

# COBOL資産を活用した UNIX,Windowsへのマイグレーション事例

富田薬品株式会社  
情報システム部  
開発2課 村上康彦

## Agenda

1. 会社紹介
2. ダウンサイジングの概要
3. メインフレーム→UNIX(オンライン・バッチ)
4. メインフレーム→Windows(バッチ)
5. メインフレーム→Windows(オンライン)
6. メインフレーム→Windows(Web)
7. COBOLの移行性

# 1. 会社紹介

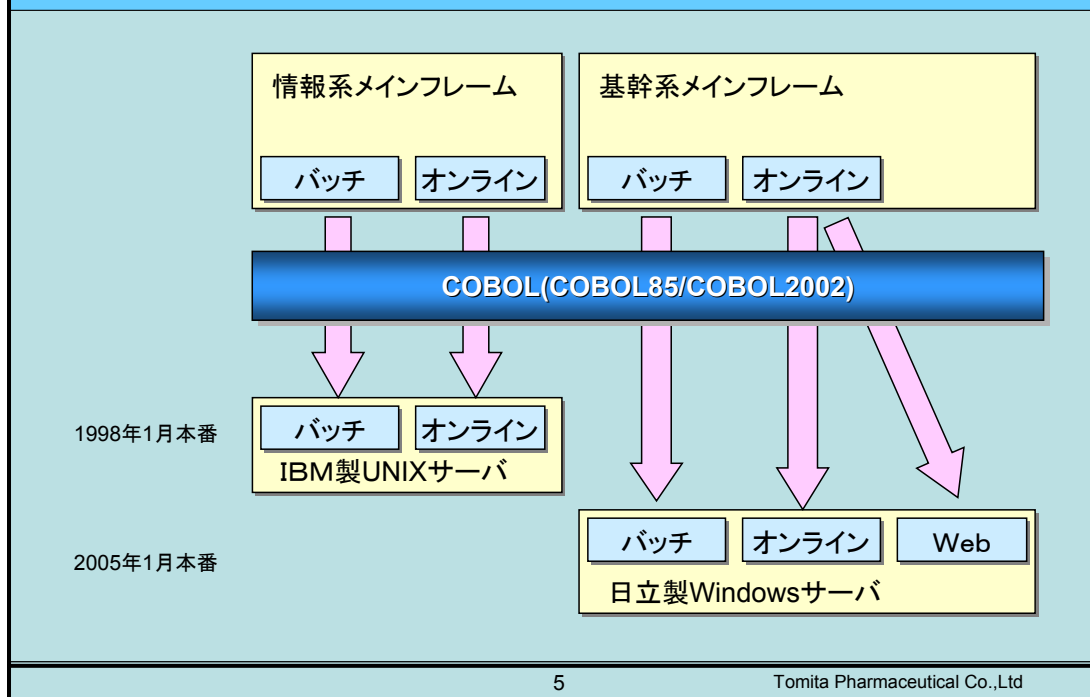
- 創業 : 明治25年8月
- 資本金 : 24億1,570万円 (平成16年3月31日現在)
- 年商 : 970億3,149万円 (平成16年3月期)
- 従業員 : 799名 (平成16年3月31日現在)
- 事業内容 :
  - ・医薬事業 ... 医療用医薬品を病院・診療所・調剤薬局等へお届けする医薬品卸売業です。
  - ・ヘルスケア事業 ... ドリンク薬、総合感冒薬などの一般用医薬品とトイレタリーなどの生活関連商品を薬局・薬店等にお届けしています。
  - ・A&S事業 ... A&S事業はAnimals & Sanitation(動物&衛生管理)を指しており、動物用・水産用医薬品並びに化成品を主な取扱品目としております。
- URL ... <http://www.tomita-pharma.co.jp>

