実行日が固定されていない月次処理 突発的な処理日のスケジュール変更 必要に応じて実行する臨時処理の実行手順
	運用体制・ルールの見直し	- 開発と本番それぞれの環境における運用、移行ルールの策定 - 移行後のオペレーターの作業分担の見直し 使用するデータの正当性チェック手順 JOBの開始終了の確認方法、障害時対応
テスト	総合・運用テスト	- 帳票の出力結果の照合 - 更新後のファイルの照合 - 本番運用を想定したシステム全体の動作検証 JOBのセットアップと実行 デイリー処理から月次処理へのデータ連携 外部システムとのデータ授受 データのバックアップ、リストア

3-7. バッチシステムのオープン化: ポイント～JCL移行

メインフレームから
JCL取得
(テキスト形式)

変換①
BJEX動作不可
部分の変換

変換②
JCL⇒XML

変換③
XML⇒XML
独自処理組込み

◆メインフレームのJCL

```
//STP020 EXEC PGM=PGM100
//FILE1 DD DSN=VOL001.DATA1,
// UNIT=DASD,VOL=SER=VOL001,DISP=(OLD,KEEP),
// DCB=(RECFM=FB,LRECL=300,BLKSIZE=6000)
//FILE2 DD DSN=VDATE1,DISP=SHR
//FILEA DD DSN=VOL2.DATAA,
// UNIT=DISK,VOL=SER=VOL002,DISP=(RNW,KEEP
// DCB=(RECFM=FB,LRECL=195,BLKSIZE=1950),
// SPACE=(TRK,(15,15),RLSE)
//SYSOUT DD SYSOUT=*
/*
```

◆オープン環境への変換後JCL (XML)

```
<STEP NAME="STP020">
  <EXEC PGM="PGM100" LANG="COBOL" />
  <DD NAME=" FILE1 " TYPE="FILE" DSN=" VOL001.DATA1.dt " DISP="OLD,KEEP" />
  <DD NAME=" FILE2 " TYPE="FILE" DSN=" VDATE1.dt " DISP="SHR" />
  <DD NAME=" FILEA " TYPE="FILE" DSN=" VOL2.DATAA.dt " DISP="RNW,KEEP" />
  <DD NAME=" SYSOUT " TYPE="SYSOUT" SYSOUT="*" />
</STEP>
```


3-8. バッチシステムのオープン化: ポイント～COBOL移行

メインフレームから
COBOL取得
(テキスト形式)

変換
自動変換
※独自開発ツール

修正
自動変換不能
部分の手修正

◆自動変換ツールの変換例 (主なもの)

変換項目	変換概要
チャネル定義	チャネル定義の行を削除
漢字コード(16進)	16進記述された「VALUE句」削除
プリンタ制御コード	制御コードの変換
リターンコード	JCLに返すリターンコードの変換
EVALUATE文	複数条件の場合の記述方の変更
PICTURE句	符号なしパック⇒符号有りパック
VSAMステータス	ISAMのステータスに変更
サブプログラム呼出	呼出し方の変更

**自動変換（削除）箇所には、
運用性を考慮し、すべてコメント行を挿入**

◆変換後COBOLソース内のコメント例

000000* 2008/08/29 ツール自動変換 [005600] チャネル定義削除対応

000000* 2008/12/22 ツール自動変換 [003300] パックエリアサイン付対応

3-9. バッチシステムのオープン化：ポイント～ツール開発

■メインフレームでの運用をオープン環境でも踏襲する目的で 新たに運用ツールを開発！

◆主な開発ツール

ツール	処理概要
データ世代管理	・バックアップデータの世代管理 ・CGMTなどの外部媒体で管理していたデータを、ディスク上で管理 ・指定した世代数のデータを保管 ・保管データから、指定データを抽出
パンチデータ管理	・委託業者から受領したパンチデータの保管 ・パンチデータの利用管理（不正利用等の防止）
ユティリティ	・Wait（処理待ち）機能 ・ファイル分割／結合機能 ・データの簡易印刷機能 ・データ複写機能 などなど
JOBスケジュール管理	・月間JOB実行スケジュールをマスタ化 ・当日実行JOBを抽出し、JP1コマンドを生成して自動実行

