



IBMメインフレームアプリケーションの開発支援ツール

Rocket® Enterprise Developer

(formerly a Micro Focus® product)



TM

Enterprise Developer は、IBM メインフレームで稼働しているバッチ / オンラインアプリケーションをオープン環境で有効に活用できる製品です。業界標準の Eclipse、Visual Studio、軽量で速い Visual Studio Code 上で開発や単体テストを行うことにより、COBOL と PL/I アプリケーションの生産性と品質の向上を支援します。Visual COBOL の全機能を含む上位製品となるため、将来のモダナイゼーションにも備えることができます。

概要

IBMメインフレームアプリケーション開発のモダナイゼーション

IBM メインフレームは現在でも日々世界のビジネスランザクションの重要な部分を担って稼働し続けています。メインフレーム上の基幹システムは、多くの企業にとってその業務の心臓部分を担うものであり、欠くことのできない経営資源となっています。

しかしながら、メインフレームアプリケーションの開発は、モダンなソフトウェア開発の潮流とは異なる独自の道を歩んできており、開発スキルはJava や.NET のコミュニティのそれと乖離したものになります。Enterprise Developer はIBM メインフレームアプリケーションの開発のための最も先進的な環境を提供し、この問題を解決します。

Enterprise Developer はEclipse、Visual Studio、Visual Studio Code の利用が可能で、COBOL を熟知した開発者にもCOBOL が初めての開発者にも高い生産性をもたらします。また、Enterprise Developer に搭載されている「JVM COBOL」によりCOBOL アプ

リケーションをJVM 上で、「.NET COBOL」によりCOBOL アプリケーションを.NET 上で稼働させることができます。

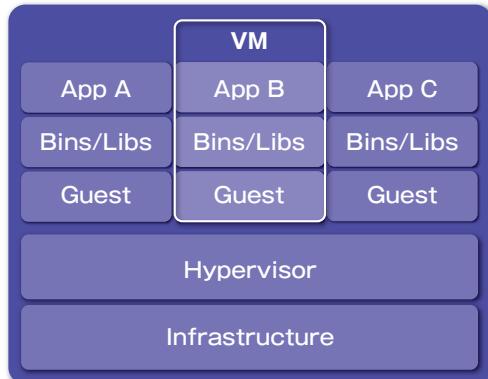
加えて、コンテナ型の仮想環境にも対応しています。

物理サーバー上で複数のOS を稼働させる仮想化はサーバー資源の効率化を実現しますが、現在主流のハイパーバイザー型の仮想化は仮想環境（仮想マシン）毎にゲストOS が必要です。

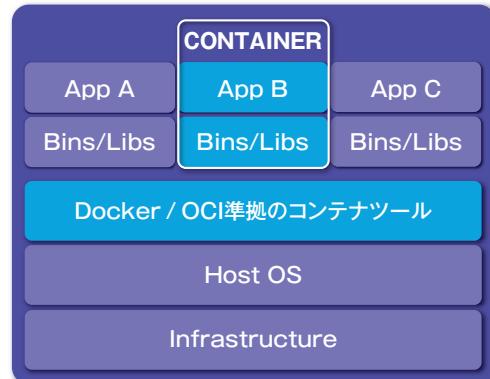
一方、コンテナ型の仮想化は、Linux カーネルの「コンテナ」機能を使って実行環境を他のプロセスから隔離し、その中でアプリケーションを動作させ、ホストOS を共有します。コンテナのオーバーヘッドはプロセスとほぼ同一で、仮想マシンのオーバーヘッドと比較すると非常に軽量です。また、コンテナはアプリケーションとライブラリを同一のコンテナ内に固めるため、アプリケーションの移動やデプロイが容易です。

