

# *Linux*環境での COBOLマイグレーション

2007年12月13日

エステー株式会社  
ITグループ 照井琢磨



# 目次

- 会社紹介
- マイグレーションの背景と選択した手法
- スケジュール案とその実際
- 移行手順
- 導入効果
- プロジェクトを振り返って

# 会社概要



会社名	エステー株式会社 (8月1日、エステー化学株式会社からエステー株式会社に商号変更)
設立	1948年8月31日
主な事業内容	消臭芳香剤・防虫剤・手袋・除湿剤などの製造販売
従業員数	連結 602名、単体356名 (2007年3月31日現在) パートタイマー・嘱託を除く。
本社	東京都新宿区下落合
資本金	70億6,550万円
上場証券取引所	東京証券取引所 市場第一部
発行済株式総数	29,500,000株

## 日用品業界に属するメーカー

衣類ケア (防虫剤)



湿気ケア  
(除湿剤)



エアケア (消臭芳香剤)



脱臭剤



セルフメディケーション・  
美容・健康をサポート

ハンドケア  
(手袋)



クリーナー



サーモケア  
(使いすてカイロ)



# システムの歴史



2007年 新体制・エステーへ商号変更

2007年 新環境へ

2000年 GS8300をホスティング

1995年 FACOM - M1700にリプレース

1990年 FACOM - M760にリプレース

1991年 東証第1部に指定

1989年 プラネット物流に参加

1986年 UNIVAC - 2200にリプレース

1986年 東証第2部に上場

1985年 プラネット参加

1982年 FACOM - M140Fにリプレース

1975年 FACOM - Vにリプレース

1970年 FACOM230 - 10導入

1948年設立

## ■ 処理能力の限界

- 日次処理の終了が27時(午前3時)になるときもあった

## ■ H / W増設が費用的に困難

- ディスク容量の慢性的な不足

## ■ 情報系サーバーへのデータ送信量増大

- もっともCPUを消費する処理が情報系サーバーへのデータ送信

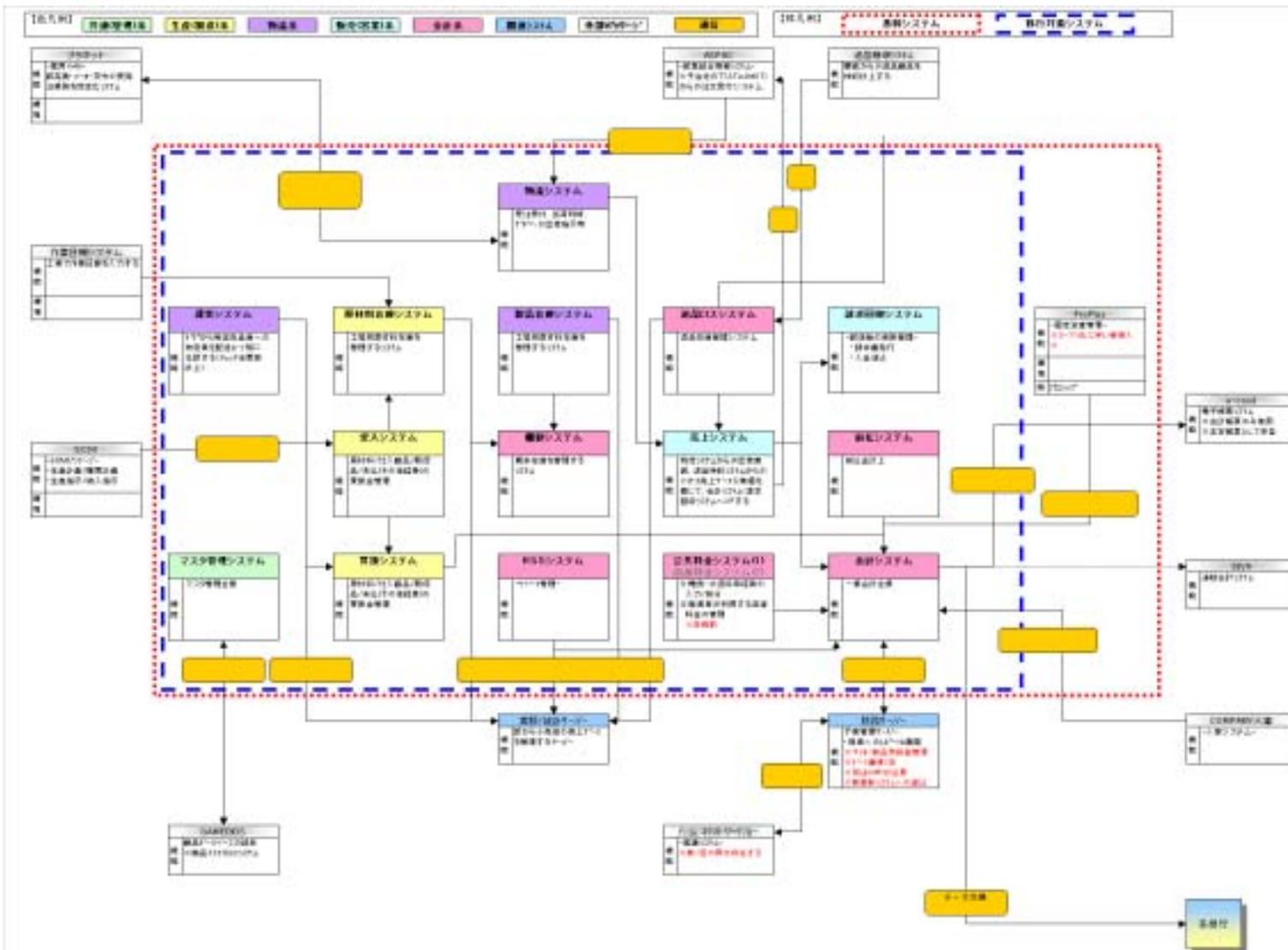
## ■ ホスティング契約期限

- ホスティング契約先から契約は2007年3月末までとの通告を受けた。



**汎用機リプレースを含めた再構築検討が必須**

# 移行対象システム



## ■ 処理速度、レスポンスの向上

- 現行機のサービスレベルは維持が最低条件。より速いシステム設計を。

## ■ 汎用機レベルの可用性の確保

- 全基幹システムを移行するため、安定したサービス提供が必要
- 障害発生時にも迅速に復旧できる(30分以内の復旧)

## ■ 必要に応じて機能アップできる拡張性の確保

- 処理能力やディスク容量が不足した場合、拡張が容易にできる
- 他システムとの連携が容易にできる

# 方法・効果・費用・期間の比較

対応方法		メリット	デメリット	追加費用	期間	運用費用 3年間	現場 適用度	IT 難易度