今まで培った運用ノウハウ、運用設計思想を活かし、新たに構築

3-10. バッチシステムのオープン化：まとめ

■ 当時を振り返って

- ・ 今回は3社のオープン化を順次着手した。
COBOLやJCLの変換ツールも、徐々に精度があがり手作業部分が大幅に削減された。
- ・ COBOLそのものが、業務ノウハウであり、強みでもあり、オープン化手法「リホスト」は、その蓄積された資産を十分活かすことができた。

■ 効果

- ・ オープン化移行後は、各社ともにJOB実行時間が、平均して1／3程度に短縮。
- ・ 3社のサーバ共同利用（インフラ共用）により、運用・保守要員の削減や、インフラ維持費用の削減などを実現し、運用コストがダウン！！

■ まとめ

- ・ 20～30年前のプログラムがまだまだ現役であること、逆に言うと引退させなくても良いという点から、あらためてCOBOLの保守性の高さを感じた。

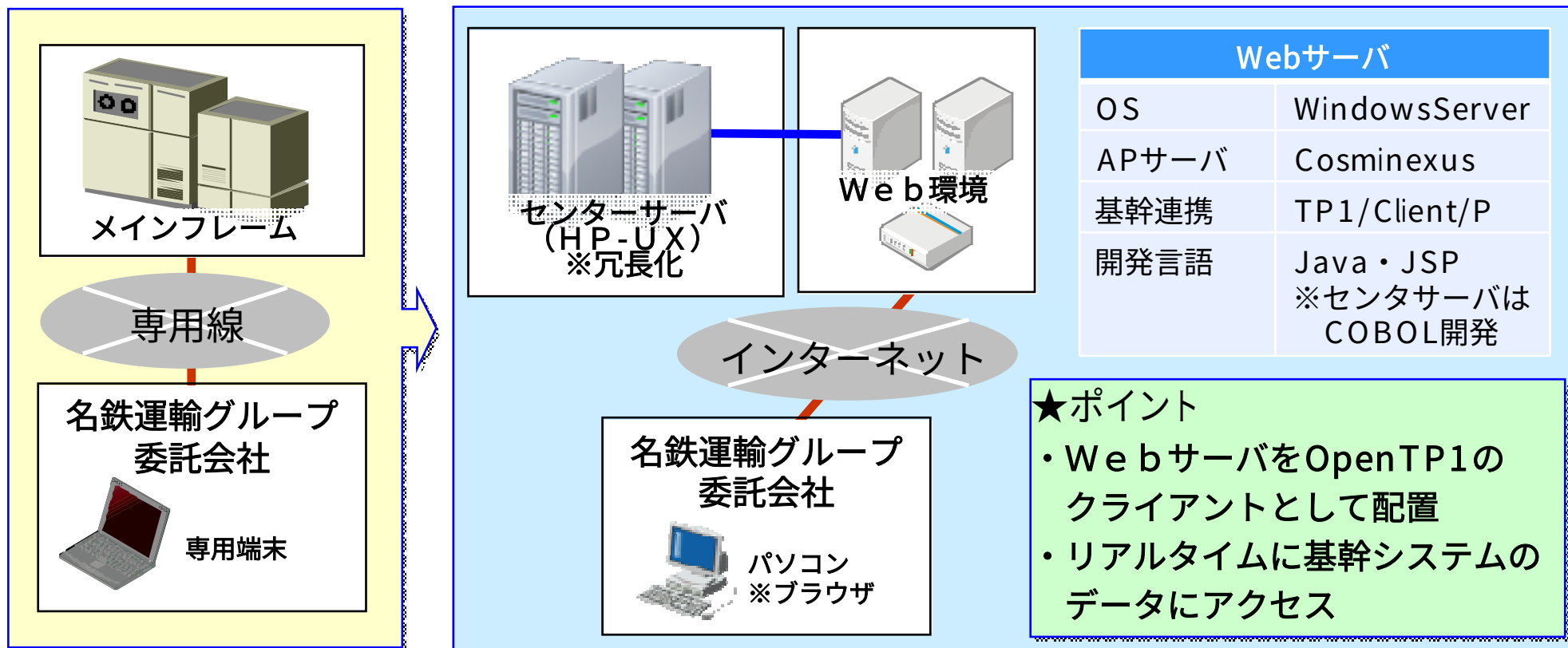
1. 基幹システムオープン化の背景
2. オンラインシステムのオープン化
3. バッチシステムのオープン化
4. Webシステムとの連携
5. まとめ