# 2. ダウンサイジングの概要

## 2-1. ダウンサイジング概要

## 2-2. ダウンサイジングの背景とCOBOLの選択

## 2-1.ダウンサイジング概要



## 2-2.ダウンサイジングの背景とCOBOLの選択

### ■ダウンサイジングの背景

- ・競争の激化／コストダウン
- ・メインフレームの限界／オープン化技術への対応
  - PCとの親和性
  - インターネットEDI等得意先への対応

### ■COBOLの選択

**移行コスト** … COBOL資産をオープン系言語に移行するとコスト高

**移行性** …全てのプラットフォームで稼動

**スキル** …COBOL技術者を他のオープン系言語へシフト？

**保守性** …システム要員削減→他人のプログラム保守

**将来性** …COBOL2002の策定

## 3. メインフレーム→UNIX(オンライン・バッチ)

### 3-1. 移行概要

### 3-2. 移行結果

## 3-1. 移行概要

### ■ 移行の背景

- ・メインフレームの老朽化・処理速度の悪化
- ・WindowsNTサーバへの移行には未だ不安があった

### ■ 移行概要

|         | 移行前                  | 移行後                    |
|---------|----------------------|------------------------|
| OS      | MVS                  | AIX(UNIX)              |
| DBMS    | DB2                  | DB2                    |
| OLTP    | CICS                 | CICS                   |
| 言語      | COBOL85(VS COBOL II) | COBOL85(COBOL for AIX) |
| ジョブ制御言語 | JCL                  | shell                  |

※初めてのオープン系への移行であり、ストレートな移行が目標

## 3-2.移行結果

### ■移行結果

#### オンライン

・・・COBOLのCICSインターフェース、画面インターフェース等を修正。CSP等の簡易言語があったため思ったよりも修正に手間取った。

#### バッチ

・・・COBOLの変更は殆ど無かったが、JCLからshellへの変更に手を取られた。

- ・shellは高機能だが使いこなすにはスキルが必要
- ・COBOL技術者にはストレスが大きかった



JCL移行の大切さを痛感

## 4. メインフレーム→Windows(バッチ)

### 4-1.移行概要

### 4-2.JP1/AJS2の採用

### 4-3.COBOLO開発ツールの構築(1)

### 4-4.COBOLO開発ツールの構築(2)

### 4-5.電子帳票の採用(1)

### 4-6.電子帳票の採用(2)

## 4-1.移行概要

### ■ 移行の背景

- ・Windows Serverの安定化により、メインフレーム完全撤廃を決断
- ・COBOL2002の策定・発売によりCOBOL2002による移行を決定

### ■ 移行概要

|         | 移行前     | 移行後                              |
|---------|---------|----------------------------------|
| OS      | VOS3/AS | Windows Server 2003 EE × 2(MSCS) |
| DBMS    | PDM II  | Oracle 9i EE                     |
| 言語      | COBOL85 | COBOL2002                        |
| ジョブ制御言語 | JCL     | JP1/AJS2                         |

### ■ 移行結果

- ・バッチジョブ実行時間が60～80%短縮。
- ・COBOL2002の移行に関しては問題なし。  
※ただし、COBOL2002の新機能に関しては今回は使用せず。

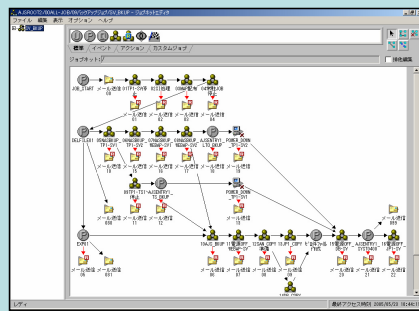
## 4-2.JP1/AJS2(統合システム運用管理)の採用

### ■ JP1/AJS2のメリット

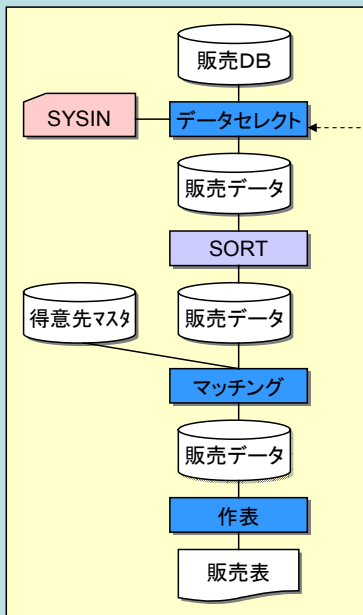
- ・マルチOS対応
- ・GUI対応で習得が簡単
- ・プログラムの流れが視覚化されるためジョブフローが不要
- ・条件分岐等プログラミング言語としての機能が豊富
- ・リラン処理が簡単
- ・ジョブスケジューラ機能と統合

### JCLを全てJP1/AJS2に移行

- ・JCLはジョブの変更があったため手作業で移行
- ・JP1/AJS2/DAを利用してExcelに変換後、チェックや一括変換を実行



## 4-3. COBOL開発ツールの構築(1)



従来はジョブ制御言語でテキストエディタを利用して記述していたが、オープン化に伴いベンダーが分離

COBOL  
エディタ



SORT  
エディタ

ジョブ  
制御言語  
エディタ



Windows  
テキスト  
エディタ

OracleやOLTPの導入に伴いCOBOLコンパイルが複雑化

バッチ(DBアクセス有、無)、オンライン、Web、サブルーチン...

