

Rational software

IBM Rational Software Architect Standard Edition

主な特長

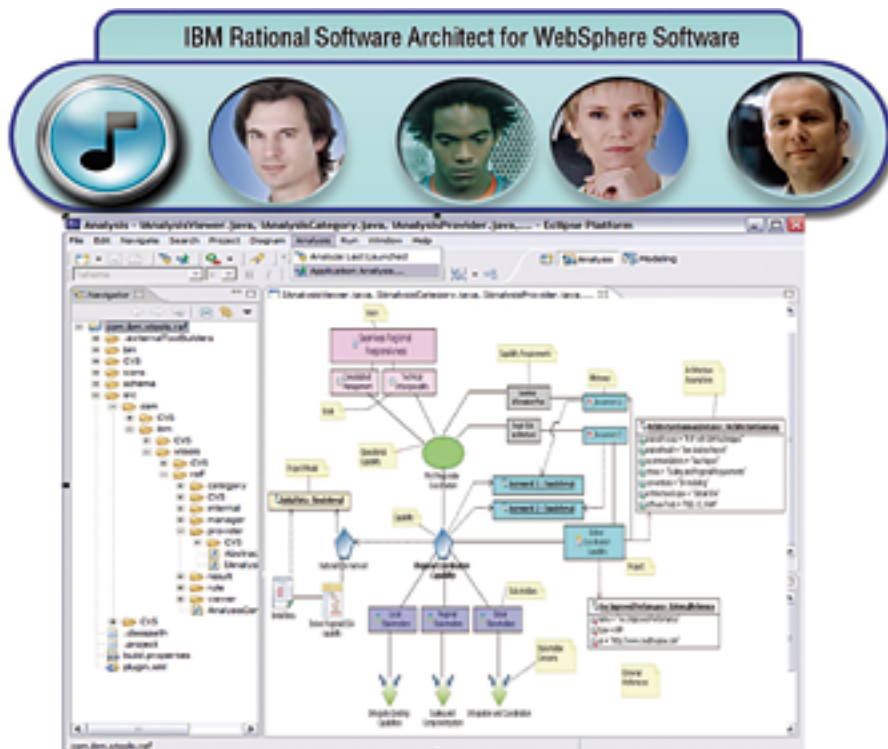
- 抽象化、自動化、簡素化を実現する各種ツールにより、リスクの削減とソフトウェアの開発・配備の迅速化を支援
- ビジネスに合わせた設計環境を構成するための各種ツールを提供
- UML 図での Java/C++ コードのデータ構造と動作の視覚化および編集が可能な、統合された設計・開発環境
- 実証済みのプログラミング・モデルおよび先進的なプログラミング・モデル用の包括的な設計ツールと開発ツール
- ソフトウェア・ライフサイクルおよび開発チームのプロセスと統合

ソフトウェアのデリバリーを成功に導くためには、プロジェクト・リスクの軽減、問題領域とソリューション領域の明確な記述とコミュニケーション、新しい技術の迅速な習得と適用、効果的なチーム・コラボレーション、開発のベスト・プラクティスの再利用と自動化などを支援するツールが必要となります。IBM Rational® Software Architect Standard Edition Version 7.5 は、このような目標の達成を支援するアプリケーションです。

統合された強力な設計・開発環境である IBM Rational Software Architect Standard Edition を使用すれば、システム・アーキテクト、ソリューション・アーキテクト、開発者は、チーム内で、世界中で、またさまざまな技術分野にわたって、システムとソリューションを把握、設計、管理、開発できます。IBM Rational Software Architect Standard Edition の抽象化機能、分析機能、レポート機能は、効果的なコミュニケーションとコラボレーションを促進します。また、自動化ツールとインテリジェントな編集ツールは、生産性の向上とアーキテクチャー管理の強化を支援し、Java™ Platform, Standard Edition (Java SE)、C/C++、CORBA ソフトウェア・コンポーネントといった開発環境における設計からコーディングへの移行を容易にします。

抽象化によりソフトウェア・デリバリーを簡易化

おそらくホワイトボード図やテキスト文書といった抽象化手法をすでに利用していることと思いますが、これらの手法は、プロジェクトの関係者が各地に分散している環境では使用するのが困難です。要件が変化し、実装プロセスで新たな技術的観点が明らかになるに従って、このような設計はすぐに陳腐化してしまいます。その都度このような設計書に戻って仕様を更新するには、コストがかかり、納期が迫っているなどの切迫した状況では現実的には不可能な場合もよくあります。さらに、このような設計の仕様は意味論的な正確性のない「不明瞭なデータ」にしかならず、自動分析に使用したり、アーキテクチャーの統一性や管理性を確保するために活用したりすることができません。