4-1.Webシステムとの連携:概要

■開発の背景

- ・基幹オンラインシステムのオープン化に伴い、「付帯簡易端末システム」のリプレースが必須であり、2002年4月に開発開始。
- ・物流業界における「ビジネス環境の変化」「同業他社の動向」「利用していただくお客様への先進的イメージ」などを考え、Webシステムとしての【リビルド】を選択。

■構成の新旧比較



4-2.Webシステムとの連携：開発のポイント

■開発環境の選定（当時）



◆Webのフロントエンド部分 ⇒ Java (Cosminexus)

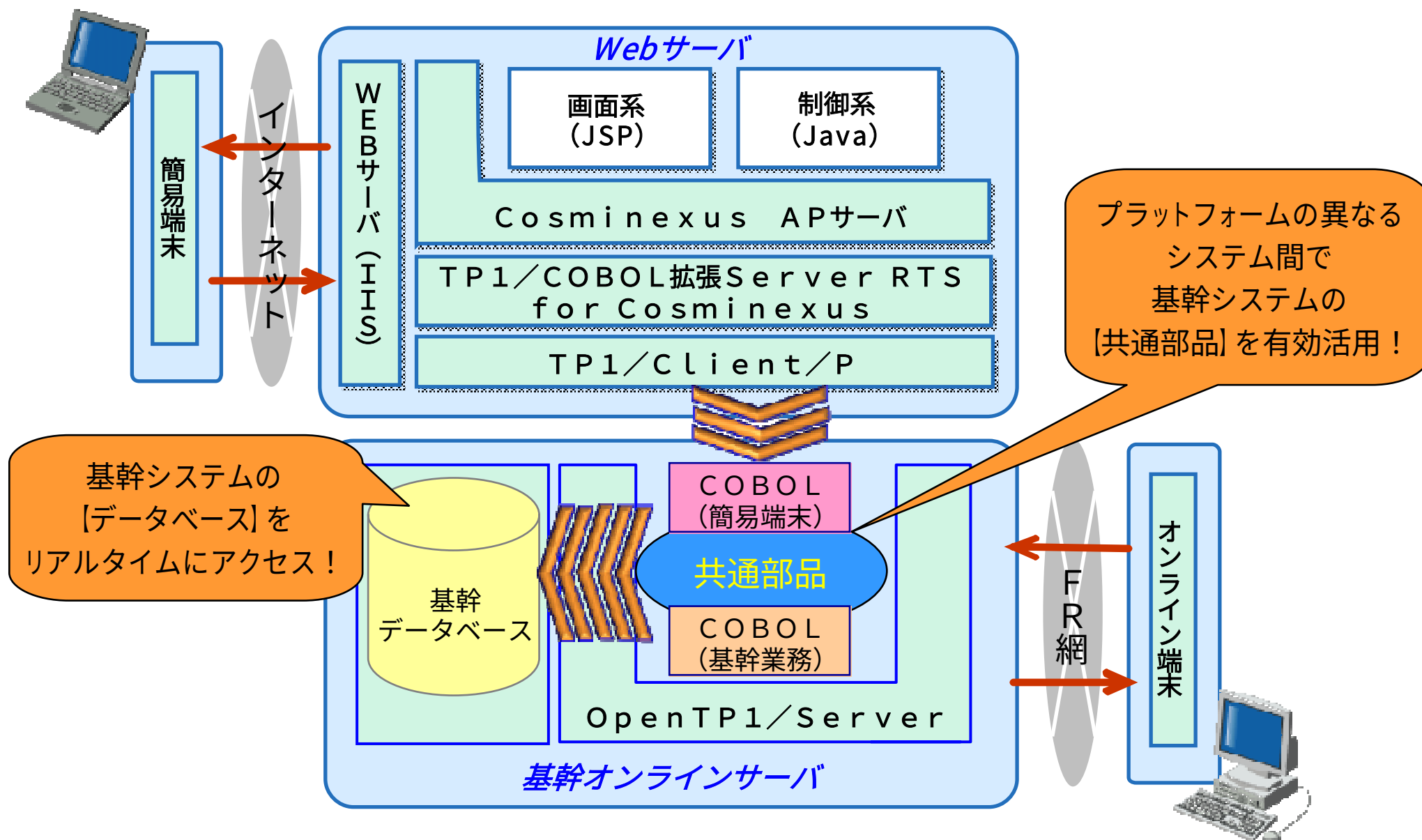
主な選定理由：サーバサイドにて一元管理でき、基幹システムと連携可能

◆ビジネスロジック(バックエンド部分) ⇒ COBOL

主な選定理由：工数の掛かる業務ロジックは、既存資産（人／ロジック）を
最大限に有効活用

- COBOLとJavaとそれぞれの開発要員を配置し、完全分業が可能
※開発要員間の接点は連携インターフェースのみ
- 既存資産を活かしながら、Web技術の取込み
- 基幹オンラインシステムの共通部品を有効活用
- UIが変更になったのみで、機能は踏襲

4-3.Webシステムとの連携：構成詳細



4-4.Webシステムとの連携：まとめ

■当時を振り返って

- ・Webシステム構築は初めてで、且つ3ヶ月という短期開発が条件でもあった。しかし、主となるビジネスロジックをCOBOLで開発したことにより、蓄積されたノウハウの有効活用ができ、品質の高いシステムを遅延なく導入できた。
- ・当時、COBOL資産を活かしたWebシステムは少なかったため、「日経システム構築」(現：日経SYSTEMS)のケーススタディに掲載していただいたり、COBOLセミナーの講演をしたり、一時期だけ有名人となった。

■今後…

- ・Web環境のリプレースを実現し、今後は顧客サービスの向上、現場作業の効率化／スピード化などを目的としたシステム拡張をして行きたい。
当然、基幹オンラインシステム側は、COBOLを活かす開発を続ける。