オープン系	ERPパッケージ 導入による 全面刷新	<ul style="list-style-type: none"> <li>・業務プロセスが世界標準化</li> <li>・全体最適化がはかれる</li> <li>・リアルタイム処理が可能</li> <li>・最新技術が導入しやすい</li> <li>・統合的なシステムの実現</li> </ul>	<ul style="list-style-type: none"> <li>・業務プロセスを適応しないと多額の費用が発生</li> <li>・ライセンス料、ソフト保守料高い</li> <li>・自社仕様の開発をしないとソフトが進化するのに対応できない</li> <li>・現場運用が破綻する危険性あり</li> <li>・現状全業務は対応できない</li> </ul>	中上	12～15 ヵ月	中上	低	中 (現場調整は高い)
	新言語による 全面刷新	<ul style="list-style-type: none"> <li>・自由度が高く、当社に最適なしくみが作れる</li> <li>・段階的な構築が可能</li> <li>・業務リスクが少ない</li> <li>・最新技術が導入しやすい</li> </ul>	<ul style="list-style-type: none"> <li>・導入に時間を要する</li> <li>・費用がとても高い</li> <li>・新言語の習得が発生</li> </ul>	大	18～24 ヵ月	小	高	高
	適応パッケージ 組み合わせ	<ul style="list-style-type: none"> <li>・導入期間が比較的短い</li> <li>・最新技術が導入しやすい</li> <li>・システムの成長性がある</li> </ul>	<ul style="list-style-type: none"> <li>・システムの統一感がない</li> <li>・費用はピンキリ</li> <li>・複数のパッケージを導入すると保守費用等が高くなる可能性あり</li> </ul>	中下	10～24 ヵ月	中下	中	中
	既存言語の変換	<ul style="list-style-type: none"> <li>・導入期間が比較的短い</li> <li>・最新技術が導入しやすい</li> <li>・費用は比較的安い</li> <li>・自由度が高い</li> <li>・言語の習得が容易</li> </ul>	<ul style="list-style-type: none"> <li>・システムの成長は自社努力</li> <li>・自社で対応する場合、期間が長引く可能性あり</li> </ul>	中下	10～15 ヵ月	小	高	低
ホスト系	現状継続	<ul style="list-style-type: none"> <li>・最も安価</li> <li>・移行に伴うリスクがない</li> <li>・堅牢性が高く安定している</li> </ul>	<ul style="list-style-type: none"> <li>・新技術に対応できない</li> <li>・開発要員が減少しており、将来性はない</li> <li>・システム拡張はオープン系</li> <li>・リアル性が低い</li> <li>・システムの成長は自社努力</li> </ul>	小	0ヵ月	中上	-	低
		<ul style="list-style-type: none"> <li>・従来のしきみ以外はオープン系で開発する</li> <li>・ハードの費用はやはり高い</li> </ul>						

