

# COBOLの世界を拡げる.NET

～基盤システム構築現場より～

日本ユニシス株式会社

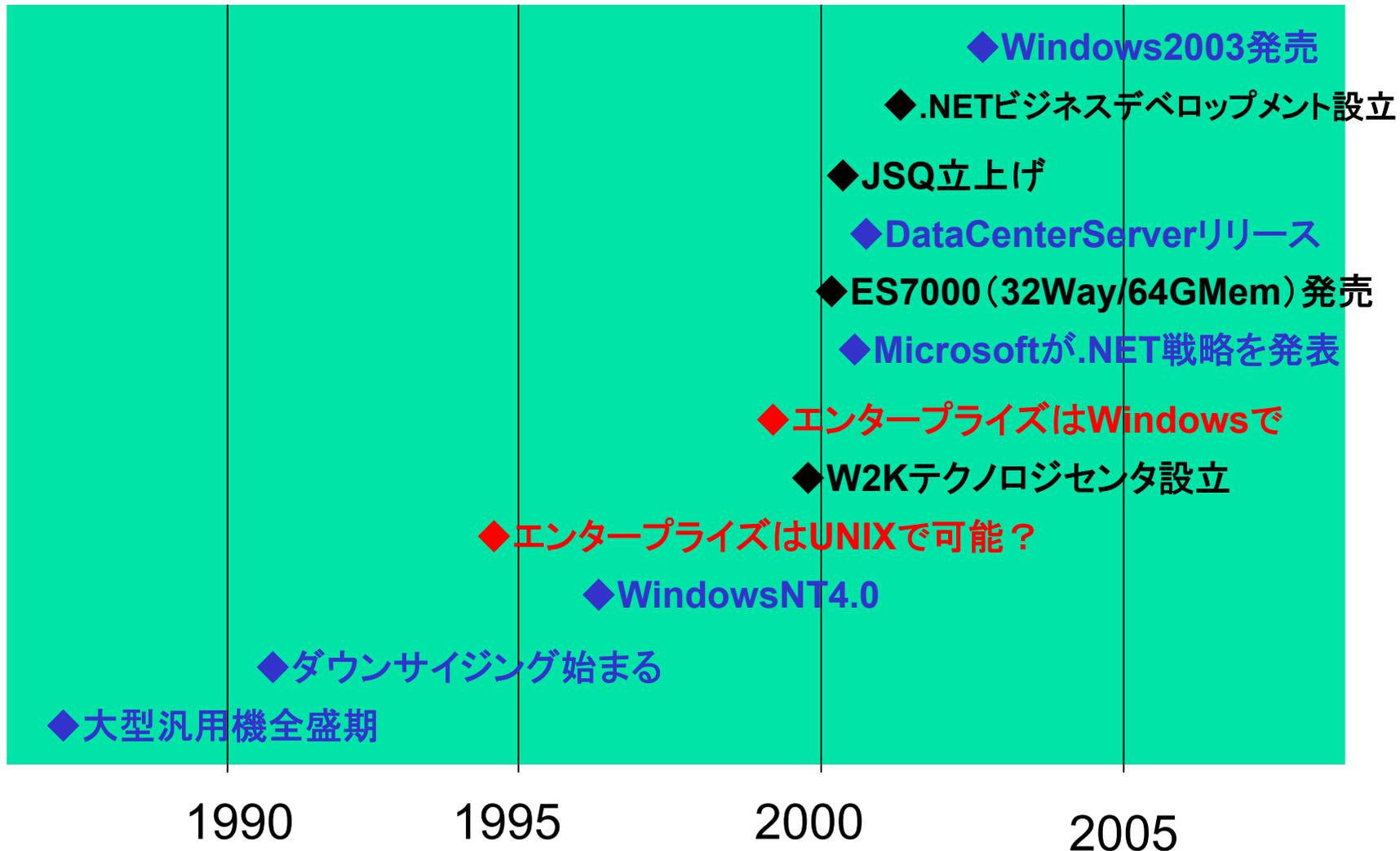
細川 巧

**UNISYS**

# Agenda

- 弊社と.NET戦略
  - .NETBD、Win2003、ES7000とMicrosoft.NET戦略と歩調を合わせて
- エンタープライズな需要
  - エンタープライズで利用するには信頼のおけるWindowsでなければ
  - サーバ:ES7000紹介
  - プログラム:.NET
- .NETは
  - マネージドコード
  - 言語はなんでも可→COBOLは大きい選択肢
  - NetCOBOL for .NET
- NetCOBOLによる開発のポイント
  - NetCOBOLは完全に.NET Frameworkを継承していること
  - COBOLである意義
  - COBOLチックな作り方、標準化
  - プレゼン層をCOBOLで作ることは悲劇？感激？
  - すべてがマネージドコードである意義
- ユニセックス層作りには、.NETでのCOBOL

# 日本ユニシスと、NET戦略



# エンタープライズサーバに求められる要件

ピーク時の処理能力に対応する柔軟な構成	処理性能
業務拡大時の処理能力に対応する柔軟な拡張性	拡張性
24時間365日のシステム稼動に耐えうる信頼性・可用性	信頼性・可用性
集中管理によるシステム管理の簡素化・自動化	一元管理
迅速且つ確実に障害対応できるサポート体制	サポート
将来にわたり進化／継続するプラットフォーム	将来性
セキュリティホールへの即時対応・修正コードとソース開示	安全性

