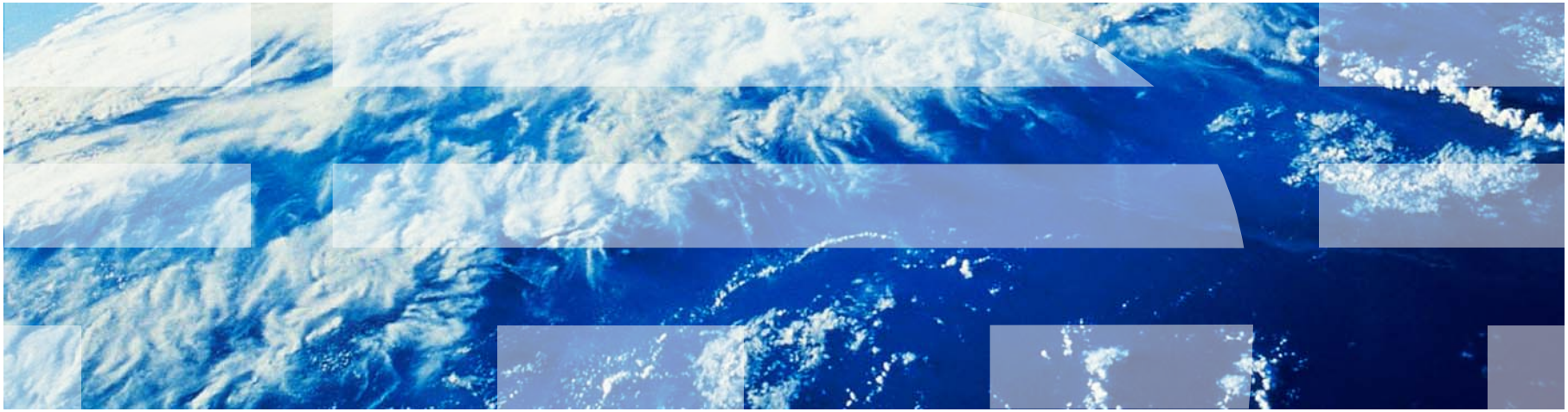
原田 誠 - 日本アイ･ビー･エム株式会社 ソフトウェア開発研究所 ラショナル･エンタープライズツール開発部長

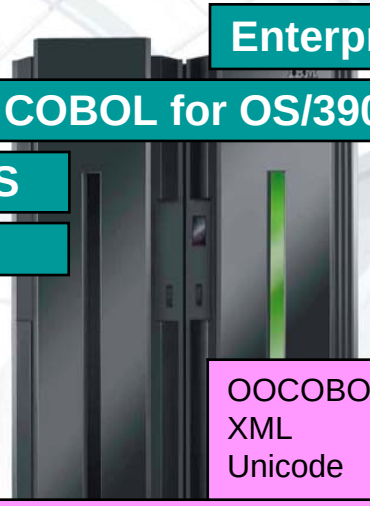
16 April 2010

COBOL開発ライフ･サイクルを刷新するエンタープライズ･モダナイゼーション



IBMメインフレームCOBOLの50年の歴史

1960 1970 1980 1990 2000 2010



Enterprise COBOL

COBOL for OS/390

COBOL for MVS

VS COBOL II

OS/VS COBOL

機能拡張							
							OOCOBOL (Java) XML Unicode
						31桁10進数 DB2 coprocessor OS/390 Unix	
				DBCS CICS	OOCOBOL(SOM) 組み込み関数		組み込み関数
COBOL 標準	85			○	○	○	○
	74		○	○	○	○	
	68		○				
デバッグ機能			固有ツール (TESTCOB)	固有ツール (COBTEST)	Debug Tool (言語環境プログラムを共用)		
実行環境			固有ライブラリー (COBLIB)	固有ライブラリー (COB2LIB)	言語環境プログラム		