1. 基幹システムオープン化の背景
2. オンラインシステムのオープン化
3. バッチシステムのオープン化
4. Webシステムとの連携
5. まとめ

5-1.まとめ：今後の展開

■基幹システムのリプレースの際に・・・

将来、オープン環境で稼働する基幹システム（オンライン・バッチ）のリプレースがやってくることも考えられる。

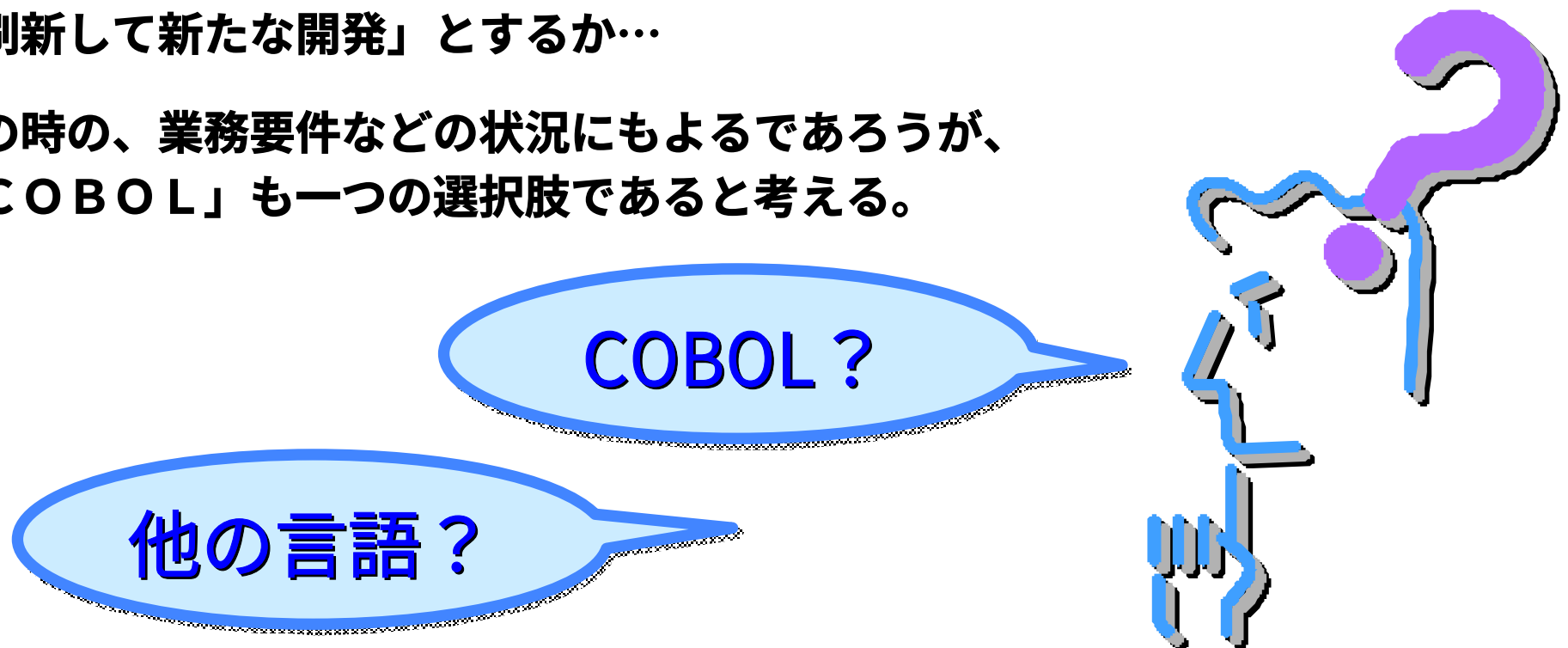
その際、開発環境の選定はどうなるだろうか…

「COBOL？」それとも「他の言語？」

今回のオープン化同様に、「活かす開発」とするか…

「刷新して新たな開発」とするか…

その時の、業務要件などの状況にもよるであろうが、
「COBOL」も一つの選択肢であると考える。



5-2.まとめ:COBOLについて①

■2002年12月の「第4回インターネット時代のCOBOL活用セミナー」資料より

◆多種多彩なCOBOL開発環境

現在でもベンダー各社より、COBOL開発環境を提供
Webに対応する環境の充実、技術の向上

⇒ 今後もCOBOL需要は続くと思われる

◆COBOL要員の高齢化

COBOL開発要員が年々高齢化している

⇒ 新たな人材育成体制の確立

● 8年経った現在も、同様にベンダー各社からCOBOL環境を提供

- ・ VisualStudioへのプラグイン
- ・ クラウド上でCOBOL資産が稼働できる時代

● COBOL要員はますます高齢化へ？

- ・ アサインする人材は、ベテラン世代
- ・ COBOLは社内教育カリキュラムに無い ⇒ スキルトランスファー頼み

COBOL開発環境は進化し続けるが、人材が増えてないと感じる。

⇒ 8年前と同じ状況？・・・重要な課題

5-3.まとめ: COBOLについて②

■ COBOLは誕生50周年

- ・他の言語にはない、長期にわたり基幹システムを支える言語
- ・まだまだ、多くの企業で利用され続けている
- ・「リホスト」によるマイグレーション実施により、オープン環境でもCOBOLは利用されている

なぜ？

- ・読解し易く、保守性が高い
- ・スキルレベルによる品質の差が小さい
- ・大量レコード処理が得意
- ・言語仕様が安定且つ、時代のニーズに合わせ進化
- ・幅広いプラットフォームで動作

基幹システムのオープン化を検討中の皆様、
「既存資産（COBOL）を活かす開発」も検討してみてください。
COBOLはまだまだ現役です。

ご清聴ありがとうございました



お問合せ先：solution@meitetsucom.co.jp