

# Outsourcing Accelerator

## － オフショア開発を推進するRational®ソリューション －

日本において、オフショアを利用したソフトウェア開発は、2007年まで年率30%の勢いで増加していました。2008年からは経済状況の変化で、その増加率はいったん鈍化しましたが、今後も引き続き増加傾向にあるといわれています。そうしたソフトウェア開発環境の変化の中、IBMではオフショア開発をさらに進めたグローバル開発を10年ほど前から大きく進めてきました。特にRationalでは最新の戦略の中心であるJazz™関連の開発に、グローバル開発が重要な役割を果たしています。実際、中核製品のRational Team Concert™ (RTC)には、地理的に分散した環境で協調して開発を進めるための機能が多くサポートされています。このRTCの機能とIBMの経験を基に、ソフトウェアのオフショア開発のためのソリューションを作成し、提供しています。本解説では、こうしたオフショア開発の現状と課題を概観し、それらの課題の解決を目指す、このソリューションについて紹介します。

### ① はじめに

ソフトウェアのオフショア開発は、2003年ぐらいから本格的に件数が増加し始め、2008年の金融危機直前まで年率30%の勢いで増え、同年のオフショア取引額が1,000億円を超えるまでに至りました。その後の増加率は、世界経済停滞の影響で鈍っていますが、今後再び増加に転じると予測されています [1]。特に従業員1,000人以上の大企業に限れば、すでに60%以上がオフショアを利用しており、そのうち80%以上が現在でも利用を拡大する傾向にあるといわれています。

IBMでは、すでにインド、中国がグローバルに見て最大規模の開発拠点になっており、ソフトウェアの製品開発では全体の約30%の開発者がインド、中国などのEmerging地域と呼ばれる国々の開発者で占められています。現在IBMのRationalでは、今後のソフトウェア開発におけるさまざまな課題に対応するためのプラットフォームとしてJazzを推進しており、オフショア開発を含めたグローバル開発のサポートを強化しています。実際

## Outsourcing Accelerator:

### - A Rational Solution for Offshore Software Development -

The offshore development of software is a growing trend despite the current economic situation. IBM has been working in this area over the last ten years in a more advanced way, namely, the global development of software. Rational is a development of our strategic Jazz-based products into a globally distributed model. In particular, Rational Team Concert(RTC) has many features helpful for such models and is used extensively in internal development. Based on these features, as well as on IBM's experience, we have created an offshore software development solution based on RTC. This article explains the challenges of offshore development and presents main features of this solution.

Rationalでは、RTCなどJazzに基づいた製品開発を中国、インドを含めた世界10数カ所の拠点で協調しながら開発を進めており、Jazzプラットフォームを利用してグローバル開発を効率化しています。

そこで今回、わたしたちは中国の開発部隊と協力してJazzのプラットフォームを利用して、RTCを中心にした製品を使い、オフショア開発のためのソリューションを構築しました。本稿ではオフショア開発の現状と課題を簡単に説明した後、このソリューションについて解説します。

### ② オフショア開発の現状と課題

参考文献 [1] では、国内の数百社のIT企業とユーザー企業のアンケート調査に基づいた結果を基に、最近のオフショア開発の現状がまとめられています。それによると、オフショア開発を利用する最大の理由は、約95%の会社でコスト削減となっており、欧米も同様であるといわれています。それに次いで多いのは、国内で不足している人材の確保といわれています。

次にオフショア先の国ですが、日本の場合圧倒的に中国が多く、80%を超える会社が中国を利用しています。地理的な近さ、人材確保の容易さ、日本語でのコミュニケーションが可能なことなどが寄与していると思われます。これに対してインドは2007年以降やや減少傾向で、2009年にはベトナムがインドを超えて2番目のオフショア先になったと報告されています。

オフショア利用の評価を見ると、全般的にコスト削減が当初期待したレベルに達していないといわれています。例えば、中国の件費は日本の開発者の大体4分の1から3分の1といわれていますが、それを考えると理論的には開発コストを半減させることも可能なはずですが、文献[2]によれば、実際には25%程度の削減実績にとどまっている案件が最も多いと報告されています。図1に文献[1]で報告されているデータを載せましたが、これによると約13%の会社では実質的な削減効果が見られず、30%の会社では削減額が1割以下となっています。しかし、その一方で4割以上の削減に成功しているという会社も15%強あります。オフショア開発がコスト削減につながっている会社とそうでない会社にかかなりの開きがあるというのが現状です。