信頼のおける**Windowsプラットフォーム**

Unisys  
Enterprise Server  
ES7000/200

# ES7000のアプローチ

- Windows® Data Center
  - エンタープライズ・システム、データセンター

Windows Data Center

UNISYS

データ  
センター

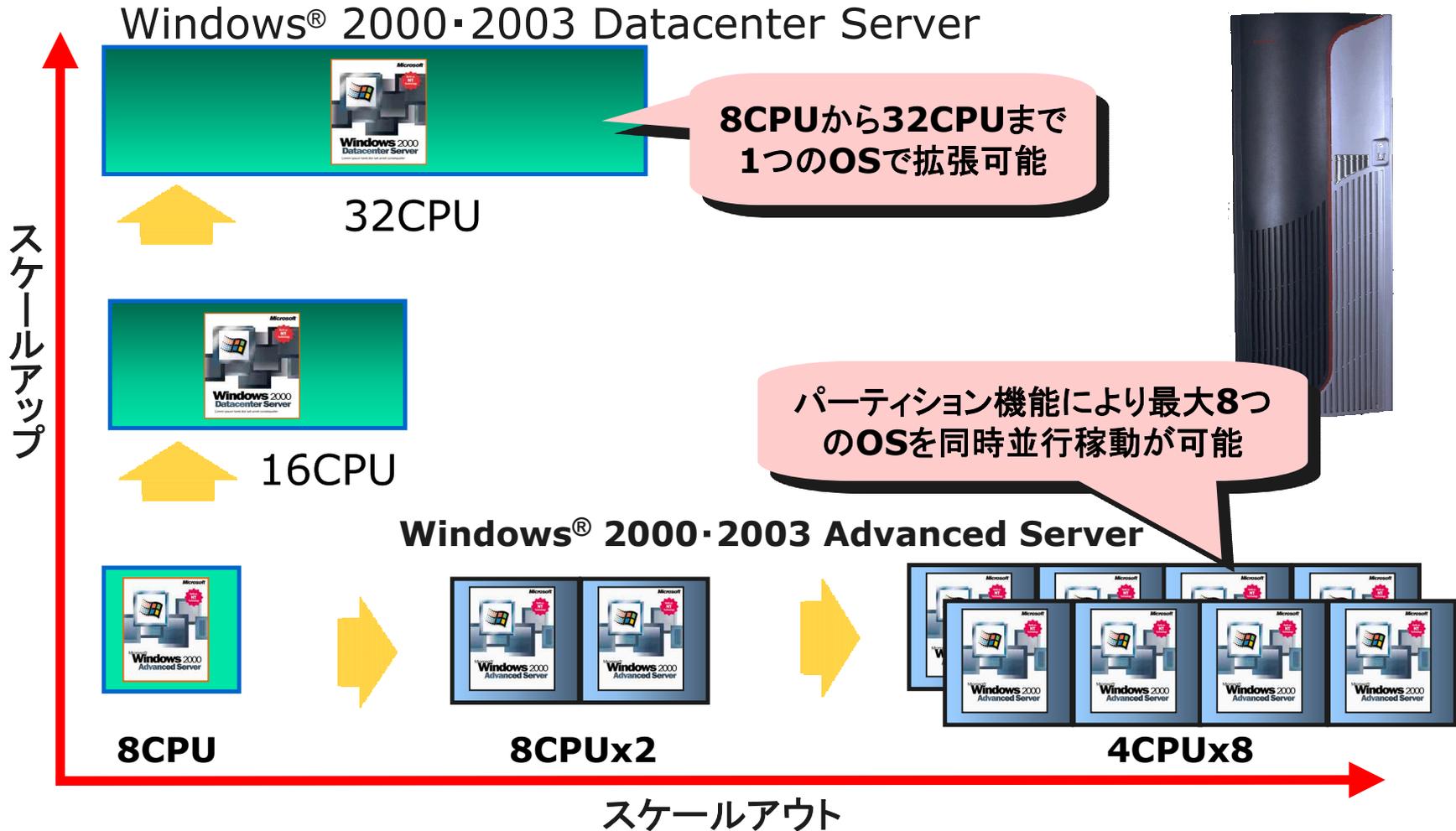
エンタープライズサーバー

部門サーバー

デスクトップ

Microsoft

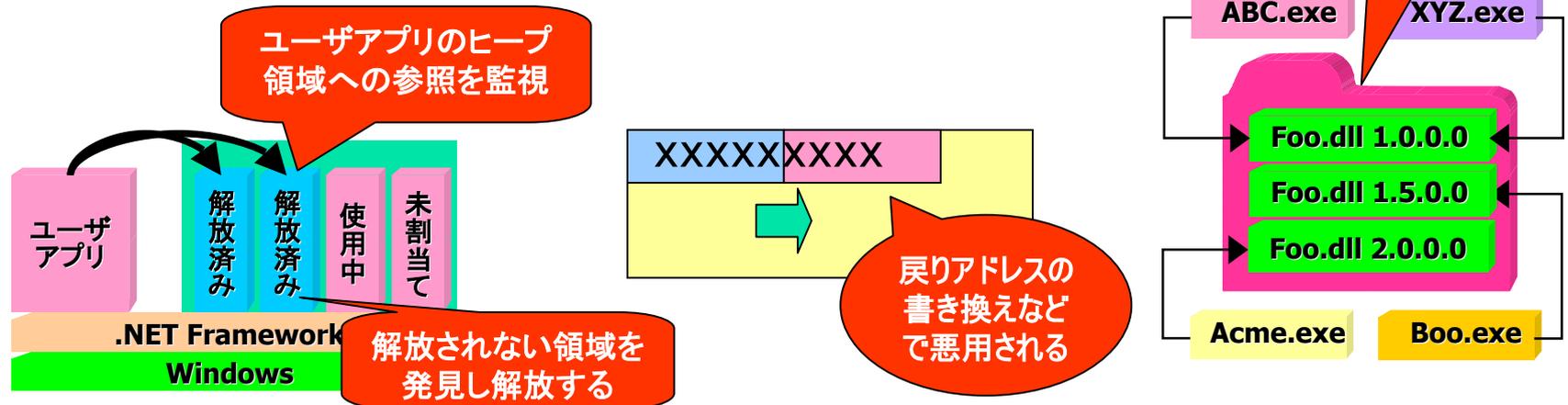