## 【既存言語の変換】 方式の提案概要

### ■ 汎用機資産 (COBOL, JCL等) を活用する

- 東京システムハウス株式会社のメインフレームマイグレーションサービス (MMS) を利用

### ■ 可用性、経済性を確保するため OS は Linux

- 安価で安定性の優れた Linux を採用することで、可用性、経済性の両方を確保

### ■ ファイルシステムは Oracle

- 信頼性、他システム連携、データ利用の容易さ

### ■ 文字コードは SJIS

- 既存のオープンシステムの文字コードが SJIS

## ■ 東京システムハウス株式会社のMMSを採用

- 既存汎用機からのマイグレーション実績があり、短期間かつ正確にオープン環境に移行可能という提案を受け決定

## ■ OSはLinuxを採用

- 不安はあったが、OSとしては安定しており安価という第三者の意見もあり決定

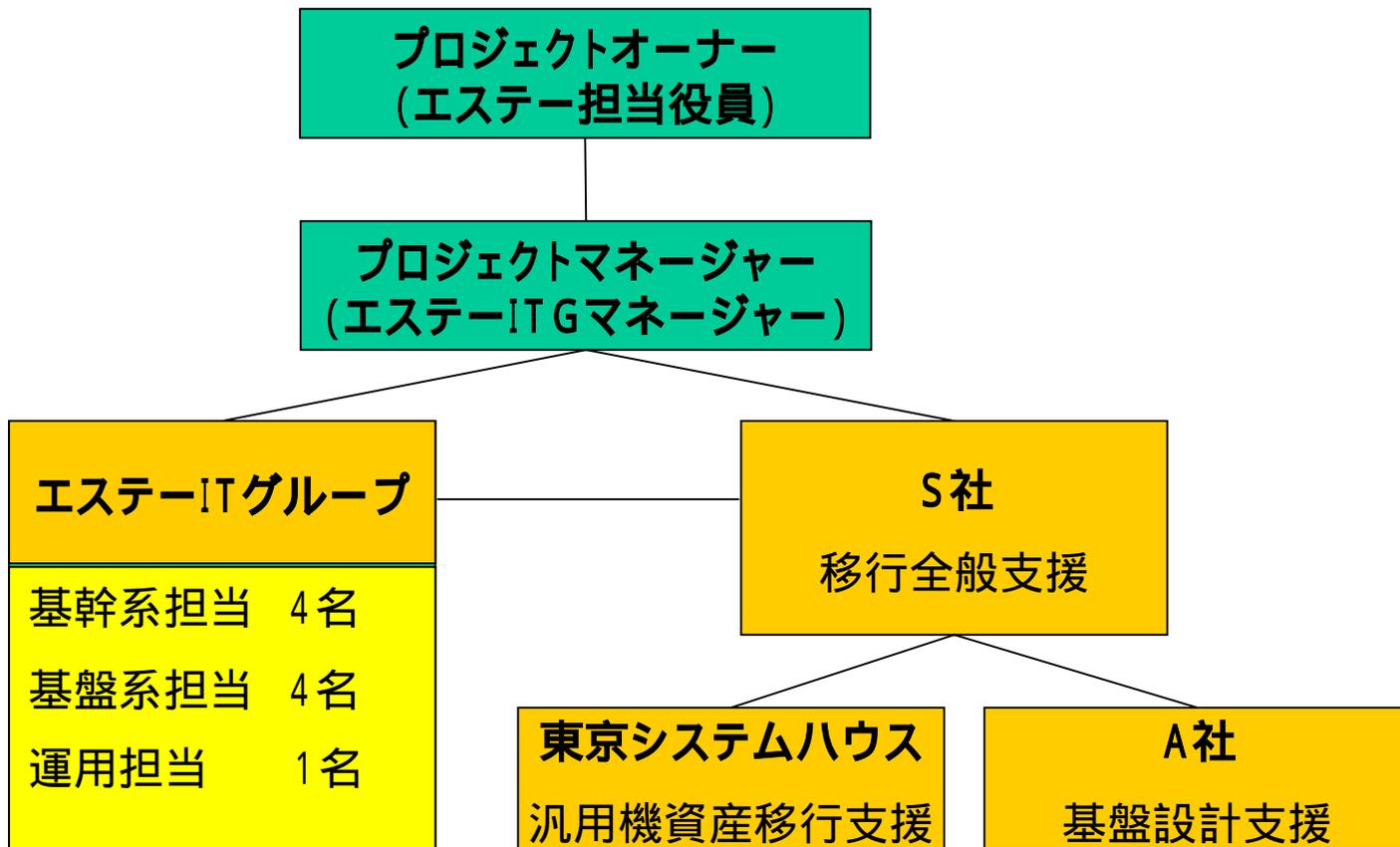
## ■ ファイルシステムはオープンCOBOL独自システムを採用

- 既存汎用機の索引ファイルの方式そのままに移行が可能との事で決定

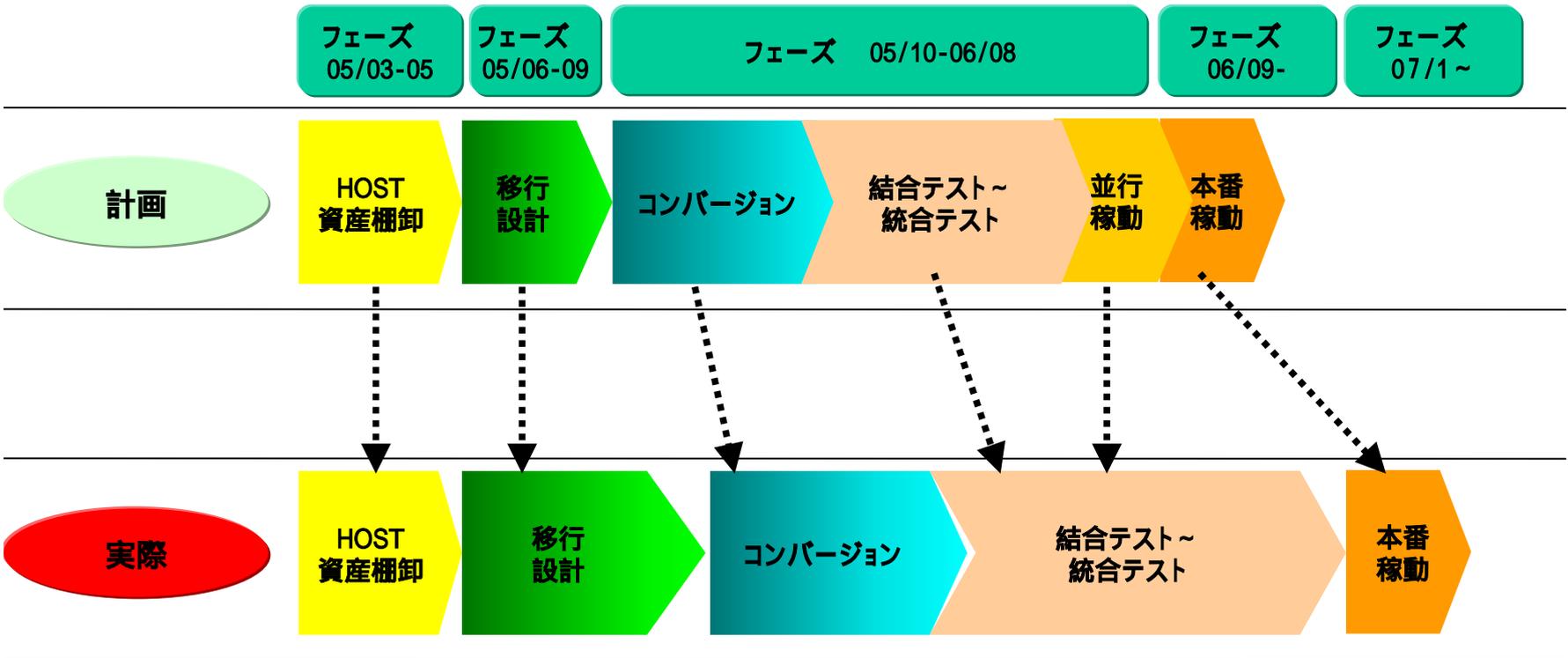
## ■ 文字コードはSJISを採用

- 扱いやすさを考えるとSJISがベストということで決定

# プロジェクト体制図



# 基幹システム移行スケジュールと実際



# フェーズ1 汎用機資産棚卸

## ■ バッチCOBOLプログラム、JCL

直近1年2ヶ月間の稼動実績から、実際に実行したバッチをピックアップ