コスト削減が十分に達成できない理由には、いろいろな原因が考えられますが、よく挙げられるのは、言葉や文化、

さらに地理的に離れていることでコミュニケーションがうまく取れないこと、仕様やプロジェクト計画などに関して合意形成が不十分なこと、オフショア先の進捗よなどの情報提供が不十分なこと、さらに品質に関する考え方の違いからくるトラブルなどがあります。そうした課題への解決策として、ブリッジSEが間に入って改善に努めるのですが、多くの場合十分ではなく、結果として手戻りが起こったり、頻繁に直接会って会議を行うことが必要になったりということで、かえってコストが増えてしまうのが現状です。特に日本側の要求する品質基準は、一般には非常に高いので、オフショア先の認識とのずれが問題を起こすケースが多いといわれています。中国に限らず日本の会社にとってオフショア先に対して共通に満足度が低いのは品質であると報告されています。同じ文献[1]ではその理由とされる点についてのデータが図2のように示されています。

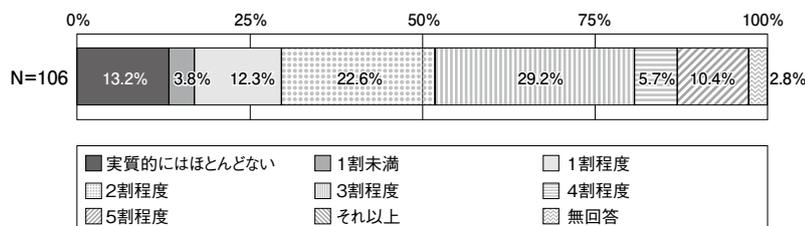
### ③ IBM におけるグローバル開発

IBMでは製品開発とサービス・デリバリーの分野でオフショアでの開発を積極的に活用してきました。特にインドと中国は製品開発、サービス・デリバリーともそれぞれ数千人の規模でのソフトウェア開発拠点となっています。サービス・デリバリーにおける状況や課題、解決施策などについて

は文献[3]を参照してください。本章では製品開発での状況について、特にRationalグループを例に説明したいと思います。Rationalグループは、開発ツールを中心としたソフトウェア開発のソリューションを提供しているグループであり、そこでは自らの製品を実際の製品開発に活用しています。特に最新のJazzベースの製品群の活用はIBMの中で急速に広がり、Rationalでの事例がIBMの中で典型的な方法になりつつあります。

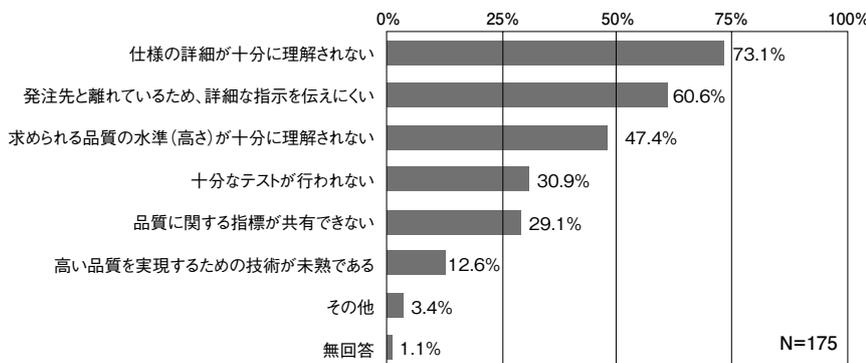
現在のRationalグループは、図3に示したように主なサイトだけで世界で14カ所、約2,000人の開発者が分散して開発しています。ほとんどの製品が、少なくとも3カ所以上のサイト協調となる開発体制になっています。

こうした環境で開発を進めるのに重要な要因としては、共通の開発プロセ



出典：独立行政法人情報処理推進機構，2010年IT人材白書，(2010). p75

図1. オフショア開発によるコスト削減効果



出典：独立行政法人情報処理推進機構，2010年IT人材白書，(2010). P81

図2. オフショア開発における品質管理が難しいと考える理由

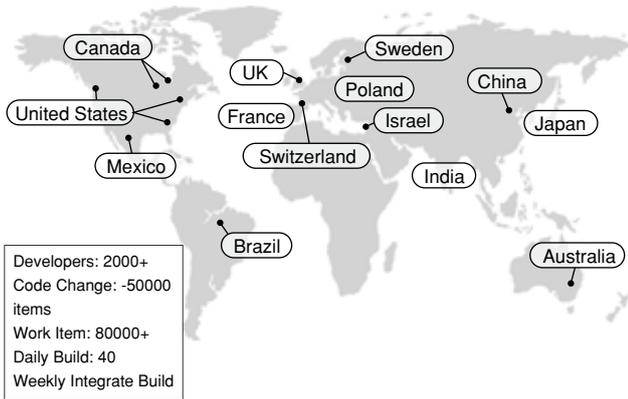


図3. IBM Rationalの主な開発サイト

スとチームの自主性を尊重しながら推薦する方法論の利用と支援、そして世界中で共通に使える開発インフラとツールのサポートが挙げられます。

開発プロセスについて、IBMでは共通のIPD (Integrated Product Development) というプロセスに従って行われています。これは製品を開発する上で重要なポイントとなり、ビジネス計画や開発プラン、リリースやサポートなど情報を共有し、討議して決定するものです。