# ES7000ファミリのカバー範囲



# エンタープライズでは.NET

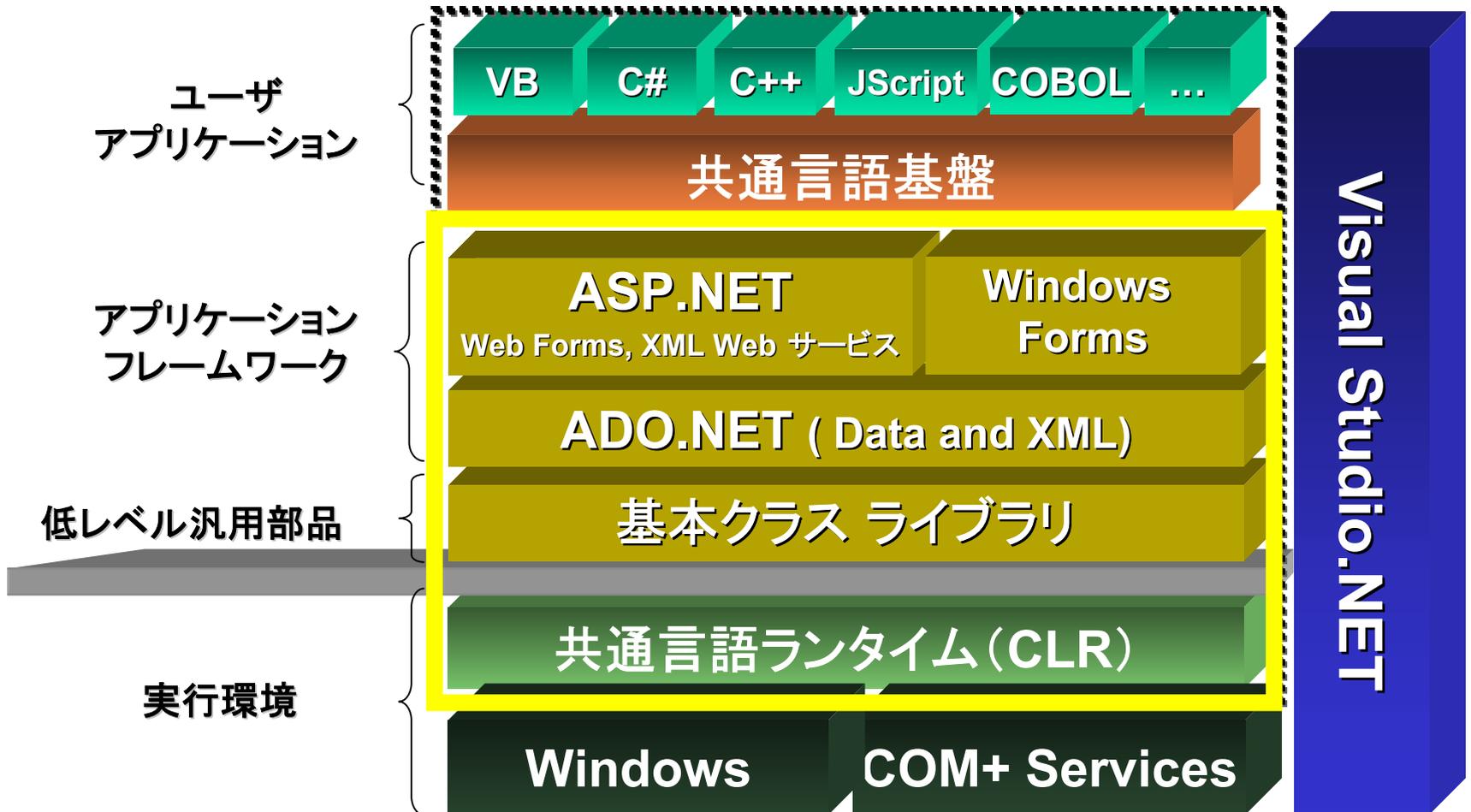
1. Windowsにおける信頼性は飛躍的に向上
2. デバイスドライバを作り易く、品質＝信頼性の向上
3. エンタープライズの信頼性は、  
業務アプリケーションレベルに求められる

