

大規模プロジェクトでのコード品質を確保する 「Javaで脱COBOL」に挑む損保ジャパン 成功のために選んだ手段とは

損保ジャパンの基幹システム刷新は、複数ベンダーが携わる大規模な開発プロジェクトだ。開発者の技術レベルが異なり、プログラミング言語が「COBOL」から「Java」へと変わる中、ソースコードの品質をどうやって確保したのか。



損保ジャパン



SOMPOシステムイノベーションズの
加藤 修氏



SOMPOシステムイノベーションズの
黒澤直哉氏



SOMPOシステムイノベーションズの
福田浩一氏

企業を取り巻く状況が日々変わる中、既存のシステムに引きずられて変革のスピードが鈍ることは許されない。デジタルトランスフォーメーション（DX）を通じてシステムの開発サイクルを高速化し、さまざまな変化に応じてニーズにマッチしたサービスを提供することは、企業にとっての死活問題だ。

DXの基盤となるシステムの品質を確保する手段の一つが静的解析ツールだ。損害保険ジャパン（以下、損保ジャパン）は、オープン系技術を採用して基幹システムを約30年ぶりに刷新する「SOMPO-MIRAI」プロジェクトを進めている。その中で静的解析ツールを利用して、数千人の開発者が参加する大規模プロジェクトでソースコードの品質を確保することに成功したという。静的解析ツールの選定から活用、将来の見通しまでを担当者に聞いた。

数千人が携わるSOMPO-MIRAIプロジェクトの課題

国内大手損害保険会社の一社である損保ジャパンは、「安心・安全・健康のテーマパーク」構築を目指す SOMPO ホールディングスの中核として事業を展開している。世界情勢が不確実性を増す中で、顧客のニーズに応える高品質なサービスを迅速に提供するために、損保ジャパン、SOMPO システムイノベーションズ、SOMPO システムズは2015年、SOMPO-MIRAI プロジェクトを立ち上げた。損保ジャパンが1980年代に開発を開始したメインフレーム中心の基幹システムは、事業の拡大や合併などを経るたびに改修を続けた結果、肥大化して複雑度も増した。このため、新たなビジネスモデルや新商品のために基幹システムの改修や機能追加を迅速に実施することが困難になっていた。

損保ジャパンの基幹システムを刷新し、ビジネス変化に迅速に順応できるようにすることが SOMPO-MIRAI プロジェクトの目的だ。この目的達成のため、損保ジャパンは日立製作所とのジョイントベンチャーという形で SOMPO システムイノベーションズを設立し、プロジェクトを推進してきた。

SOMPO-MIRAI プロジェクトは主に3つのフェーズに分かれる。まず第1期に、全ての保険サービスに共通する部分と傷害保険を支えるシステムの稼働を開始した。2021年8月現在は、第2期として損保ジャパンの中核である自動車保険システムの開発が進行中だ。第3期で火災保険システムの開発を計画しており、「2025年の崖」を迎える頃までに主要種目の移行作業の完了を目指す。

複数ベンダーが携わる SOMPO-MIRAI プロジェクトは、約2000～3000人の開発者が携わる「超」大規模な体制だ。その中で浮上した課題が「ソースコードの品質を一定水準に均質化する施策」だった。

これだけの規模になると、開発者の技術レベルはまちまちになる。使用するプログラミング言語が「COBOL」から「Java」に変わり、開発の自由度は増した。「それと引き換えに、成果物であるソースコードの品質がばらばらになってしまうのではないかという懸念がありました」と、SOMPO システムイノベーションズの加藤 修氏（基盤本部現行基盤チーム チームリーダー）は語る。

コーディング規約やチェックリストを整備して共有するといった運用面の工夫は凝らしたが、それだけでは限界がある。そこで加藤氏は「ソースコードの静的解析ツールを使って不具合や脆弱（ぜいじやく）性を開発者に通知して、コードの品質を均質化する仕組み」が必要だと考えた（図）。