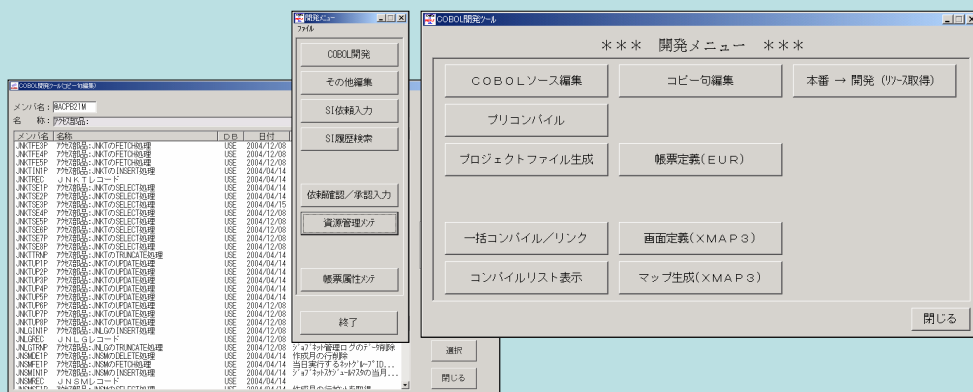
## 4-4. COBOL 開発ツールの構築(2)

3つの機能を統合したCOBOL開発ツールを構築

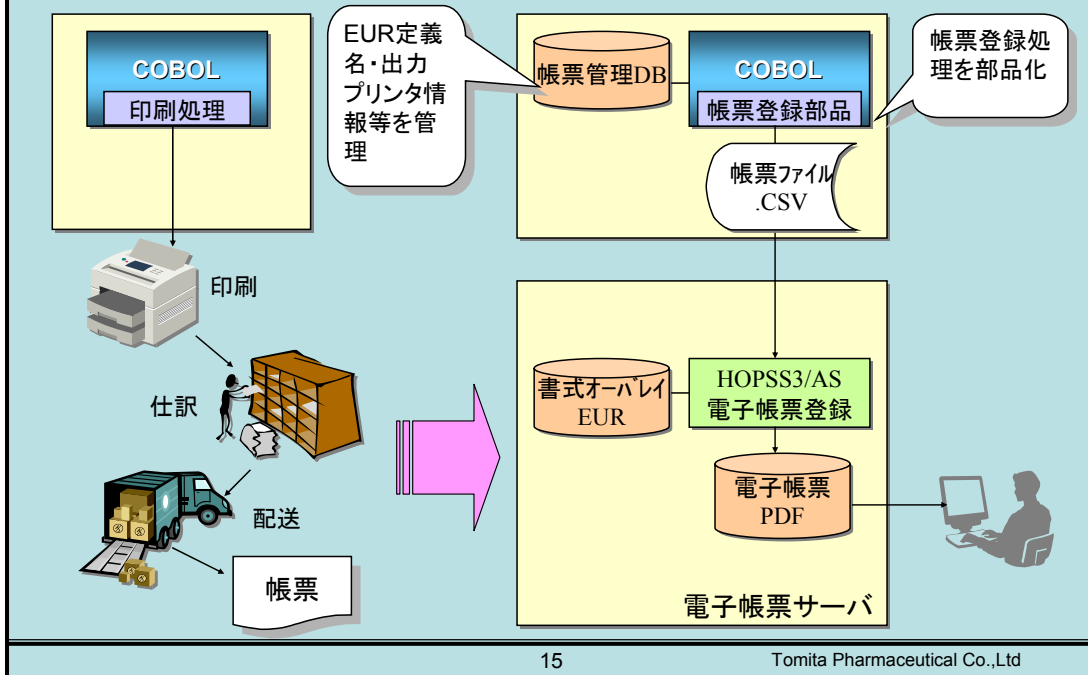
エディタランチャ

COBOLコンパイラ

SI(デプロイ)ツール



## 4-5. 電子帳票システムの導入(1)



15

Tomita Pharmaceutical Co.,Ltd

## 4-6. 電子帳票システムの導入(2)

### ■ 帳票出力処理システムの構築

- ・帳票管理マスタを設け、帳票名とEUR(書式オーバーレイ)、出力プリンタ等の情報を一元管理し、プリンタ出力及び電子帳票登録をコントロール
- ・帳票管理システムとのインターフェース部分を部品化してCOBOLに組み込む