開発の方法論は、さまざまなスタイルが採られていますが、最近ではアジャイル開発が多くなっています。Rationalの場合70%以上がXP (エクストリーム・プログラミング) やスクラムなどのアジャイルの方法論を採用し実践しています。

最後に開発インフラとツールですが、これは極めて重要であり、こうした分散したサイトで協調し、特にある程度以上の規模で統一して有効な開発を行うとするとツールなしでは不可能でしょう。このために利用しているのがJazzのプラットフォームと、それに基づいた製品群です。具体的にはRational Team Concert (以下、RTC)、Rational Quality Manager (以下、RQM)、そしてRational Build Forge® (以下、Build Forge) が中心です。またこれらに基づく共通のインフラはRational Engineering Service (RES) といわれる専任グループがホスティングしています。図4にこのインフラの絵を示します。

こうした環境を利用して分かってきたことは、地理的に離れたサイト同士で協調するには、幾つかの要素が極めて重要であるということです。それはまず、情報の一元管理と透

明性です。世界中どこでも情報や成果物が一元管理されていて、プロジェクトにかかわる人は基本的には誰でも共有できるということです。またいろいろな情報や成果物は関連がありますから、それらの関連が管理されていてトレーサビリティが確保されていることも重要です。次はプロジェクト・チームが使っているプロセスが明確になっていて、現在どの時点にいるのかが分かること、誰が何をやって進捗がどうか一目で分かることです。さらに開発者や管理者がほかのサイトの人と協力するために会議をスケジュールしたりせず、ネットワーク上で簡単に打ち合わせできるなどのコラボレーション機能の充実などが挙げられます。また協力して何かを決定したときは、その理由や経緯がデータベースに保存され、共有できることも重要です。

こうした経験とツールの中心的な役割を果たしているRTCとRQMをベースにして、多くのお客様が活用できることを目指したのが今回紹介するオフショア開発のソリューションです。Build Forgeは大規模なビルドを頻繁に行う必要があるときには不可欠となりますが、そうでない場合はRTCのビルド機能で代行できますので、今回のソリューションには組み込まないことにしています。もちろん必要に応じてBuild Forgeを組み込むことも可能です。

ソフトウェア開発は、外部調達・供給のプロセスの有無によってプロセスが大きく変わります。IBM Rationalのように、開発に社内リソースを利用する場合には調達・供給といったプロセスは軽くなるのでRTC/RQM単独でも効果があります。しかし社外リソース、今回の場合オフショア側への契約型開発業務委託となると少し事情が異なっ