解析結果の優秀さとCIツールとの相性を評価してCoverityを選定

そこで SOMPO システムイノベーションズは、静的解析ツールの導入に向けて複数の製品を比較した。商用ツールを取り寄せてオープンソースのJavaプログラムを実際に解析することで検証し、選定したのが日本シノプシスの「Coverity」だった。

SOMPO システムイノベーションズの検証では、セキュリティ問題も含めた重大な不具合の検出率は、Coverity が他の製品に比べて1.6倍高い結果となった。「不具合ではないものを不具合と判断する誤検知が多過ぎる静的解析ツールは、開発者にとって使い

づらい。Coverity はそうした誤検知が他製品の約半分でした」と加藤氏は説明する。

もう一つのポイントは CI（継続的インテグレーション）ツールとの親和性が高かったことだ。開発対象が基幹システムであることから、SOMPO-MIRAI プロジェクト全体はウォーターフォール型の開発工程を採用している。一方でコーディングの速度と品質を向上させるため、工程の一部でアジャイル型の開発手法も取り入れている。このアジャイル型開発で利用しているのが、CI/CD（継続的インテグレーション／継続的デリバリー）ツール「Jenkins」だ。

SOMPO システムイノベーションズの黒澤直哉氏（基盤本部現行基盤チーム）は Coverity の活用方法についてこう語る。「Jenkins をトリガーに Coverity を呼び出してソースコードを解析させる、というプロセスを日次で実行する運用をしています。随時不具合を検出して修正する作業を継続しています」

他にも、コーディング規約の確認やプログラミング言語のバージョンアップへの追従といった特徴を加味し、Coverity が優れていると判断した。

「不慣れなJava」の取り扱いに関する懸念を解消

Coverity への投資に見合った効果は得られたのだろうか。Coverity を利用しない場合と利用する場合を比較すると、利用する場合は問題 1 件の改修にかかる時間の見積もりが大幅に短縮された。これは、Coverity は問題があることを指摘するだけでなく具体的な修正方法も提示するため、改修にかかる検討時間を短縮できるためだ。「導入した方が圧倒的にコスト面で有利になるという試算結果が出ました」と加藤氏は話す。

Java に不慣れな SOMPO システムイノベーションズでも、ソースコードの品質を定量的に評価できる点もメリットだという。

SOMPO システムイノベーションズの福田浩一氏（役員室 フェロー）は「Java に関する知識が不足していた部分もあり、パートナー企業の開発者が記述したソースコードをきちんと評価するための判断基準には不安がありました」と説明する。Coverity を使ってチェックすることで、問題に対する見解を把握できるようになった。「『この品質状態ならば大きな不具合がなく、本番環境での稼働に堪えられる』という安心感を得ることができています」と福田氏は述べる。

SOMPO システムイノベーションズは、開発工程の中で小まめに Coverity を利用することに加え、週次でレポートを作成して品質に対する統制を利かせるという工夫も取り入れている。ソースコード解析の結果を共有することで、ソースコードの品質はもちろん、セキュアコーディング（セキュリティを意識したコーディング）をはじめとするセキュア開発の意識を高める役にも立っているという。

開発者からするとソースコード解析という作業が 1 つ加わることになるが、この点について黒澤氏は「開発者は必要性を理解しており、利用が定着している」と感じているという。

「もし Coverity を導入していなかったら、多くの手戻りや品質問題が発生していたのではないかと思います」と黒澤氏は話す。多様なパートナー企業が参加し、チームも複数に分かれる

中で、いきなりそれぞれが作成したソースコードを用いて結合テストをすると欠陥数が膨大になる恐れがある。この点について黒澤氏は「大規模プロジェクトだからこそ、初期段階から Coverity を利用して短いサイクルで問題抽出を繰り返すことが大事」と考えているという。

「Coverity から出てきた検査結果を利用して、運用にも取り入れています。SOMPO-MIRAI プロジェクトの中で CI を回していく上で、Coverity は欠かせないツールだと感じています」（黒澤氏）