要求情報は、さまざまな方法で表現できます。Rational Requirements Composer は、入り組んだ情報を取り込み、関連付けて要求を定義できるようにします。

IBM Rational Software Architect Standard Edition は、次のことを実現するプラットフォームを提供します。

- 要件と仕様のコミュニケーションとコラボレーション
- プロジェクトのサイズ、複雑さ、状況およびリスクの分析と管理
- アーキテクチャーの統一性と整合性の管理
- プロジェクトの成果の改善

Rational Software Architect Standard Edition に組み込まれているモデル・テンプレート、プロファイル、デザイン・パターン、変換機能などの各種機能を使用すれば、統一モデリング言語（UML）分析や設計モデルといった抽象化手法に基づく文書を迅速かつ効率的に作成できます。また、組み込みのプロファイル・エディターである Rational Software Architect Standard Edition を使用すれば、各プロジェクトの問題領域とソリューション領域に適合するドメイン固有のソリューション分析・設計環境を作成できます。

その後、ボタンをクリックするだけで、プロファイル固有のモデル・テンプレート、パレット、コンテキスト・メニュー、プロパティ・ビューなどを生成できます。また、事実上の業界標準である DoDAF（Department of Defense Architecture Framework）もサポートしています。

自動化により、コストを削減しながら開発を迅速化

意味論的なリッチな抽象化にはそれだけでも価値がありますが、それらを開発の自動化に利用することでさらに大きい見返りを得ることができます。Rational Software Architect Standard Edition には、モデルからコードへの変換機能およびコードからモデルへの変換機能が組み込まれています。これらの機能は、Java、C/C++、XSD（XML Schema Definition）、CORBA、C#（IBM Rational Modeling Extension for Microsoft® .NET ソフトウェアを追加した場合）に対応しています。

さらに、IBM Rational Software Architect Standard Edition には、これらの組み込みの変換機能に加え、モデルからモデルへの変換機能やモデルからテキスト/コードへの変換機能を独自に作成するための強力なツールが用意されています。UML ベースや DSL（Domain-Specific Language: ドメイン固有言語）ベースの抽象化とともにこれらのツールを活用すれば、実装コードのかなりの部分を生成することができます。これにより、投資の大きな見返りを実感できるはずです。生成されたコードは多くの場合一貫性が高く、ハンドコーディングによるミスを含んでいないため、必要なテスト量やコンポーネントやシステムの全ライフタイムにわたるメンテナンス作業が減少し、さらにコストを削減できます。

構造的レビュー機能と管理機能による Java アーキテクチャーの維持

Java コード用の構造的レビュー機能と管理機能を利用することによって、適切なアーキテクチャーで構築された Java アプリケーションの構造を維持することができます。既存のコードを利用してプロジェクトに着手する場合、設計担当者は構造的レビュー機能を利用することによって、設計と開発を始める前にそのシステムの構造を把握できます。それにより、好ましくない副作用（例：想定していなかった依存関係によって引き起こされる、機能、スケーラビリティ、メンテナンス関連の問題）を取り入れずに、システムに適切な変更を加えることができます。また、既存のデザイン・パターンを視覚的に表示し、構造的なアンチ・パターン（ソフトウェアのメンテナンスとアップデートを困難にする好ましくない典型的な問題）を自動的に検出できます。

拡張性の高いオープンな Eclipse プラットフォームの活用

Rational Software Architect Standard Edition は、オープンな Eclipse 3.4 プラットフォーム上に構築されているため、個々のプロジェクトの要件に応じて簡単にソフトウェアの機能を拡張できます。Microsoft Windows[®] および Linux[®] 上で提供される Rational Software Architect Standard Edition には、サードパーティー製のプラグインを組み込むことができます。例えば、SysML (Systems Modeling Language) モデリングやモデルベースのシミュレーション機能を Rational 環境に追加できます。Rational Software Architect Standard Edition と Eclipse プラットフォームでは個々の機能がモジュール方式で提供されているため、インストール環境をかなり自由に構成できます。また、このようなモジュール方式が採用されているため、Eclipse 上でデータ、アプリケーション、各種ソフトウェア開発環境を統合することができます。例えば、同じ Eclipse 環境に、包括的なデバイス開発環境である Wind River Workbench を組み込むことができます。

開発と配備のギャップを埋める

システム/ソフトウェア開発企業では、しばしば開発担当チームと配備担当チームの間のコミュニケーション・ギャップという問題に直面します。例えば、配備にたびたび失敗するために、インフラストラクチャーの再構成、パッケージの再作成、場合によってはコンポーネントのアーキテクチャーの再設計を緊急に行わなければならない、といった問題が発生します。