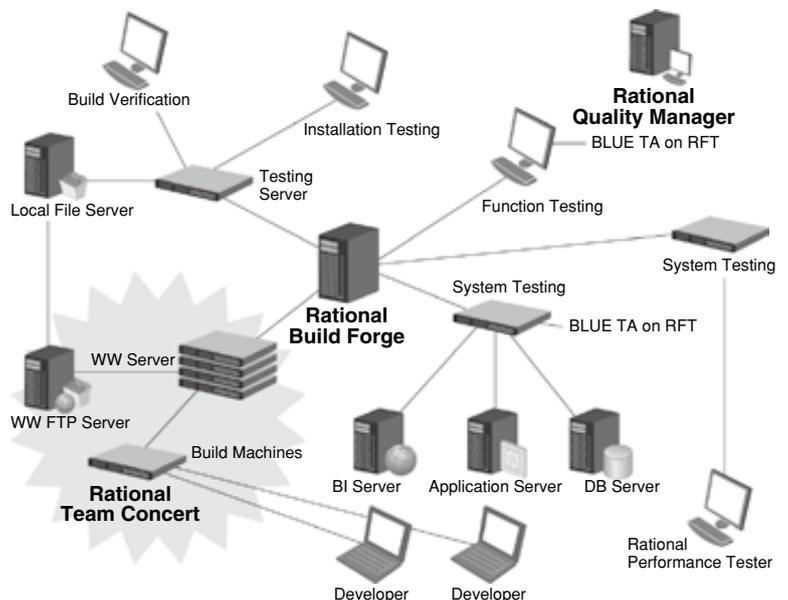


図4. Rationalでのグローバルな開発インフラストラクチャー

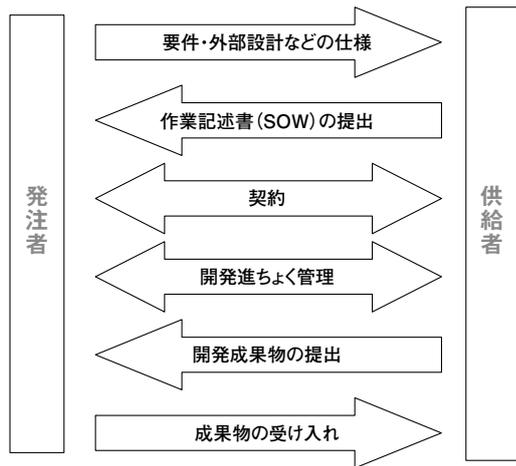


図5. 発注者、供給者での取得・供給プロセス  
※典型的なプロセスの全体像は文献 [4] を参照

できます。今回の Outsourcing Accelerator ソリューションは、RTC/RQM による効果の上に、日本側・オフショア側で典型的に発生する調達・供給のプロセスを実装することで Jazz による協業プロセスや開発の効率化を促進することを目指しました。

#### 4 オフショア開発のプロセスと強化ポイント

オフショア開発の大多数は、発注者と供給者との間での契約に基づいてなされます。その形態はプロジェクトごとの契約から年間である程度の開発リソースを確保した契約に基づく場合など形態に違いはありますが、おおむね仕事の流れに大きな違いはありません。ここではプロジェ

クトごとに契約する場合の大まかな調達および開発の流れを図5に示しました。発注者はソフトウェア要求、外部設計などの仕様を基に、開発工程の一部をオフショア会社に業務委託します。オフショア会社（供給者）は、作業見積もりを実施した上で、作業記述書（Statement of Work:以下、SOW）を作成し、契約プロセスに入ります。契約後は、作業記述書に記載された作業内容を実施します。成果物を受け入れた発注者は、受け入れ検査作業を実施し、受理します。

開発進捗管理のフェーズにおいては、監視・コントロールが重要になり、以下の項目の強化が求められます [5]。

- 質疑応答（Q&A）による要件確定プロセス
- 品質メトリックスの測定・監視プロセス
- 追跡可能性（トレーサビリティ）監視プロセス
- 公式レビューの定期実施・監視プロセス

質疑応答（Q&A）は、オフショア開発で発生しがちな「要求が正しく伝わらない」という問題への対処方法として利用されています [5] [6]。オフショア側は、日本側からの要求仕様や外部設計の説明では、文書記述に漏れがあったり、あるいは日本側からの説明不足や、オフショア側の理解不足もあって正確に伝わらない場合があります。そのため、質疑応答にて仕様や動作記述についてオフショア側から質問をし、日本側が回答するプロセスが有効な手段として活用できます [6]。Q&A プロセスは多くの場合、電子メールでのやり取り、あるいはそのやり取りを表計算ソフトウェアに転記するといったような地道な作業になりがちです。結果、メールが返ってこない、あ