全JCL、プログラムを実行実績有り無しに分ける

実行実績有り、無しそれぞれを担当者が全てチェックし移行対象を決定

## ■ 画面プログラム

全資産を担当者がチェックし移行対象を決定

## ■ COBOL COPY句

全プログラム中のCOPY句を解析しCOPYライブラリの資産を使用、未使用に分ける

使用、未使用それぞれをチェックし移行対象を決定

## ■ その他資産

担当者が選定し決定

# 資産棚卸結果

	未使用	使用	計	移行対象
COBOL本体	848	2219	3067	1804
COBOL COPY句	1310	2161	3471	1669
アセンブラ	68	34	102	16
JCL	751	1684	2435	1225
オーバーレイ	300	163	463	114
ユーティリティ等	0	52	52	10

## ■ 先行コンバージョンの実施

### ■ 先行コンバージョンとは

移行対象プログラム・JCLから色々なパターンのものを選択し、それらをコンバーターを使ってコンバートを行い、コンバート後のオブジェクトで処理を実行した結果とメインフレームでの実行結果を比較検証しながらコンバーターをチューニングして行くことにより、コンバーターの変換精度を向上させ、マイグレーション仕様を確定させる

80強のJCLと画面PGで実施

## ■ 各種アプリケーションソフトウェアの選定

• 帳票出力・管理、運用管理、EDI機能を実現するソフトウェアの選定