IBM COBOLの特長

- **IBMがご提供するCOBOLコンパイラー**
  - IT業界屈指のCOBOLコンパイラー
  - 高い信頼性と安定性
  - プラットフォームの性能を最大限に活用
  - プラットフォーム間で共通の言語仕様
- **プラットフォーム**
  - System z - z/OSおよびz/VSE
  - Power Systems - AIXおよびIBM i
  - Windows - ラショナル製品組み込みのローカルCOBOLコンパイラー
- **お客様のご要望に対応**
  - アプリケーション開発の生産性向上
  - アプリケーションのモダナイゼーションを支援
  - ミドルウェアとの連携、他言語との相互連携、SOAサポート
  - HW性能の最大限の活用、実行スピードの最適化

IBMのコンパイラー開発チーム

- プログラミング言語とコンパイラーの領域で50年以上にわたる技術革新をリード
- 主要開発拠点：カナダのトロント、およびUSのシリコンバレー
- 技術者の数：約350名
  - 世界最大のコンパイラー開発チームの一つ
- 多くの言語標準活動で中心的な役割
- 開発しているコンパイラー：COBOL、PL/I、C/C++、Fortran, およびJava JIT
- 以下のチームと緊密な連携
  - IBM研究所：ワトソン(US)、東京(日本)、ハイファ（イスラエル）
  - 学術機関：UPC, UT Austin, U of Toronto, U of Albertaなど
  - IBMのCPU開発チーム：AustinおよびPoughkeepsie
  - 業界のCPU開発チーム：AMDおよびIntel

IBMのCOBOL戦略

- **COBOLコンパイラー開発への投資を増強**
  - COBOLは最重要なプログラミング言語
  - 実行を支えるミドルウェアとの連携強化
  - System z コンパイラーのパフォーマンスと機能に重点的に取り組む
- **最適化の技術革新はまだ成熟途中**
  - パフォーマンス向上および新アーキテクチャー活用に、技術革新の長いパイプライン
- **最新コンパイラーでTCO削減に寄与**
  - 既存および新しいHWで、アプリケーションのパフォーマンスを引き上げる

IBM COBOLの重点投資領域

- **アプリケーション開発の生産性向上**
  - Eclipseベースの開発環境: Rational Developer for System z (RDz)
  - CICSやDB2用のコプロセッサー
  - 開発ツールとの連携: Debug Tool, File Manager, Fault Analyzer, Asset Analyzer
  - プラットフォーム間で共通の言語仕様
- **相互連携、ミドルウェアとの連携**
  - DB2, CICS, IMS, Java, WebSphere
- **グローバル化**
  - Unicodeサポート
  - 各国特有のコードページやロケールに対応
  - DB2 Unicode データベース、DB2とCOBOLのコードページ連携
- **SOA対応**
  - XMLサポート
  - Webサービスに対応
  - コンパイラーのメタデータをツールが活用
- **プラットフォーム性能の活用とパフォーマンス最適化**
  - z/ArchitectureのHWインストラクションを有効活用
  - DFSMSのアクセス・メソッドを有効活用

COBOLを取り巻く現状

- **COBOLは世界のほとんどの商用基幹アプリケーションを稼動**
  - 世界のビジネス・データの**75%**、金融取引の**90%**を処理
  - **2,000億行**のコードと推定
- **アプリケーションと開発環境の刷新が必要**
  - SOA対応のため、Web 2.0、Java、XMLなどの新しいテクノロジーと統合できるよう、既存アプリケーションを拡張
  - **100万人**のメインフレーム開発者、**90%**は旧来のツール使用
- **System zのHW設計は、パフォーマンス最大化に重点**
  - コンパイラーはプラットフォームの可能性を実現する鍵
  - プログラミング・モデルを単純化し、複雑さの健在化を防ぐ
- **開発現場の課題**
  - 保守費用がかかり過ぎ、新規投資が難しい
  - COBOL開発要員が不足
  - ビジネスの変化にアプリケーションの対応が間に合わない