開発 / テスト工程ではトラブル発生時の開発環境の再構築や大規模開発における開発環境の展開を実現し、本番運用では検証済み実行環境から本番環境を構築可能です。

アプリケーションレベルの仮想化イメージ



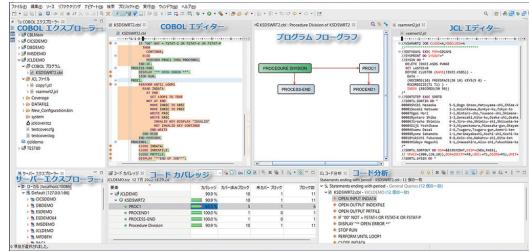
コンテナレベルの仮想化イメージ



製品の特長

1. 業界標準 IDE によるスキル活用

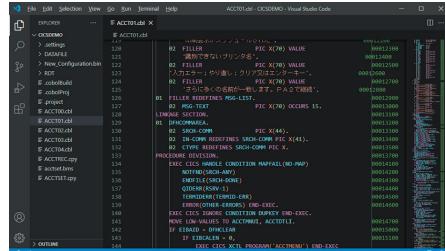
Enterprise Developer の提供する開発環境は、Eclipse、Visual Studio、Visual Studio Code がベースとなっています。このため Java や .NET などのオープン環境の開発者が COBOL、PL/I の開発に参入する際のハードルを低くすることができます。また、同じ IDE を使用することにより、幅広い開発言語のスキルを持つ開発者の育成に貢献します。



Eclipse IDE



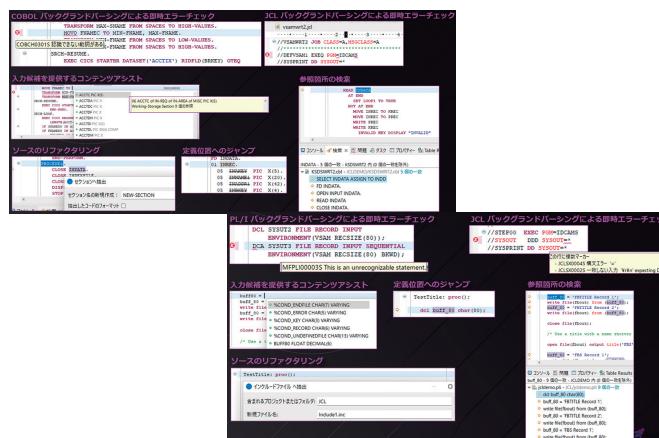
Visual Studio IDE



Visual Studio Code

2. プログラマの生産性向上

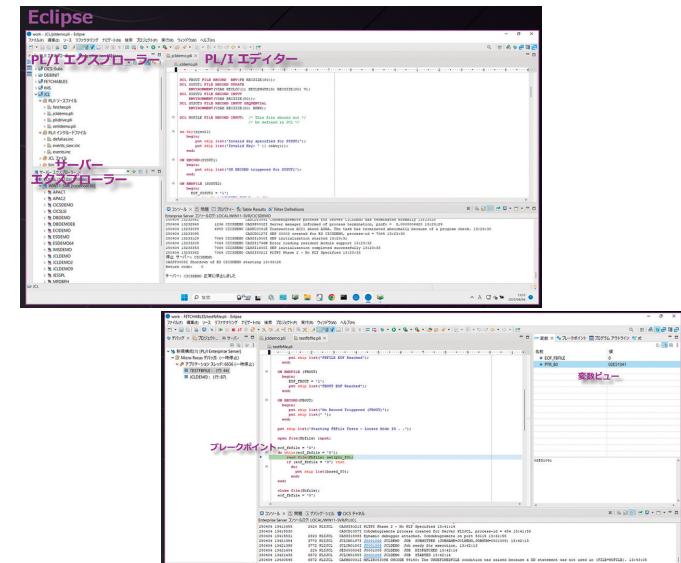
Enterprise Developer の IDE は、これまで Java や .NET の開発者が享受してきた高度なプログラミング機能を COBOL、PL/I 開発者に開放します。バックグラウンドパーサリングやコード入力アシストなど、オープン環境の開発者にとっては当たり前の機能を COBOL、PL/I 開発者も同様に利用でき、生産性の向上を支援します。