## ■ ハードウェア構成の設計

• サーバー、ストレージ、ネットワーク機器の選定と構成設計

# 移行設計フェーズの結果

## ■ 先行コンバートで発生した問題点など(1)

- **データ変換用マッピング定義の精度が悪い**  
定義間違いによる誤変換、マルチレイアウト対応モレ等

- **画面プログラムの変換の難易度が高い**  
汎用機固有の機能に依存する動作の再現が難しい

- **自動変換の限界**  
プログラム、JCLとも、自動変換できない人間の判断が必要な記述がある

- **既存ホストCOBOLとオープンCOBOLの文法と動作の差異**  
コンパイルが通らない、動作が異なる等  
既存汎用機では一度も発生しなかった潜在バグが発見された

# 移行設計フェーズの結果

## ■ 先行コンバートで発生した問題点など(2)

### ● アセンブラプログラムの機能移植

アセンブラプログラムは同機能を有するCOBOLプログラムに書き直す必要がある

### ● 可変長ファイルの扱いが汎用機と異なる

SORTでのキー位置指定がズレてしまう

### ● 既存汎用機特有ユーティリティの機能移植

VSAMファイルユーティリティなどの機能

### ● 既存汎用機特有機能の移植

JOBグループによる多重実行制御などの機能

# 移行設計フェーズの結果

## ■ 各種アプリケーションソフトウェアの選定

### ■ 帳票出力・運用システム

DURL (プリズム (当時))

BSP - RM (BSP)

### ■ 運用管理システム

eXsenju (NRI)