- マネージドコードがもたらす安定性・堅牢性
  - メモリリーク問題からの解放(GC)
  - セキュリティ確保
- DLL Hellの解消

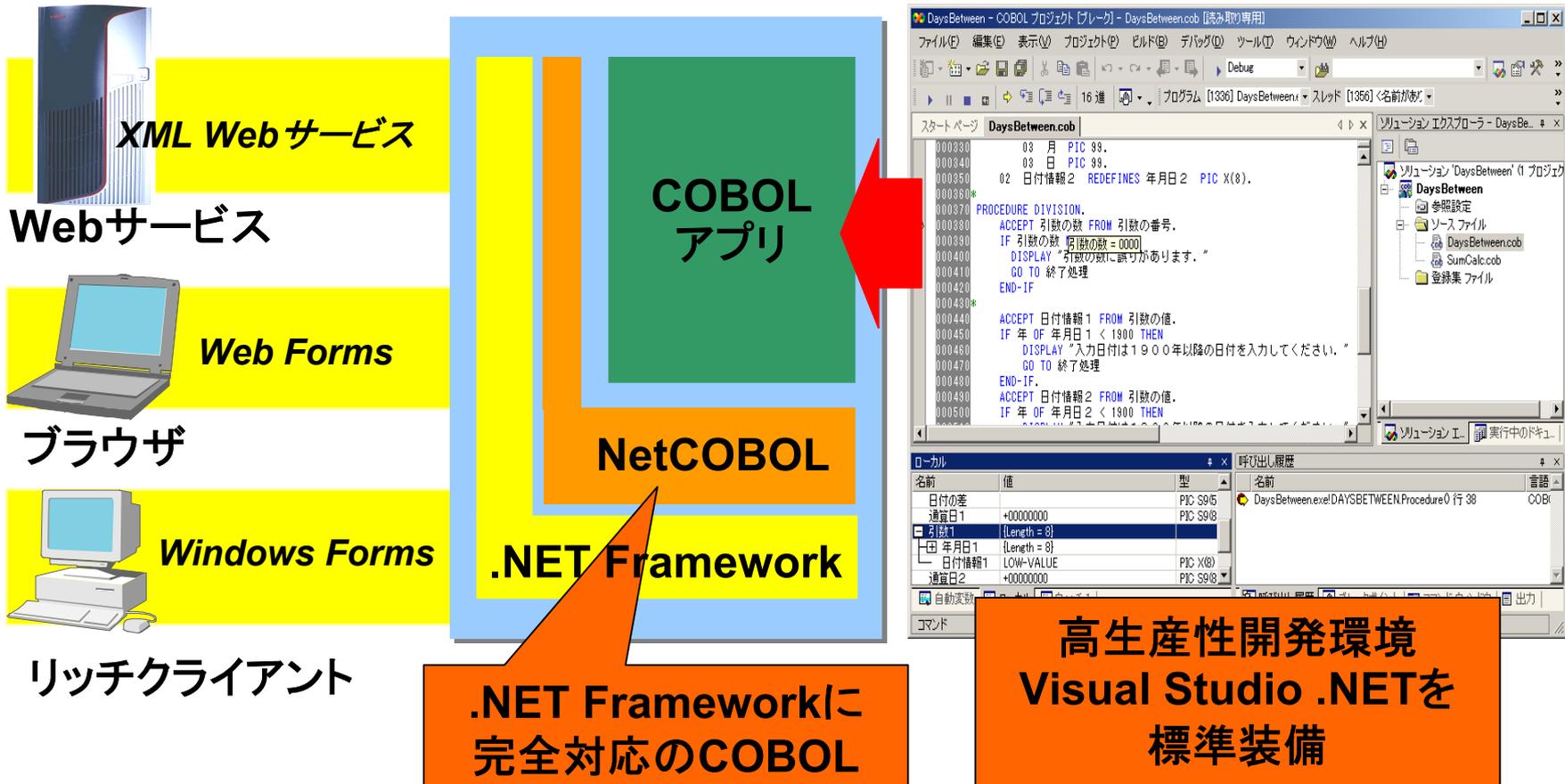


# **.NET Frameworkは**

# .NET Framework の構成

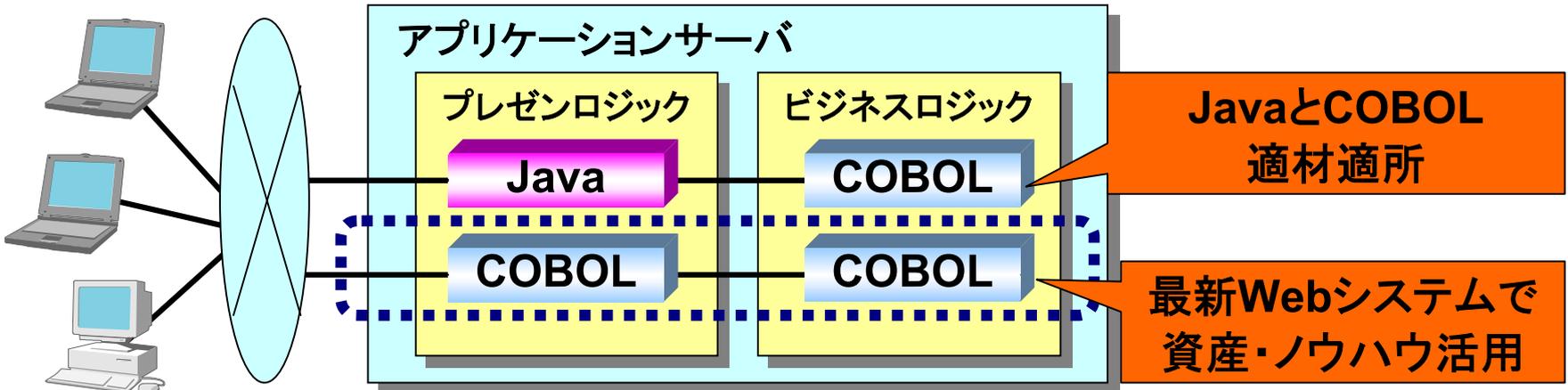


# 富士通 NetCOBOL for .NETでは



- 先端システムのビジネスロジックをCOBOLで記述
- NetCOBOL + .NET で開発効率・保守性向上

# Webアプリケーション開発では



最新インターネット技術

Webアプリの  
開発生産性

フレームワーク

開発環境

既存資産活用

開発ノウハウ活用

Java

COBOL