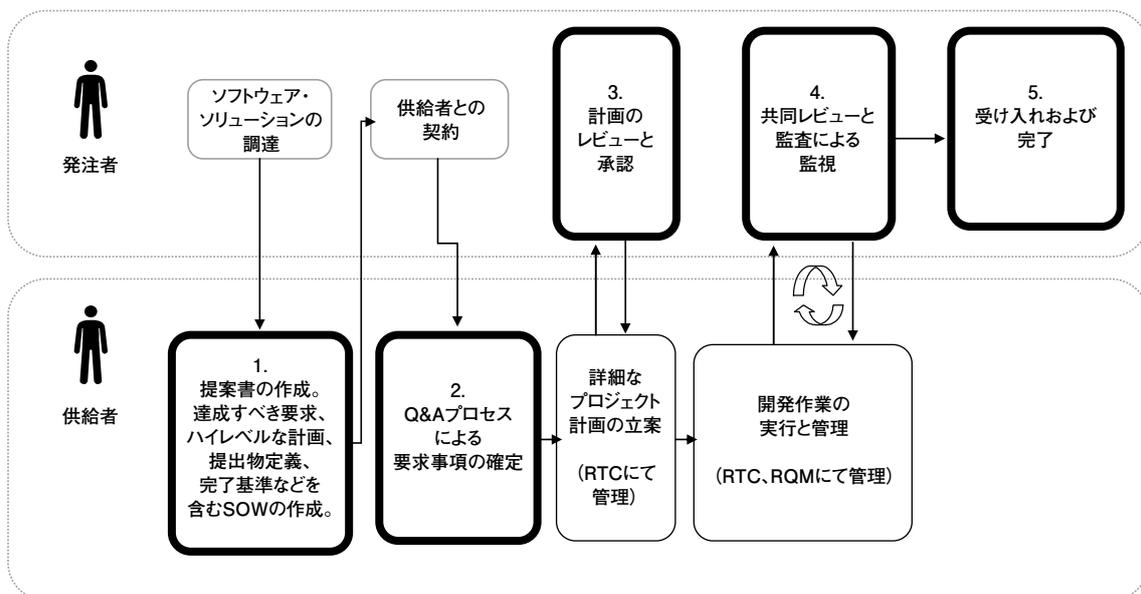


図6. Outsourcing Acceleratorでサポートするプロセス

るいは転記漏れといったミスも誘発しがちです。

ソフトウェア品質に関しても期待値に達しない場合が多く(図2) その要因として、オフショア側のスキル不足、双方のコミュニケーション不足、また日本とオフショア側での品質基準が異なるためといわれています。このため、双方が理解できる品質メトリックスを定義し、測定するという方法が一般的に採られています。

追跡可能性(トレーサビリティ)はCMMI(Capability Maturity Model Integration)では必須の技術ですが[7]、オフショア開発でも有効な手段です。オフショア側に要件の変更が正しく伝わっていないために、結果として期待と異なるソフトウェアが納入されることもあります。

レビューはオフショアに限らず実施されますが、日本側が要求するレベルに達しているかどうか、あるいは指摘事項についてアクションを取ったかどうかを確認する重要なプロセスになります。

そのほかのオフショア要素技術としては

- オフショア開発に耐えられるドキュメント記述方法
- グローバル化に対応した開発標準の作成
- 地域・文化を超えたグローバル・コミュニケーション・スキル
- グローバル対応のリスク管理手法

などがあり、日本側の体制も大きく変化する必要があります。これらに関しては、参考文献 [3] [5] および [8] を参照していただきたいと思います。

## 5 Outsourcing Accelerator のサポートするプロセスと機能

Outsourcing Accelerator は、オフショア開発で典型的に発生する図6に示したプロセスをサポートしています。このプロセスをJazzプラットフォーム[9]に実装することで、日本側、オフショア側での協調作業を行いやすくしています。このプロセス中でソリューションの調達や供給者との契約などの部分は、すでにほかのシステムで実施されることが多いため、ここでは対象外とし、プロジェクト計画の立案や開発作業の実行と管理などはRTCやRQMの機能をそのまま使うことを想定しています。従ってOutsourcing Acceleratorで固有にサポートする機能は主に次のようになります。

1. 作業記述書(SOW)作成による合意形成支援機能
2. 要件、開発タスク、テストケース、Q&Aの追跡可能性(トレーサビリティ)機能



図7. Outsourcing Acceleratorが作成するSOW文書レイアウトの重要な部分

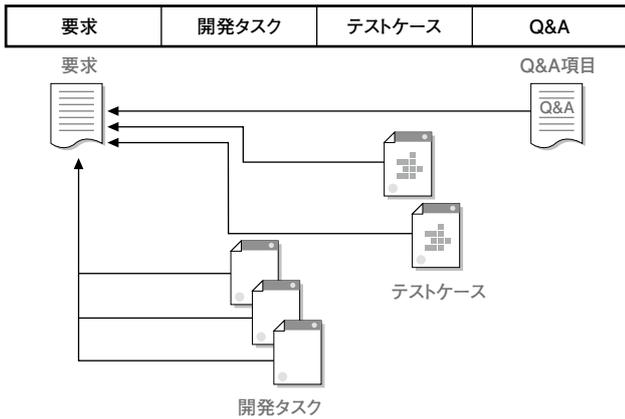
3. レビューとその進捗管理機能、およびリスク管理機能
4. 開発ダッシュボードによるプロジェクト進捗、品質メトリックスの測定、監視機能
5. 受け入れ検査支援機能

Outsourcing Acceleratorは、RTC/RQMと連携することにより、日本側で必要となるプロジェクト管理データ、品質メトリックス・データも取得することが可能となっています。

### 5.1 作業記述書(SOW)作成による合意形成支援機能

オフショア・プロジェクト失敗の原因の1つは、開発の作業範囲(スコープ)や、受け入れ・完了基準について双方の理解が不十分であることが考えられます。そこで、Outsourcing Acceleratorではカスタマイズ可能なSOW作成を支援する機能を持たせました。ここでサポートするSOW文書レイアウトの主要な部分は図7のようになっています。