### ■ 電子帳票化のメリット

- ・紙代・トナー代の削減
- ・配送費の削減
- ・仕訳作業の削減
- ・保管場所・保管費用の削減
- ・検索性の向上
- ・PCへのダウンロードが可能

リアルタイム性を除けば、

オンライン処理のメリットと同等



オンラインで再作成すべきソフトウェア資産が、  
バッチ処理のまま利用可能になった。

16

Tomita Pharmaceutical Co.,Ltd



## 5. メインフレーム→Windows(オンライン)

### 5-1.移行概要

### 5-2.レスポンス向上対策

## 5-1.移行概要

### ■ 移行の背景

- ・レスポンス重視(午前中にトラフィックが集中)
- ・レスポンスを重視するエントリー画面に関してはオンラインに移行
- ・レスポンスよりも機能を重視する検索画面に関してはWebに移行

### ■ 移行概要

|            | 移行前       | 移行後                        |
|------------|-----------|----------------------------|
| OS         | VOS3/AS   | Windows Server 2000 SE × 2 |
| OLTP       | XDM/DCCM3 | OpenTP1                    |
| 言語         | COBOL85   | COBOL2002                  |
| 画面インターフェース | XMAP2     | XMAP3                      |

### ■ 移行結果

- ・レスポンス向上対策を実施  
⇒メインフレームよりも高速なレスポンスを実現

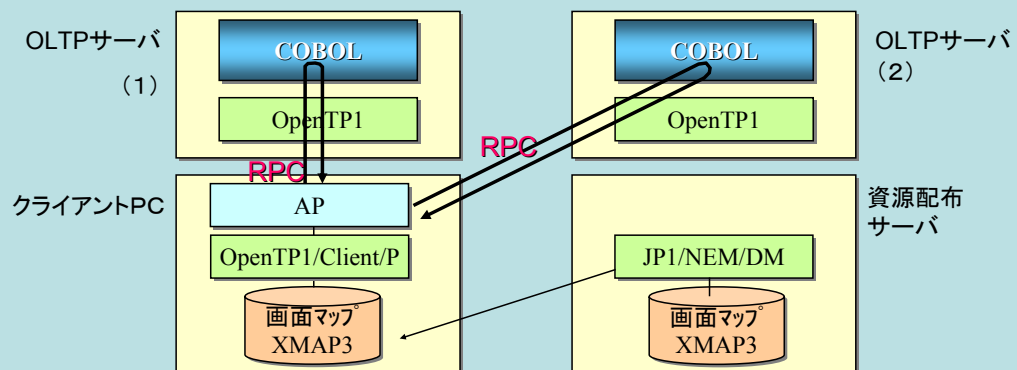
## 5-2.レスポンス向上対策

### ■COBOLの変更

- ・DCCM3 I/F ⇒ OpenTP1 I/F
- ・XMAP2 I/F ⇒ XMAP3 I/F

### ■レスポンス向上対策

- ・MAPのクライアント配布
- ・サーバ2台によるロードバランシング



19

Tomita Pharmaceutical Co.,Ltd

## 6. メインフレーム→Windows(Web)

### 6-1.移行概要

### 6-2.COBIOLによるWeb／OLTP連携

### 6-3.Cosminexus/Webフロントエンドセットの採用

### 6-4.MVC Modelの採用

20

Tomita Pharmaceutical Co.,Ltd

## 6-1. 移行概要

### ■ 移行の背景

- ・業務ロジックの共通化・・・複雑なチェック処理の保守作業省力化  
⇒複雑なチェック処理プログラムをWeb/OLTP連携で移行  
⇒上記以外はWeb/COBOL連携で再構築