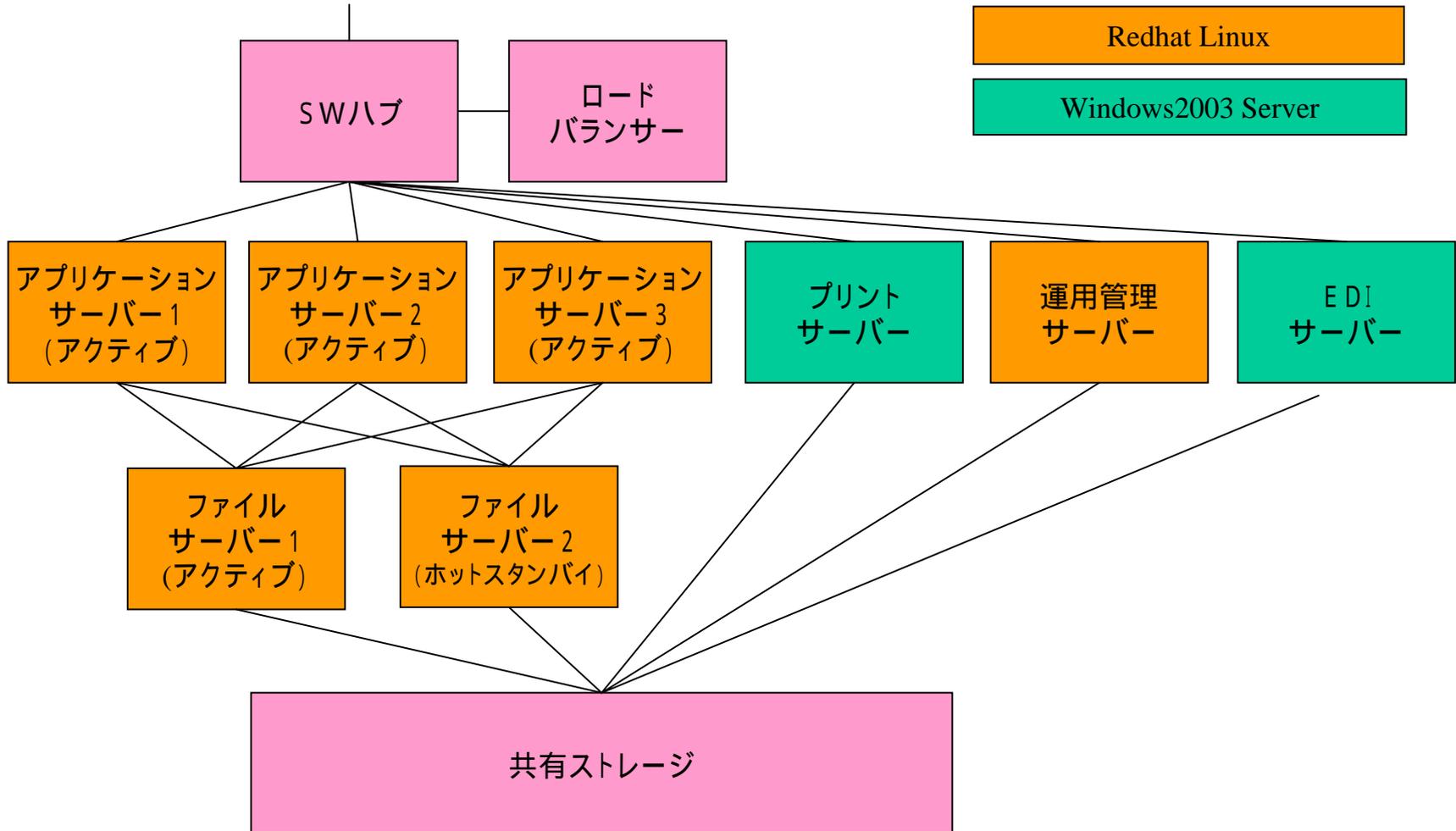
### ■ EDIシステム

NTSシリーズ (アルゴ21)

EDIマスター (アルゴ21)

# 移行設計フェーズの結果

## ■ハードウェア構成案



- 先行コンバージョンで確定した仕様で全プログラム、JCLを変換
- 変換したプログラム、JCLの受入テストの実施
- アプリケーションソフトの動作テスト、設定
- アプリケーションソフトとの連携テストの実施
- ハードウェア構成の動作検証