プロジェクトの目的や体制といった文章で記述する部分は、リッチテキストで記述します。セクションの一部はプロジェクト管理ツールや品質測定ツールと連携します。例えば、プロジェクトの計画を記述すると、RTCのプロジェクト計画に反映され、また進捗がこの合意文書に反映されるようになります。受け入れ基準となる品質メトリックスはRTC/RQMにより測定された品質データが反映されるよう



要求から開発タスク、テストケース、関連するQ&Aを追跡可能にすることで、オフショア側の要求事項への理解度の把握、また要求の変更への追跡ができるようになる。

図8. 要求、開発タスク、テストケース、Q&Aへの一貫したトレーサビリティ

になります。このSOWはいわば、“生きた合意文書”となり、合意通りに作業が進んでいるかを確認する場にもなります。

## 5.2 要件、Q&Aのトレーサビリティ機能

オフショア側では、要求に対する質問を提出し、日本側から回答することによって要求に対する理解が深まり、要求事項の確定につなげることができます。また、Outsourcing Acceleratorでは、プロジェクト管理ツール(RTC)で管理された開発タスク、テスト管理ツール(RQM)で管理されたテストケースを関連付けられます。これにより合意された要求(開発対象-作業範囲)は、Outsourcing Acceleratorによりトレーサビリティが確保されます(図8)。

日本側ではオフショア側の理解度をQ&Aの測定により

確認ができます。例えば、1つもQ&Aがない要求について理解されているかどうかを確認したり、多くのQ&Aが必要となる要求に対しては、日本側で要求に対応する文書の詳細化など見直しを実施したりすることで、双方の理解度を深めることができるようになりますと期待されます。またこのトレーサビリティにより実装漏れ、テスト漏れを防ぐことができるようになります。仮に要求に変更があった場合でも日本側・オフショア側で影響度の把握ができるようになります。

## 5.3 レビューとその進捗管理、およびリスク管理機能

本機能は、Outsourcing Acceleratorに追加した機能ではなく、RTCをカスタマイズしたワーク・アイテム“Review”および“Risk”を利用します。Outsourcing Acceleratorでは、“Review”ワーク・アイテムを利用することで、レビューの結果となった抽出項目を作業タスク、あるいはリスクとして関連付け、管理することができます。例えば、作業タスクが障害修正の場合、障害修正のためのワーク・アイテムを起票し、関連付けられます。障害修正のためのワーク・アイテムは、RTCの計画にすぐに反映されます。すなわち、計画は実作業ベースに修正され、現実を反映したものになります。ただし初期計画からずれが生じることにもなるので、影響範囲が大きな場合は、リスクとして判断し、直ちに回避措置を取るといった素早い行動を取ることを可能にします。レビュー、リスクへの対応はダッシュボードでその進捗を確認することができます。

## 5.4 開発ダッシュボードによるプロジェクト進捗、品質メトリックスの測定、監視機能

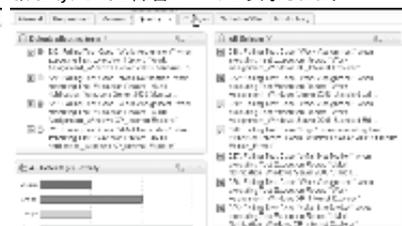
Outsourcing AcceleratorはプラットフォームにJazzを採用しているため、ダッシュボード機能が使えます。ダッシュボード機能とは、重要なメトリックスを可視化し、1つの画面にて表示する機能です(図9)。

Outsourcing Acceleratorでは、「まだ回答をしていない質問」「クローズしていないレビュー」「処理されていない変更要求(変更管理)」などのオフショア開発で必要な監視機能を持たせるとともに、RTC/RQMからのデータをリアルタイムで