COBOL+.NET



## 【A社 事例】

- A社 担保管理システム開発をH15年1月着手。

融資業務における担保資産の登記簿情報を含めた管理は、あまり行なわれていない。  
担保・資産を管理し経営支援の視点からリスク評価・管理に利用

今後の開発案件は、.NETにて開発することを視野に入れた。

- システム構築時の留意事項

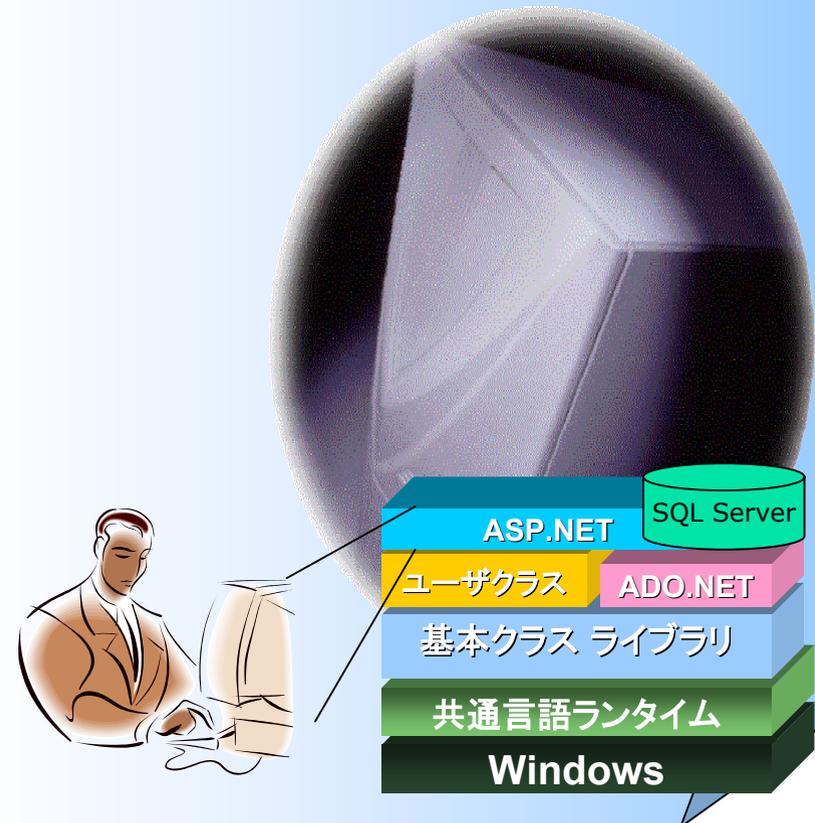
- メインフレームなみの可用性  
ハードウェア可用性のみならず、  
OS含めた業務アプリケーションの可用性が必要  
→ .NETの採用
- VB、Web開発要員不足。  
Oracle+Notes開発経験はあり。  
→ COBOL要員のシフト化で充当  
→ 一過性の開発はベンダー利用しても  
今後の業務保守を考えると内製必須