IBM Rational Software Architect Standard Edition に組み込まれている配備アーキテクチャー・プラットフォームを使用すれば、初回から常に実装可能なソフトウェア・コンポーネントを構築できます。このプラットフォームでは、システム・インフラストラクチャーのリッチな意味論的表現を取り込み、ベスト・プラクティスに基づく標準テクノロジーの構成を記述したテンプレートを作成できます。また、論理的な運用モデルを生成し、それを土台として開発段階のアプリケーションの配備要件を記述することができます。

ソフトウェア・アーキテクトはこれらのツールを利用してコンポーネントの配備要件を引き出し、そこで得られた情報に基づいて、要件に釣り合った技術インフラストラクチャーを設計するという作業を行うことができます。

IBM Rational Software Delivery Platform との統合

開発の全ライフサイクルにわたって、IBM Rational ClearCase[®]、Telelogic[®] Synergy[™]、IBM Rational Team Concert といった堅牢なソフトウェア構成管理製品を使用することにより、コラボレーションの対象ファイルを管理できます。変更は、Rational Team Concert、IBM Rational ClearQuest[®] または Telelogic Change[™] で管理できます。Rational Software Architect Standard Edition を IBM Rational Asset Manager と統合すれば、プロセス、ソフトウェア、情報、配備モデルの一部を、再利用可能なアーキテクチャーの構成要素として管理できます。また、カスタム・プロファイル、テンプレート、パターン、変換処理およびその他の主要な実行可能資産も管理できます。さらに、IBM Rational Unified Process[®] (RUP[®]) および RUP Systems Engineering プラグイン・ソリューションと統合すれば、文書化された単一のプロセスに従って作業を進めることができます。

IBM Rational Software Architect Standard Edition を お勧めする理由

競争力の高い優れたソフトウェア集約型システムを生産するためには、ソフトウェアの開発・配備の最適化が不可欠です。システムの適用範囲が広がり複雑さが増している今日でも、低コストで高品質の製品を提供することが重要であることには変わりありません。

IBM Rational Software Architect Standard Edition Version 7.5 を使用すれば、開発コストを抑えながら高品質のソリューションを迅速に開発できるとともに、ソリューションの全ライフタイムにわたるメンテナンス・コストも削減できます。

詳細の参照先

IBM Rational Software Architect Standard Edition についての詳細は、IBM 担当員または IBM ビジネス・パートナーにお問い合わせいただくか、以下の Web サイトを参照してください。

ibm.com/jp/software/rational/products/rsd/



お問い合わせは、IBM ビジネス・パートナー、製品販売店、弊社営業担当員または、ダイヤル IBM (0120-04-1992) へ。
受付時間：月～金 9:00～18:00 (祝日 12/30～1/3 を除く)
携帯電話でおかけのお客様は下記の電話番号でご利用ください。
ダイヤル IBM 03-6220-8002 (この場合通話料金はお客様のご負担となります)。

©Copyright IBM Corporation 2008

日本アイ・ビー・エム株式会社
〒106-8711 東京都港区六本木 3-2-12

Produced in Japan
2008 年 11 月
All rights reserved

IBM、IBM ロゴ、ibm.com、Rational および Telelogic は、International Business Machines Corporation の米国およびその他の国における商標です。これらおよび他の IBM 商標に、この情報の最初に現れる個所で商標表示 (® または ™) が付されている場合、これらの表示は、この情報が公開された時点で、米国において、IBM が所有する登録商標またはコモン・ロー上の商標であることを示しています。このような商標は、その他の国においても登録商標またはコモン・ロー上の商標である可能性があります。現時点での IBM の商標リストについては、ibm.com/legal/copytrade.shtml の「Copyright and trademark information」をご覧ください。

Microsoft および Windows は Microsoft Corporation の米国およびその他の国における商標です。

Java およびすべての Java 関連の商標およびロゴは Sun Microsystems, Inc. の米国およびその他の国における商標です。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。

本書に記載の製品、プログラム、またはサービスが日本においては提供されていない場合があります。日本で利用可能な製品、プログラム、またはサービスについては、日本 IBM の営業担当員にお尋ねください。

本書に含まれる情報は、情報提供のみを目的に提供されています。本文書の情報の完全性および正確性については最善の努力を払いましたが、本文書の内容は法律上の取極担保責任を含むすべての明示もしくは黙示の保証責任を負うことなく、現存するままの状態を提供されます。また、本文書は IBM の現在の製品プランまたは戦略に基づくものです。この製品プランまたは戦略は予告なく変更されることがあります。IBM は本文書およびその他関連文書の使用に起因するいかなる損害についても責任を負いません。本文書は、IBM (または IBM のサプライヤーまたはライセンサー) にいかなる保証責任を負わせるものではなく、また、IBM ソフトウェアの使用に際し適用される、プログラムのご使用条件の内容も変更するものではありません。