RequirementタブではQ&Aの進捗が管理されます



Qualityタブでは障害レポートが表示されます



Changeタブでは、計画などの“変更依頼”が表示されます



ダッシュボードの出力形式はカスタマイズ可能です。例えば、バグ収束曲線や、バーンダウン・チャートなどをダッシュボードのレポートに含めることができます

図9. ダッシュボード機能

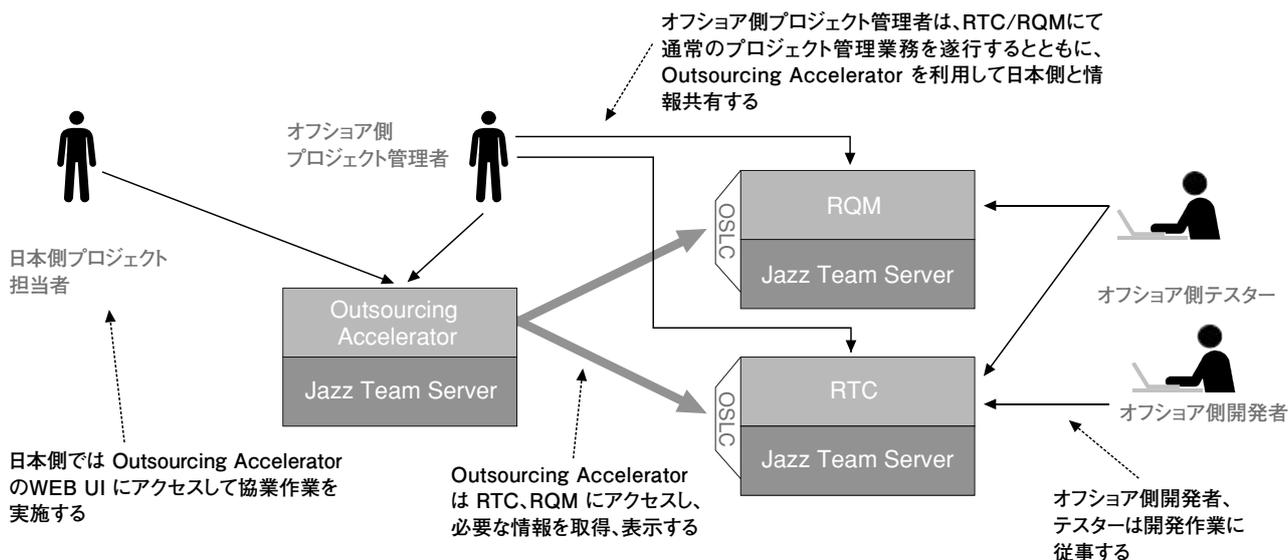


図10. Outsourcing Acceleratorアーキテクチャーイメージ

表示させ、RTCで管理された「プロジェクトの進捗」、RQMで管理された「テストの進捗」、RTCで管理された「欠陥の収束」といった一般的なプロジェクト管理指標も同時に表示が可能です。プロジェクト進捗報告などは、オフショア側が手作業にて作成したレポートを使うことが多いと思われそうですが、Outsourcing Acceleratorでは実際のデータに基づくメトリックスをリアルタイムでダッシュボードに表示するため、自動化され、オフショア側での進捗報告作業負担軽減にもつながります。

### 5.5 受け入れ検査支援機能

Outsourcing AcceleratorのSOW機能には、「納品物」を埋め込む、添付ファイル機能があります。開発初期段階に合意された作業記述書に基づいて納品することで、過不足なく正しい成果物の納品の履行ができるようになります。

納品物を提供された日本側では、通常「受け入れ試験」を実施します。Outsourcing Acceleratorでは、「受け入れ試験」を実施している途中のサポート依頼、障害連絡などもサポートする予定です。障害の一覧はダッシュボードで再び確認し、オフショア側での対応をリアルタイムで把握することができます。

## 6 Outsourcing Acceleratorのアーキテクチャー

Outsourcing Acceleratorは、Jazz Team Serverへのアドインとして実装されています(図10)。RTC/RQM

へは Open Service for Lifecycle Collaboration (以下、OSLC [10])で定義されたインターフェースを利用してデータにアクセスします。

ツールの展開方法としては3パターンを考慮していますが、基本的にWebにアクセス可能であれば、どの配置でも問題ありません。

### ツールの配置パターン

1. RTC/RQM、Outsourcing Accelerator サーバーが日本側にあり、オフショア会社が日本側にアクセスする場合。
2. RTC/RQM、Outsourcing Accelerator サーバーがオフショア側にあり、日本からは Outsourcing Accelerator のみにアクセスする場合。
3. RTC/RQM サーバーはオフショア側にあり、日本側に Outsourcing Accelerator サーバーが存在する場合。また Outsourcing Accelerator がオフショア会社のある Software Park などにホストされる場合もこの変形と考えられる。

Outsourcing Acceleratorのアーキテクチャーでは、日本側、あるいはオフショア側の管理担当者両方がアクセスすることを想定しています。別の見方をすると、日本側はあえて、RTC や RQM などにアクセスする必要はありません。オフショア開発の進捗はすべてこの Outsourcing Accelerator に集約されているので、定期的開催される報告会も Outsourcing Accelerator により管理されたデータで議論することになります。すなわち、事のいきさつにとらわれず、現在の品質、リスクの発生、進捗の遅れなど日本側にとって極めて重要な情報の把