保守段階でもCoverityの活用を検討

SOMPO-MIRAI プロジェクトは 2021 年 8 月現在、ヤマ場である自動車保険関連のシステムの結合テスト段階に入り、開発の最終段階を迎えた。第 3 期の火災保険の業務要件定義も終盤に入っている。第 1 期で開発したシステムは保守局面となり、機能追加や改善の要望に応じている。

「『ソースコードの品質を一定レベルまで引き上げる』という Coverity の役割は、これまでの開発局面で果たしたと思います。これからは、保守局面の中でどう活用するかを検討し、運用体制をうまく構築したいと考えています」と福田氏は展望を語る。

セキュリティに対する要件が年々厳しくなっていることを受け、福田氏は Coverity のセキュリティチェッカー機能を使用して開発局面で脆弱性を洗い出すという使い方の重要性を挙げる。保守局面の開始後も、最新の脆弱性情報を随時反映して問題を検出する運用方法を日本シノプシスと共に検討したいという姿勢も見せる。

「DX 推進においてクラウドサービス利用やアジャイル型開発が不可欠だ」というのが SOMPO システムイノベーションズの考えだ。「ウォーターフォール型だけでなく、開発手法のバリエーションを増やし、Coverity で問題を一気に洗い出して修正するという新しい道を探そうと検討しています。その中で、Coverity がもたらす価値を活用したいと考えています」（福田氏）

静的解析ツールの導入により、開発者自身による品質向上と、チームリーダーのコードレビューの作業時間短縮を実現する
 ・開発用デバイスには、カバー範囲が狭いが、解析時間が比較的短いOSSツールを導入
 ・CIサーバには、カバー範囲が広いが、解析時間が長い有償ツールを導入

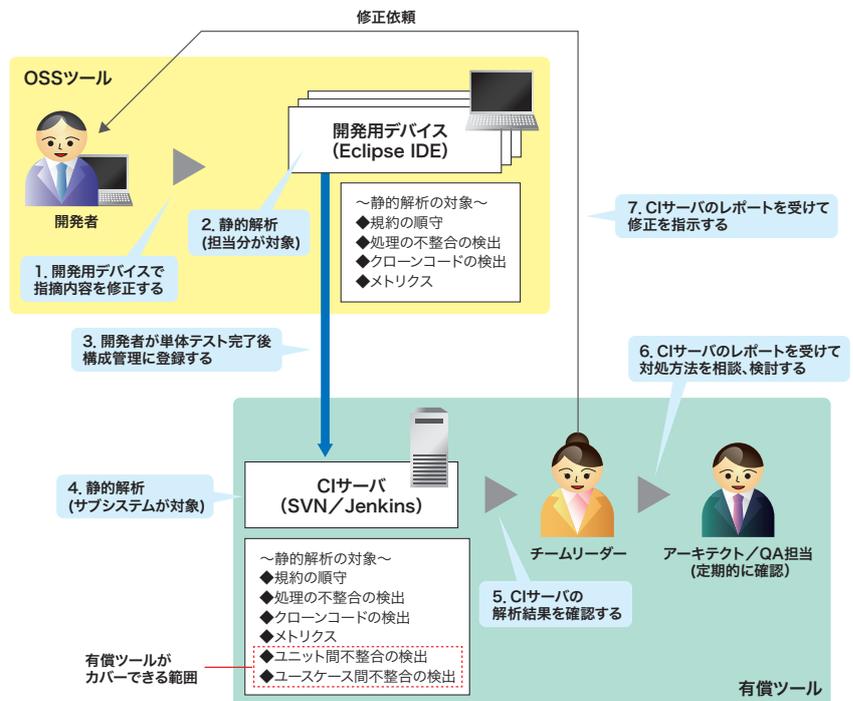


図 SOMPOシステムイノベーションズが検討した静的解析ツールの活用方法 (出典:SOMPOシステムイノベーションズ資料)