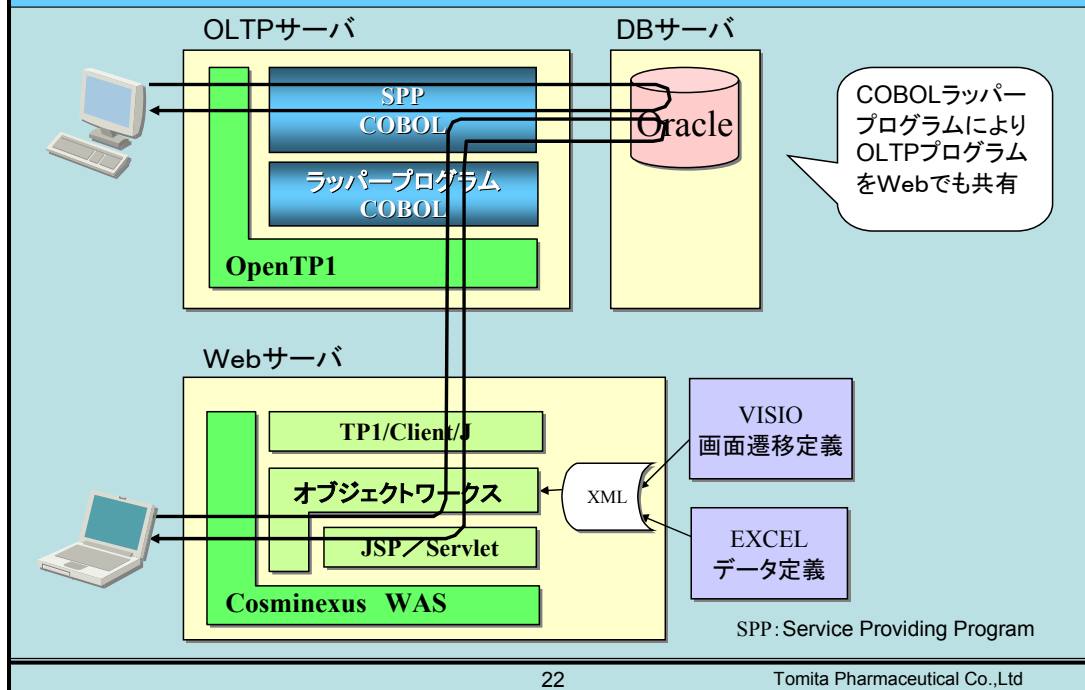
### ■ 移行概要

|      | 移行前       | 移行後                                     |
|------|-----------|---|
| OS   | VOS3/AS   | Windows Server 2003 SE × 2              |
| OLTP | XDM/DCCM3 | J2EE/Cosminexus (Webフロントエンドセット)/OpenTP1 |
| 言語   | COBOL85   | COBOL2002/Java/JavaScript/JSP           |