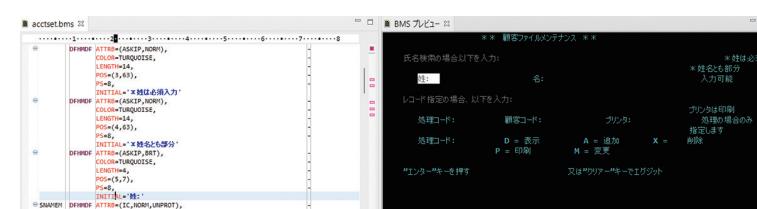
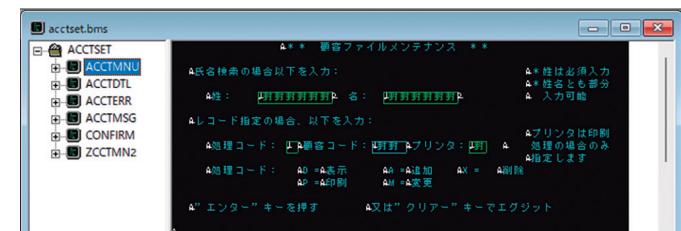
Windows マシンの Eclipse では、Linux マシンにあるソース類の編集、コンパイル、デバッグを可能にするリモートプロジェクトが提供されており、リモートマシンを使用した開発作業を支援します。



PL/I ソースの編集やデバッグも COBOL と同様に行うことができます。



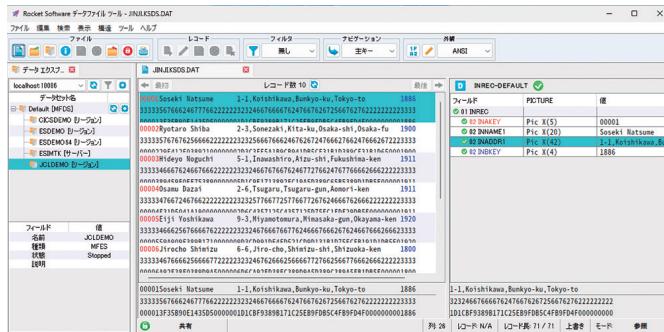
CICS の画面定義を対話型でメンテナンスできるツールや Eclipse のプレビュー機能により、オンラインアプリケーションの開発業務を支援します。



3. 単体テストの効率化

Enterprise Developer で開発したアプリケーションは、内蔵された IBM メインフレームの開発用擬似環境を使用してテストを行うことができ、JCL のサブミットや 3270 エミュレーターを使用した CICS、IMS トランザクションを実行することができます。各 IDE 内ではデバッガを使用したステップ実行や値の確認などのデバッグ機能を利用することにより、単体テストの効率化と品質の担保を支援します。

テストで使用する SAM/VSAM データファイルを対話型で保守する GUI ユーティリティもデフォルトで装備しています。このデータファイルツールは JES 機能のカタログファイルと接続してデータをメンテナンスするこどもでき、EBCDIC 文字コードデータにも対応しています。



4. IBM メインフレームの省資源化

クロス開発の場合、Enterprise Developer 内で構文チェックされ、単体テストによってロジックの正当性が保障されたアプリケーションは、IBM メインフレーム上で再コンパイルされ実稼働に入ります。Enterprise Developer を使用して十分に品質チェックされたアプリケーションは、高い確率で IBM メインフレーム上でも正しく稼働するため、コンパイル・テスト・デバッグで消費する IBM メインフレーム資源の節約に貢献します。

5. Enterprise Server, Enterprise Test Server へ移行するための開発環境

Enterprise Developer は単独の製品としてもアプリケーションの開発生産性向上に活用いただけますが、このほかに統合テスト環境製品である Enterprise Test Server や、実稼働製品である Enterprise Server へ移行するための開発環境としても利用できます。

主な機能

1. 統合化されたワークベンチ

COBOL、PL/I 開発環境である Eclipse または Visual Studio へ、既に導入済みの構成管理、要件管理、テスト管理などのフレームワークをプラグイン経由で統合することができます。構成管理では Git、Microsoft Team Foundation Server、Subversion などのプラグインを自由に選択し、ソースコードの世代・変更管理を行うことができます。