IBMが提唱する解決案：アプリケーションと開発環境の刷新

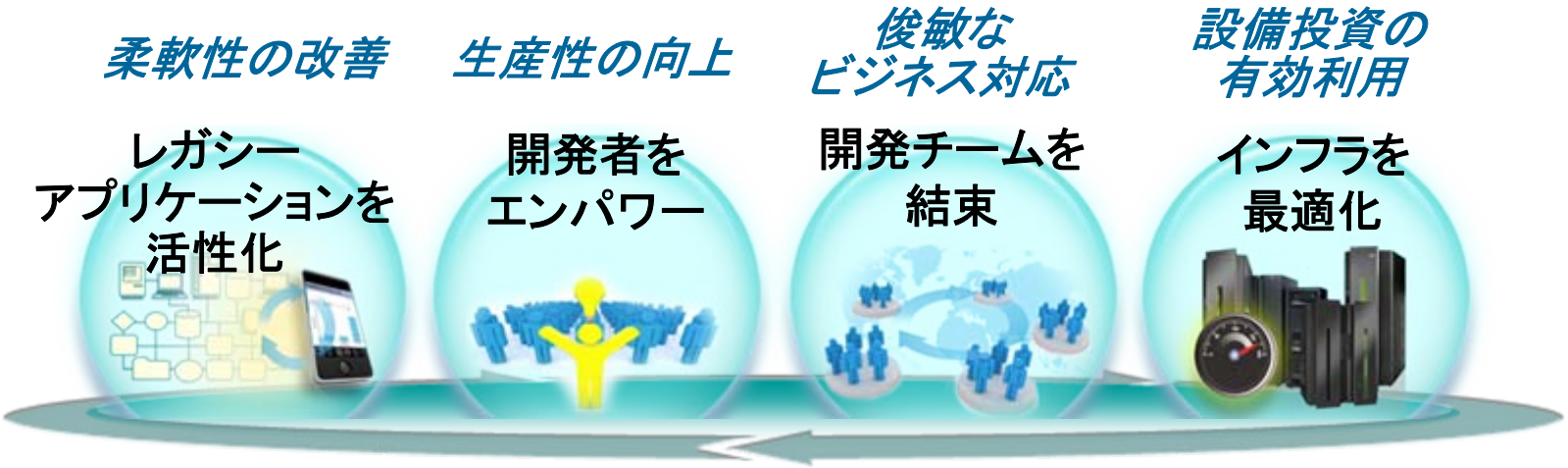
エンタープライズ・モダナイゼーション

このページ上の数字はIBMの推定

エンタープライズ・モダナイゼーションへの期待

投資対効果	期待する効果
短期的	■ 既存アプリケーションの拡張・保守作業に要する費用を下げる ■ ビジネス上の必要性に素早く対応して適合する
中期的	■ 手作業による保守作業を少なくする ■ アプリケーションの複雑度を下げる ■ 低い費用のプログラマーを活用する ■ 養成や維持が難しいスキルへの依存を無くす ■ ビジネス上の要求により良く対応するために、アプリケーションをより柔軟に、より再利用可能にする
長期的	■ 設計やテクノロジーの制約を取り除く ■ アプリケーションをより簡単に、より速く変更できるように改造する

IBM Rationalのエンタープライズ・モダナイゼーション

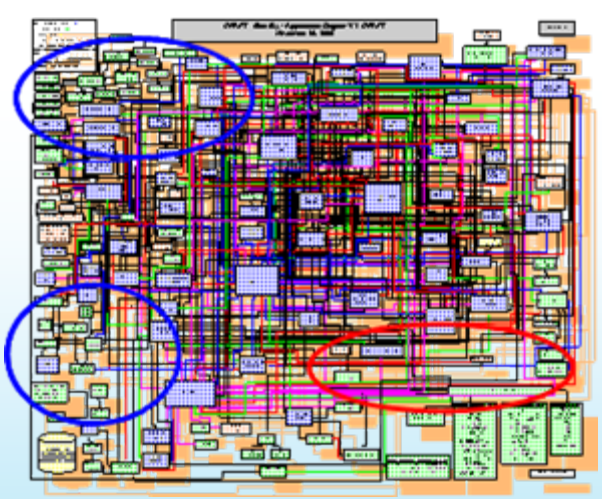


- アプリケーション・ポートフォリオを再活性し、*柔軟性を改善*
- 最新のスキルによりイノベーションを加速し、*生産性を向上*
- 組織のサイロを橋渡しして、*俊敏なビジネス対応*
- アプリケーションのインフラを最適化し、*設備投資を有効利用*