握ができるようになります。これにより、従来発生しがちであったオフショア開発管理オーバーヘッドも低減できると見込まれます。

なお、OSLC はさまざまな会社・団体が参加し策定している開発ツール間の連携の仕様を公開しているコミュニティです。OSLC は、要求管理ツール、変更管理ツール、構成管理ツール、プロジェクト管理ツールへのアクセスの標準インターフェースを定義しています。Outsourcing Accelerator は、OSLC を利用してデータをアクセスしているため、RTC/RQM 以外の変更管理ツール、品質管理ツールとも連携できることが将来期待されます。

## 7 おわりに

最初にも述べましたように、ソフトウェアのオフショア開発は 2003 年ごろから本格的になりましたが、それを可能にした技術要素としてインターネットが地理的に離れた人々を結んで、グローバルな共同作業を可能にするように進化したことが挙げられます。IBM の Rational が戦略として位置付けている Jazz のプラットフォームは、そうしたインターネットを通しての共同作業をソフトウェアのライフサイクル全般で推進することを目指しています。今回説明した Outsourcing Accelerator は、そのメリットをオフショア開発の具体的な課題解決に適用したソリューションとなっています。実際 IBM の多くの開発やサービス部門が Jazz を利用することで、ソフトウェア開発のグローバル化された体制で成果を挙げています。こうした経験を基に Outsourcing Accelerator の効果を想定すると、コスト削減や品質向上、スケジュールの維持などを中心に高い効果が期待されると考えられます。インターネットがもたらす市場のグローバル化の中で、ビジネスを支えるソフトウェアの役割がますます重要になってきています。そうした中でインターネットのインフラを利用したグローバルなソフトウェアの開発や運用が進んでいくのは当然の流れといえますし、それをいかに有効活用できるかが企業の競争力を高め、差別化の要因になると思われます。Outsourcing Accelerator はその意味で重要なソリューションとなると期待されています。

### 【参考文献】

- [1] 独立行政法人情報処理推進機構, 2010年IT人材白書, (2010).  
[http://www.ipa.go.jp/jinzai/jigyuu/docs/ITjinzai2010\\_Hires\\_20100428.pdf](http://www.ipa.go.jp/jinzai/jigyuu/docs/ITjinzai2010_Hires_20100428.pdf)

- [2] 総務省情報通信政策局情報通信経済室, オフショアリングの進展とその影響に関する調査報告書, (2007.03).  
[http://www.soumu.go.jp/johotsusintokei/linkdata/other017\\_200707\\_hokoku.pdf](http://www.soumu.go.jp/johotsusintokei/linkdata/other017_200707_hokoku.pdf)
- [3] 藤本 卓司: No.63 進化するプロジェクトマネジメント, IBM PROVISION, (2009).
- [4] 情報処理推進機構, 共通フレーム2007, オーム社, (2007).
- [5] 幸地 司 (著), 霜田 寛之 (著), 北島 義弘 (監修), 倉田 克徳 (監修): 標準テキスト オフショアプロジェクトマネジメント【SE編】, 技術評論社, (2009).
- [6] 朱小紅 郭依群 呂響亮 (杭州東忠科技有限公司): “オフショア開発における中・日間の価値(判断)基準相違点の解決方法”, SQiP シンポジウム2009
- [7] CMU/SEI, CMMI for Development, Version 1.3, Carnegie Mellon University, (2010).
- [8] 幸地 司 (著), 霜田 寛之 (著), 北島 義弘 (監修), 倉田 克徳 (監修): 標準テキスト オフショアプロジェクトマネジメント【PM編】, 技術評論社, (2009).
- [9] Jazz Homepage : <http://jazz.net/>
- [10] Open Services for Lifecycle Collaboration, home page : <http://open-services.net/>



日本アイ・ビー・エム株式会社  
大和ソフトウェア開発研究所  
アジアパシフィック地区 ラショナル CTO

上村 務 Tsutomu Kamimura

### 【プロフィール】

1986年、日本IBM東京基礎研究所入社。プログラミング言語、ワークステーション・ソフトウェア、オブジェクト指向開発などの研究、マネジメントを経験。1999年に大和ソフトウェア開発研究所でストラテジーを担当した後、2000年に米国IBM Somers に出向。2002年から大和ソフトウェア開発研究所で Rational 開発を担当し、2007年には Rational Lexington ラボに出向。2009年より現職。



日本アイ・ビー・エム株式会社  
大和ソフトウェア開発研究所  
“Unleash the Labs” AP/Japan 担当

天野 武彦 Takehiko Amano

### 【プロフィール】

2003年、日本IBM入社。以前は、日本ラショナル・ソフトウェア株式会社勤務。2008年までさまざまなお客様に対する開発ソリューションの導入支援に携わる。2008年後半よりラショナル開発に異動し、“Unleash the Labs” 活動の AP/Japan 担当となる。2010年より現職。