- 7月より本番開始

- 客先オペレータ研修中

- 構築を通して

- COBOL要員が.NET開発要員に成長。  
VB.NETも多少できるまでに



## 【B社 事例】

### ■ B社の販売管理システム開発をH14年12月着手

見積・受注・発注・在庫・物流・請求・入金におよぶ一括集中型システムを、NETベースで開発

### ■ システム構築時の留意事項

- 多数の取引先とデータ交換を実施。効率重視の為、オンラインプログラムを採用していた。  
同様の構造であればメモリマネジメントが重要に  
→ .NETの採用
- 基盤システムの巨大プログラムは巨大リソースを要求  
→ ES7000 16Way、16Gメモリを採用

### ■ 構築を通して

- 実行効率はES7000採用にてカバー
- プレゼン層とビジネスロジックを明確に分離でよりCOBOL Likeな仕上がり



# 開発のポイント

- **大規模ミッションクリティカルにはFramework作成が重要**
  - 命名規約、プレゼンテーション層の規約(Validation機能の利用法)
  - セッション情報の保持方法
  - エラーハンドリング、メッセージハンドリング
  - トランザクション処理、データアクセス方法、同時実行制御
  - セキュリティ対応(認証、権限など)
- **ホストCOBOL to VB/C#は費用面で厳しい**
- **言語はなんでも構わない = 得意な言語で！**  
**MSILでは、C#もCOBOLも同じコードになる**
- **開発の仕方:コードビハインドでWeb生産性は飛躍的に向上**
- **マネージドコードで開発者負担は軽減・安心**
- **プレゼンテーション層とビジネスロジックは、出来る限り分ける**
- **プレゼンテーション層をCOBOLにさせるのは悲劇？感激？**

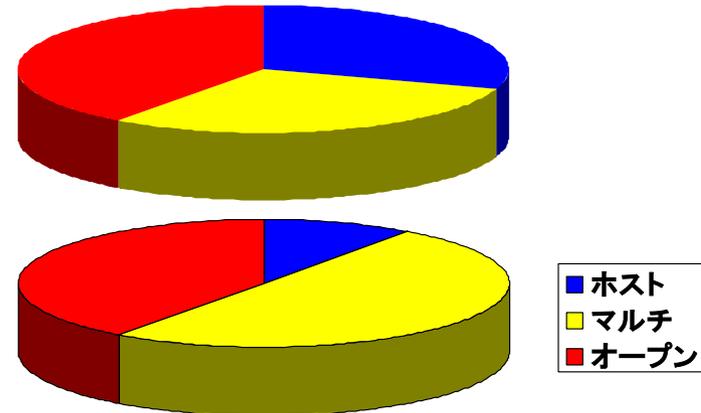


# マルチプラットフォーム化

- ソフトウェア資産の有効活用よりは人材の有効活用！
- 要員のマルチプラットフォーム化（ユニセックス化）
- 要員人口構成は、
  - 40% → 40%:オープン
  - 30% → 50%:マルチプラットフォーム
  - 30% → 10%:ホスト

このマルチプラットフォーム+オープン= 90% は  
仕事量に合わせてフレキシブルに登用

- VB要員、プレゼン層作成要員が不要
- プレゼン層は、画面作成ツールを覚える感覚  
→ 違和感のないシフト
- .NET経験でソフトランディング



## ●まとめ

- 完全な.NET環境を提供するCOBOLは、Webシステム開発において画期的な役割を担う。Frameworkの恩恵を充分受けうる。
- NetCOBOL for .NETの利用は過去資産(プログラム)の継承という位置づけはもとより、ホスト(COBOL)開発者を.NETへ導くきっかけを提供し、ホスト&オープンシステムのユニセックス要員人口を大幅に増大させることができる。
- 大規模ミッションクリティカルなシステムにおいては、Framework対応が不可欠である。

with Unisys.