# コンバージョン、テストフェーズの結果

## ■ コンバージョン、結合テストで発生した問題点 (1)

### ■ コンバーターが完璧でないので、全数検査が必要

先行コンバートで色々な問題点があることがわかったので、結局全数検査を行った

### ■ 500件弱の問題点、要変更点が発生

先行コンバートで対象にしなかったプログラムに高難易度のものが存在  
画面プログラムは動作、見栄えなどの点で修正が多発

SORTの複雑なSELECT条件がそのまま移行できない(考えながら手修正)

乗算と除算が混在する計算式で計算誤差が発生

計算結果がゼロの場合 0 または -0(マイナスゼロ)になる現象が発生

トランザクション制御が既存汎用機と合わない(暗黙指定時の挙動の違い)

オーバーレイのズレ

各種アプリケーションソフトとの連携が思ったようにできない

同じキーボード(特にテンキー)操作でもカーソルの挙動が違う

画面のコピー & ペーストが1項目ずつしかできない

等々

# コンバージョン、テストフェーズの結果

## ■コンバージョン、結合テストで発生した問題点(2)

- システム構成に問題がありパフォーマンスが悪くなり、構成変更しなければならなかった

ファイルサーバーのパフォーマンスが悪く、検討の結果外す事になり、結果的にアプリケーションサーバーが1台構成となった

- 移行中にプログラムを変更しなければならないことがあった

移行資産の凍結は事実上不可能

# コンバージョン、テストフェーズの結果

## ■ COBOLの問題点(1)

### (既存ホストCOBOLとオープンCOBOLの仕様差異)

- 既存ホストCOBOLでは省略していたのに、オープンCOBOLでは省略不可

EVALUATE XXX ALSO YYY のALSOは省略不可

- 既存ホストCOBOLではANSI規格違反の記述でも正しく動作する

異なるSECTION への GO TO ジャンプ

- 異なる属性間のMOVE命令で値が勝手に変換されてしまう

Xタイプから9(数値)タイプへの転送で A が 1 に自動変換

- PERFORM UNTIL 文の書き方によりエラーが発生する場合がある

PERFORM VARYING IDX UNTIL AAA(IDX) = SPACE OR IDX > 6 ではコケる

# コンバージョン、テストフェーズの結果

## ■ COBOLの問題点(2)

### (既存ホストCOBOLとオープンCOBOLの仕様差異)

#### ■ 既存の除算式の書き方では除算例外が発生しない

ON SIZE ERROR句を書かないと発生しない(標準仕様)