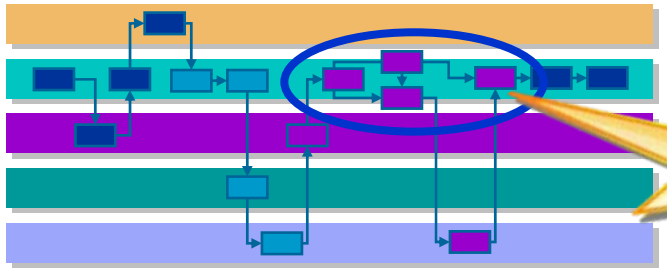
Rational Software Delivery Platform *powered by* Jazz

レガシーアプリケーションを活性化：SOAで柔軟性の改善

強力なツールを使用して、実証済みプログラム資産を活用・再利用することにより、SOAへの進化に素早く着手できます



- EGL
- C/C++
- COBOL
- PL/I
- RPG
- SQL
- CICS, IMS
- Java applications
- Stored procedures



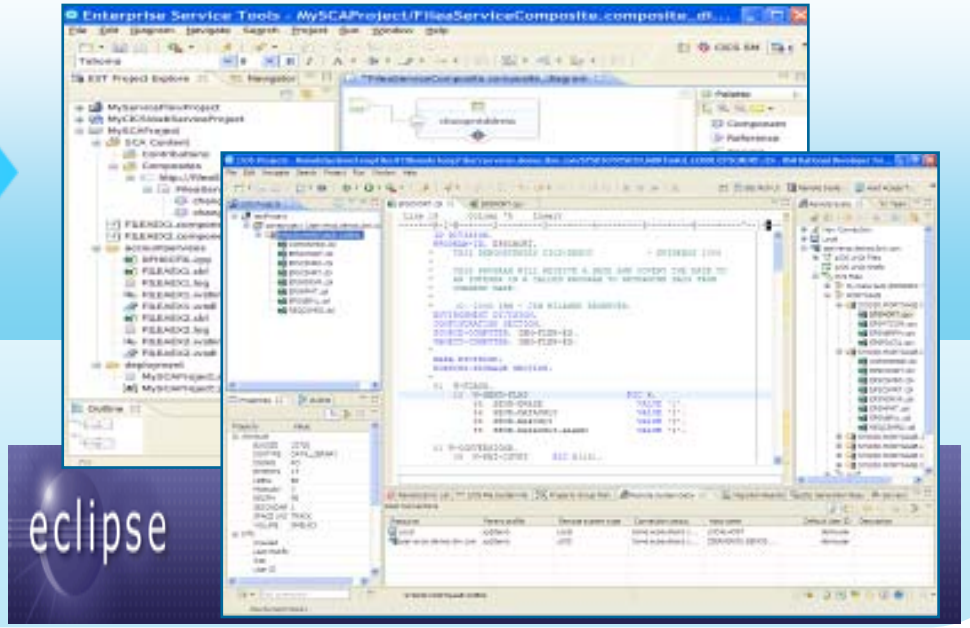
WAS、CICS、IMS、Datapower環境で稼動するサービスを開発！

- ✓ これまでの投資を再利用して ROI を持続的に実現
- ✓ 強健な既存アプリケーションを利用してリスクとコストを低減
- ✓ 開発サイクルを迅速化して運用開始までの時間を短縮

- Rational Developer for System z
- Rational Application Developer
- Rational Business Developer
- Host Access Transformation Services

開発者をエンパワー: Eclipseで生産性の向上

複数のプラットフォームやプログラミング言語にまたがるエンタープライズ・アプリケーションを開発・保守するために、Eclipseベースの最新開発ツールに更新する



- ✓15%以上の開発生産性向上(*)
- ✓新しい人材の確保に役立つツール

(*) IBM System z ユーザー企業を対象に実施した生産性に関する調査結果のまとめ

- Rational Developer for System z
- Rational Developer for POWER
- Rational Application Developer