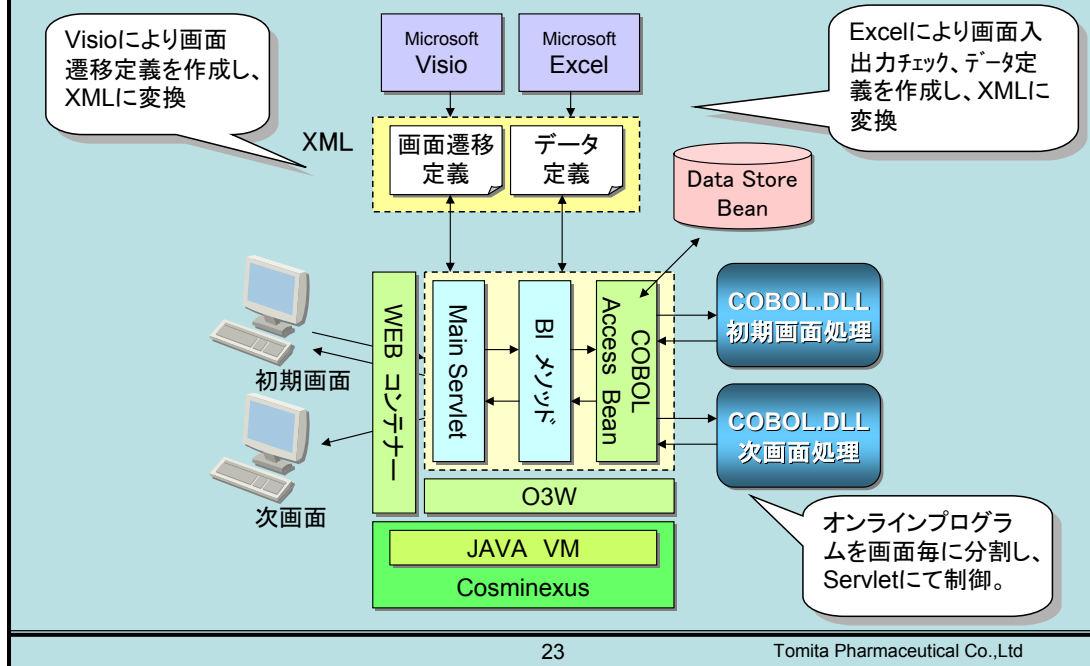
### ■ 移行結果

- ・良好なレスポンスを確保
- ・エンドユーザの反応も上々

## 6-2. COBOLによるWeb/OLTP 連携



## 6-3. Cosminexus/Webフロントエンドセットの採用



## 6-4. MVCモデルの採用

|            | 言語             | エディタ              |
|------------|----------------|-------------------|
| Model      | COBOL          | COBOLエディタ         |
| View       | JSP/JavaScript | DreamWeaver/Excel |
| Controller | Java Servlet   | JBuilder/Visio    |

- ・各言語の専任者が居る場合には開発工数が30～50%短縮  
→従来1本で作成していたプログラムを画面遷移毎に分割。プログラム行数が短くなりロジックの把握及びテストが容易になった。
- ・1人で全てを担当するには開発者のスキルアップが必要
- ・言語及びエディタが分割され、プログラムの見通しが悪くなった



**Model・Controller開発者とView開発者の2名体制を採用**  
**標準化とテンプレート化を徹底**

## 7. COBOLの移行性

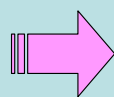
7-1. 移行コストの削減

7-2. インターフェース部分の部品化

7-3. ジョブ制御言語の統一

## 7-1. 移行コストの削減

・コンピュータシステムが進化し続ける限り「移行」は無くならない。



**移行を前提としたシステム構築が必要**

### ■ COBOL移行コストの削減

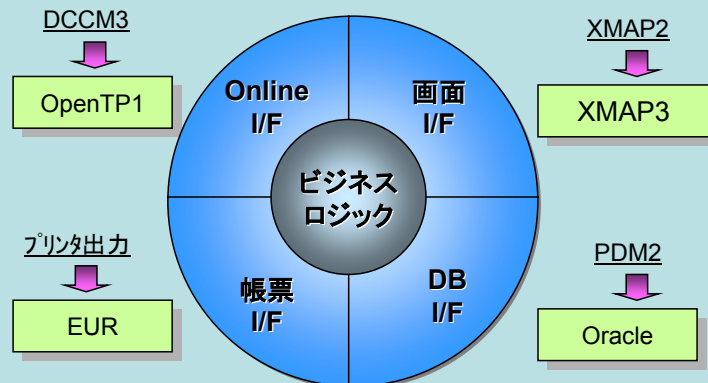
- ・16進等特殊処理を避け、スタンダードなプログラミングを行う
- ・オンライン処理よりもバッチ処理の方が移行コストが安い
- ・SQLに頼らない
- ・標準化・部品化の徹底

## 7-2. インターフェース部分の部品化

・COBOL移行では他システムとのインターフェース部分の修正がメインで、ビジネスロジック部分の変更は殆ど無い。



インターフェース部分を部品化すれば移行性が高まる



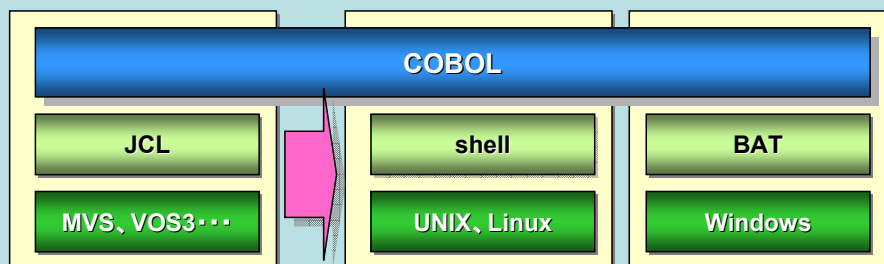
27

Tomita Pharmaceutical Co.,Ltd

## 7-3. ジョブ制御言語の統一

**COBOLは標準化されているが、ジョブ制御言語の標準化は未**

- ⇒ジョブ制御言語の選択が重要
- ⇒ジョブ制御言語の高機能化に伴い要員のスキルアップも重要
- ⇒shell、BATともにリラン処理が面倒
- ・本来はCOBOL Control Languageの制定・標準化が必要か？



28

Tomita Pharmaceutical Co.,Ltd