#### ■ 帳票ファイルの改ページの制御文字の違い

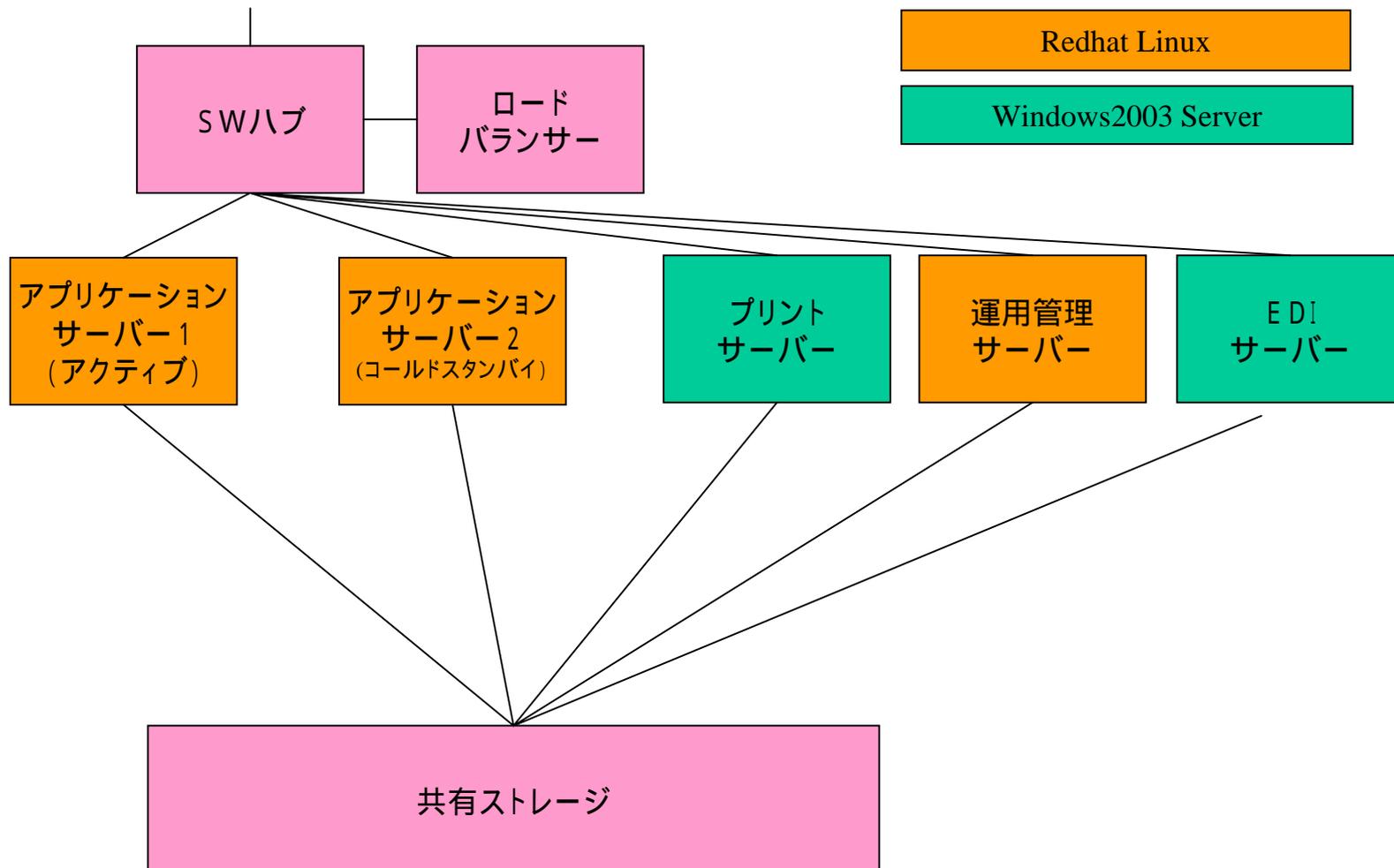
LINAGE句指定のファイルで AT EOP 条件で WRITE AFTER PAGE を実行すると  
改頁制御記号が入らない

#### ■ 半角スペースと全角スペースの違い

SJISでは半角スペースと全角スペースのコードが異なる

# コンバージョン、テストフェーズの結果

## ■ハードウェア構成



- 様々な問題が発生したが、TSH様を初め各社様と共に問題をクリアし本番稼動ができる状態になった
- ギリギリ(2006年12月)まで受入テストを行っていたため実運用テストが不十分だった
- 基幹システムの切替にもかかわらず現場説明会はほとんど行わず、イントラ上の「お知らせ」だけで済ませた
- 現場での帳票出力などのテストはほとんど実施しなかった

- 他システムとの連携などでエラーが続出した
- 帳票関係のトラブルが現象が続出した
- しかし社内の事務部門が集約されていたため、エンドユーザーサイドの混乱はあまり発生しなかった



ほぼ1週間で安定稼働

2007年1月末にホスティング契約解除

## ■ バッチ処理の高速化

アプリケーションサーバーが1台構成になったが、パフォーマンスは汎用機に比べ劇的に向上した。夜間処理の終了時間が遅くても 0時 になった

## ■ Linuxは非常に安定

本番稼動からほぼ1年経過したが、OSが原因のシステム停止は一度も発生していない  
基幹システム担当者もすんなりと受入られた

## ■ オープンCOBOLならではの機能の活用

GUI機能、Web機能を活用した画面開発などが可能になった  
Oracle等のDBMSとの接続が可能になった

## ■ 汎用機に比べかなり強力な開発環境で作業効率UP

強力なデバック機能で開発効率が大幅に向上

- COBOLからCOBOLのマイグレーションでも簡単では無い
- 早期に安定稼動できたのは、全数検査実施の成果
- Linuxの採用は正解だった
- ソフトウェア提供会社の先行きに一抹の不安あり

- COBOLはまだまだ現役でいけるし、発展もする。
- 基幹システム全面切替、Linuxの採用は相当リスクがあったがマイグレーションを行って良かった。

ご協力いただいた各社様ありがとうございました。

---

ご清聴ありがとうございました。

2007年12月13日  
エステー株式会社