開発チームを結束: Jazzで俊敏なビジネス対応

プラットフォームや地域の枠を超えたコラボレーション、プロジェクト状況のリアルタイム自動モニター、ビルトインされた開発ワークフロー、これらをサポートする新しいチーム・インフラストラクチャーを展開する

- ✓異なる IT チーム間のコミュニケーションの改善
- ✓障害の早期検出と解決の迅速化
- ✓的確な情報に基づく意思決定
- ✓ソフトウェア開発の自動化促進、透明性強化、および予測可能性向上

▪ Rational Team Concert for System z

インフラを最適化: 最新コンパイラーで設備投資の有効利用

COBOL、PL/I、C/C++、Java、および Fortran コンパイラーに新しい先進の最適化テクノロジーを活用します

- ✓ 既存ハードウェアのキャパシティーを高めて設備投資を抑制
- ✓ アプリケーション・スイートのパフォーマンスを改善
- ✓ 必要な CPU サイクルを削減して ROI を最大化

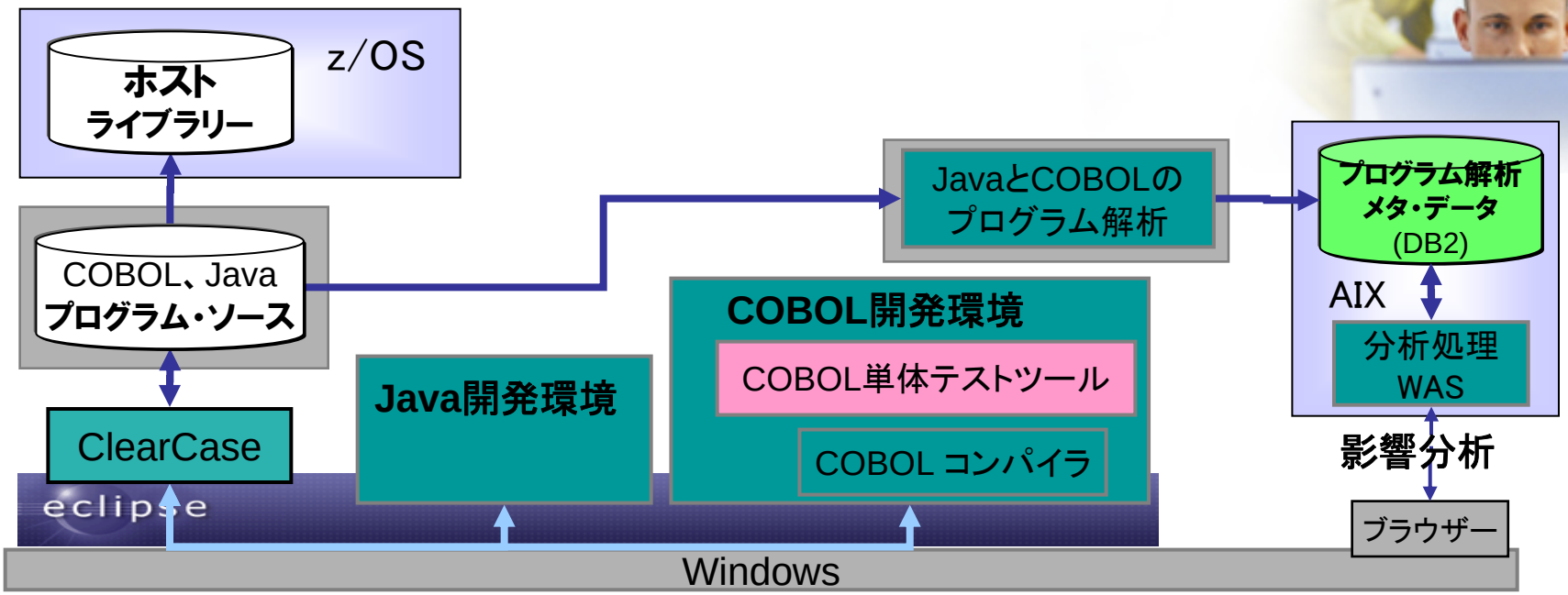
- Enterprise COBOL
- Enterprise PL/I
- z/OS XL C/C++