これによってアプリケーション開発のライフサイクル全体を支援するシステムを構築することができます。

2. IBM メインフレーム互換性コンパイラ・ランタイム

Enterprise Developer に内蔵されている COBOL コンパイラは、IBM メインフレームのクロス開発で半世紀にわたり活用されてきたノウハウと実績から、IBM COBOL コンパイラの各バージョンに対して高い互換性を持っています。このバージョンを「言語の方言」としてコンパイルすると、その挙動を模倣する多数のコンパイラ指令が暗黙的に指定され、移行作業の負荷を軽減することができます。

COBOL 方言	Enterprise COBOL for z/OS
ソース フォーマット	Enterprise COBOL for z/OS
メインフレームのコピー処理	COBOL for OS/390
デバッグ用にコンパイル	COBOL for MVS
EXIT PROGRAM を GOBACK として処理	COBOL/370 Release 1
詳細	VS COBOL II Release 4
.GNT にコンパイル	VS COBOL II Release 3
出力	VS COBOL II Release 2
指令ファイルを生成する	OS/VS COBOL
リコートファイル生成	DOS/VS COBOL
	ROCKET

Enterprise Developer でコンパイルチェック済みのプログラムは、クロス開発時に IBM メインフレーム上の再コンパイルでも非常に高い確率で正しく動作します。また、リプラットフォーム時には高い互換性を保ったモダナイゼーションが可能となります。

3. PL/I 言語のサポート

ANSI 規格準拠の PL/I コンパイラがレガシー化した PL/I アプリケーション資産のモダナイゼーションを支援します。

(注: ANSI 規格に準拠しない PL/I プログラムではモダナイゼーションの際、書き換えが必要となる場合があります。)

PL/I アプリケーションのサポート環境は以下の通りです。

- 32-Bit, 64-Bit サポート有: Windows,
- Red Hat Linux (Intel), SUSE Linux (Intel)
- 32-Bitのみサポート: AIX

稼働環境

- › Windows
- › Red Hat Enterprise Linux
- › Rocky Linux
- › SUSE Enterprise Linux
- › Oracle Linux
- › Amazon Linux
- › AIX

サポートするIDE

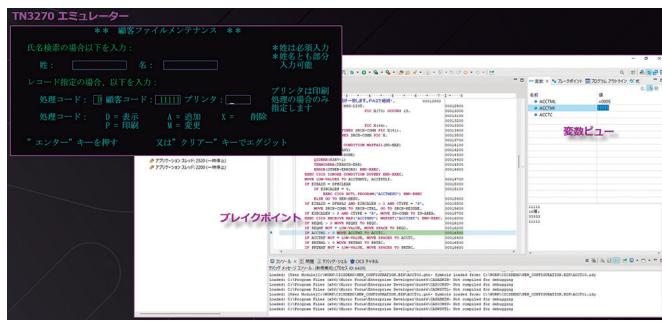
Eclipse, Visual Studio, Visual Studio Code

サポートするアプリケーション

COBOL (Enterprise COBOL, COBOL for 390, COBOL for MVS, VS COBOL/2など), PL/I, JCL (JES2, JES3, VSE), CICS, IMS DB/DC
EBCDIC/ASCII 選択可

※ Enterprise Developer の稼働環境について、最新のサポート状況は弊社ホームページにてご確認ください。

※ 開発環境にVisual Studioをご利用の際は、Visual Studioを事前にインストールする必要があります。



CICS アプリケーションのデバッグ

IMS 環境では、IBM メインフレーム上でアンロードされたIMS データベースをEnterprise Developer のIMS 環境にロードすることで、IMS アプリケーションの単体テストが可能となります。

プログラムはIBM メインフレームと同様のCALL ‘CBLTDLI’またはEXEC DLIコマンドを発行することができます。IMS/DCについてもトランザクション定義を設定することによってMFS 画面を経由したプログラムのテスト・デバッグが可能となります。

5. コンテナ型仮想化に対応

Enterprise Developer は64 ビットのベースイメージをご提供しており、Docker やPodman の利用をサポートしています。

Modernization. Without Disruption.™