ユーザー事例：保険基幹アプリケーションの開発環境

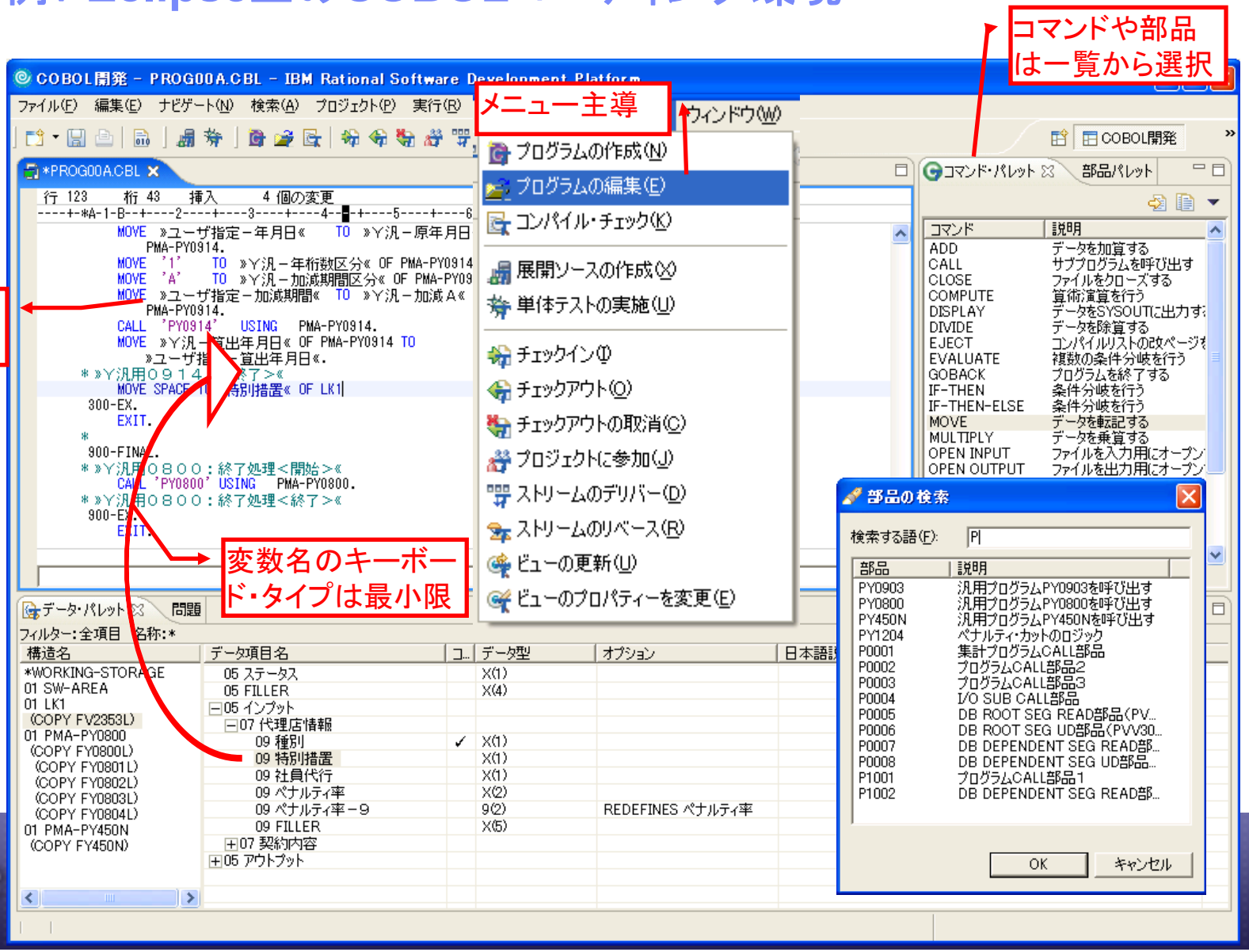
COBOL基幹アプリケーションを20-30年ぶりに順次再構築

生産性向上と品質担保の観点から、開発環境も再構築

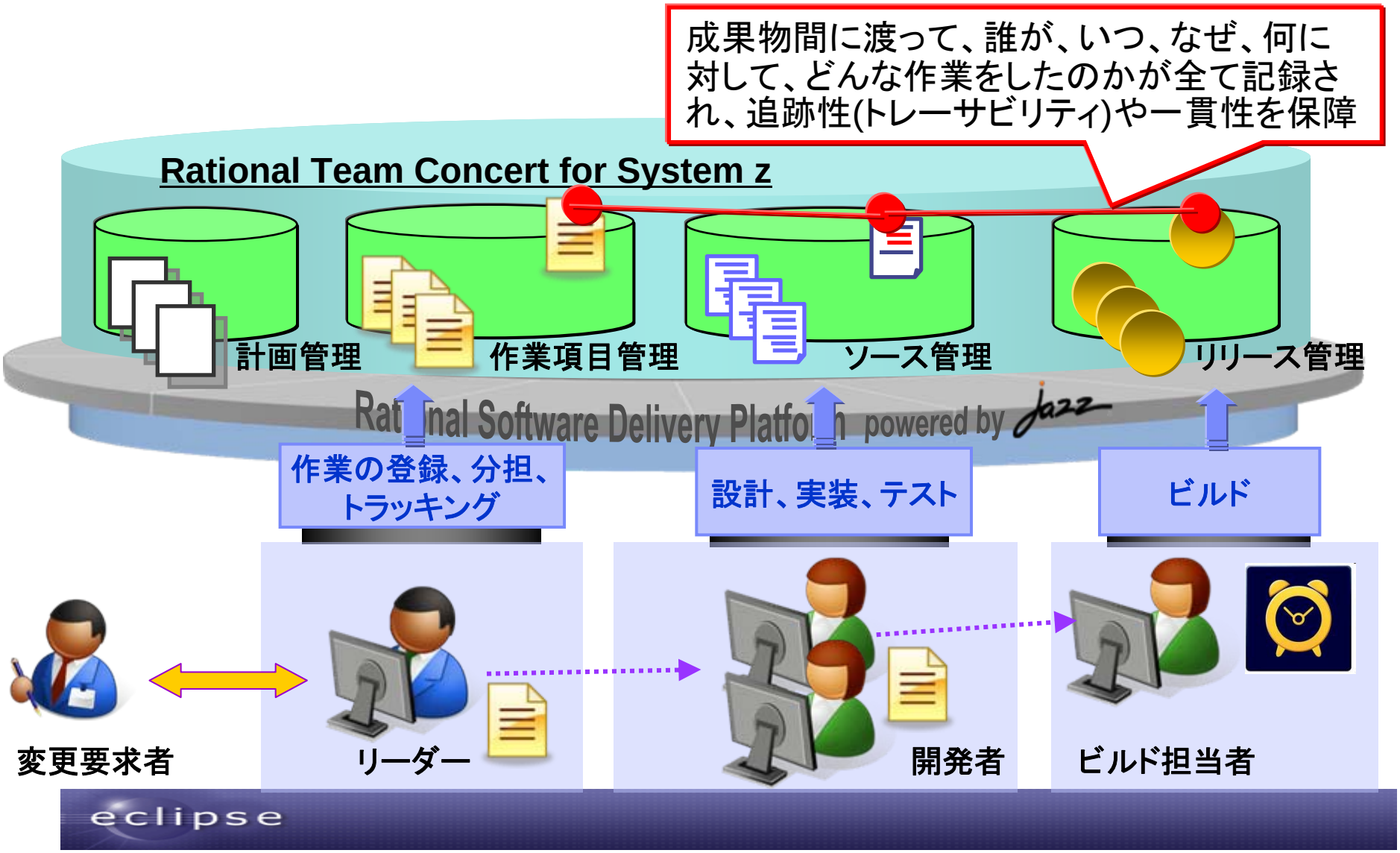
- ホストだけを用いた従来の開発手法には生産性に問題
- 単体テストで全パス・カバレッジが確認でき、納入時にそれを検証できる仕組みが必要
  - ✓ 単体テストまでをEclipseベースのRationalツールを用いてWindows上で実施
- COBOLとJavaを一緒に管理
  - ✓ ソース・コードはCOBOLもJavaもClearCaseでバージョン管理し、並行開発も可能
  - ✓ COBOLとJavaの影響分析にRationalツールを活用



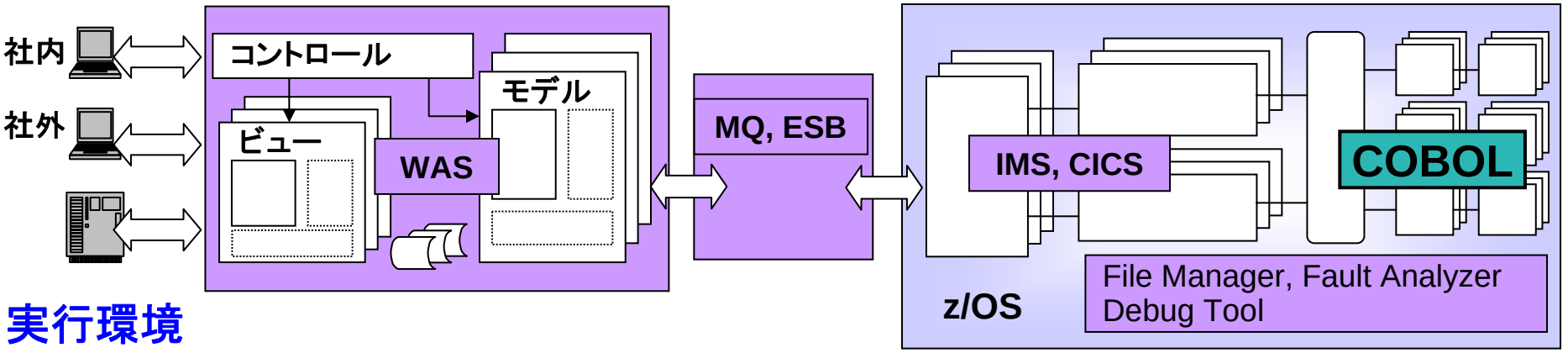
ユーザー事例：Eclipse上のCOBOLコーディング環境



Jazz環境の活用：生産性の向上とトレーサビリティの保障

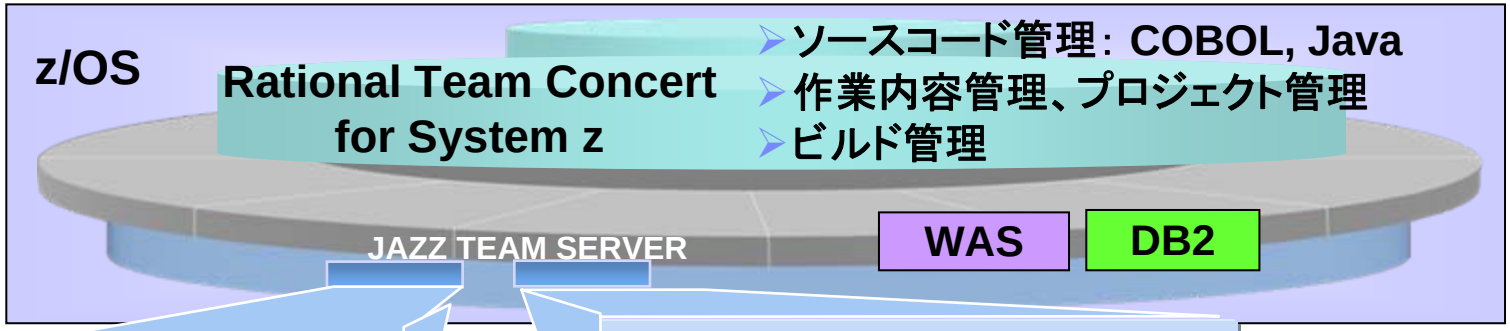


COBOL開発ライフ・サイクルを刷新するエンタープライズ・モダナイゼーション



実行環境

開発環境



Eclipse環境とJazz環境の統合