開発現場の「掟」を代行する Java コンパイラ Irenka の開発
―ソフトウェア開発ノウハウの蓄積と自動化―

１．背景

ソフトウェア開発プロジェクトは、その成熟につれて現場に様々な掟が開発される。このような掟は、その履行を担保としたプロダクトの品質向上やプロジェクトの効率化などの恩恵を与えることがある反面、掟の習得や履行、およびそれらの管理には一定のコストが発生するという問題がある。これらのコストにより、ソフトウェア開発プロジェクトにおける掟は十分にその効果が発揮されていない場面があると考え、掟を自動的に代行し、このようなコストを低減する仕組みは今後重要になっていくと考える。

２．目的

本プロジェクトで開発するソフトウェア Irenka は、そのような掟の履行をコンパイルフェーズにおいて自動的に代行し、プロジェクトメンバーの掟の履行を自動的に管理するような機能を有し、プロダクトの品質、プロジェクトの効率を高く保ちながら掟に関するコストを軽減させるようなソフトウェアを開発することを目標とした。また、それらの掟を電子的に蓄積、配布可能にし、ソフトウェア開発に必要な掟をプロジェクト全体で共有することで、プロジェクト全体の品質向上を図る。

３．開発の内容

本プロジェクトでは Irenka に Java コンパイラとしての機能を搭載させ、Java 言語を利用したソフトウェア開発プロジェクトに対する適用を目標とした。そして、掟の代行を実現するためにプログラム内の特定の構造を検出するためのクエリ言語、およびその処理系を開発し、また検出した構造に対して掟を適用するためのプログラム構造変換のための機能を開発した。そして、これらの機能を Java の統合開発環境の一つである Eclipse に組み込み、Irenka Studio という名前の Eclipse Plug-in として公開している。

Eclipse の標準の設定では、ソースプログラムを変更し、保存した際に自動的にそれらをコンパイルする。この際、Irenka が自動的に起動され、そこに組み込まれたユーザ定義の掟も自動的に実行される。図 1 は Irenka に登録したユーザ定義の掟によって警告を Eclipse 上に表示させている例で、プロジェクトの問題を表示する Eclipse 上のビューに「デバッグログ出力は ‘if (LOG.isLoggable(...))’ で囲むべきです」というメッセージを表示している。これは警告を表示する例であるが、コンパイル時にソースプログラムそのものを変更するなど、より複雑な処理を行うことも可能である。

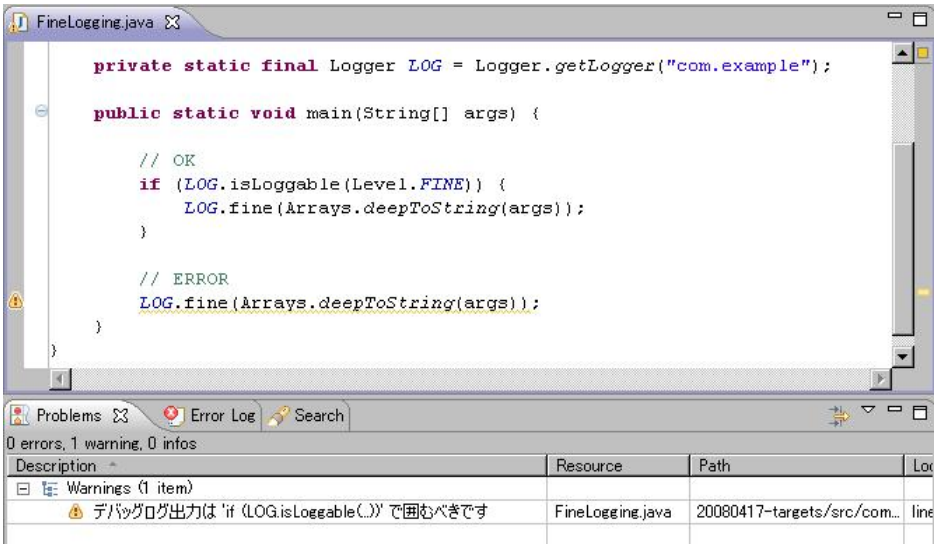

図 1.「掟」によって警告を Eclipse 上に表示させている例

Irenka および Irenka Studio は、次のような機能を有する。

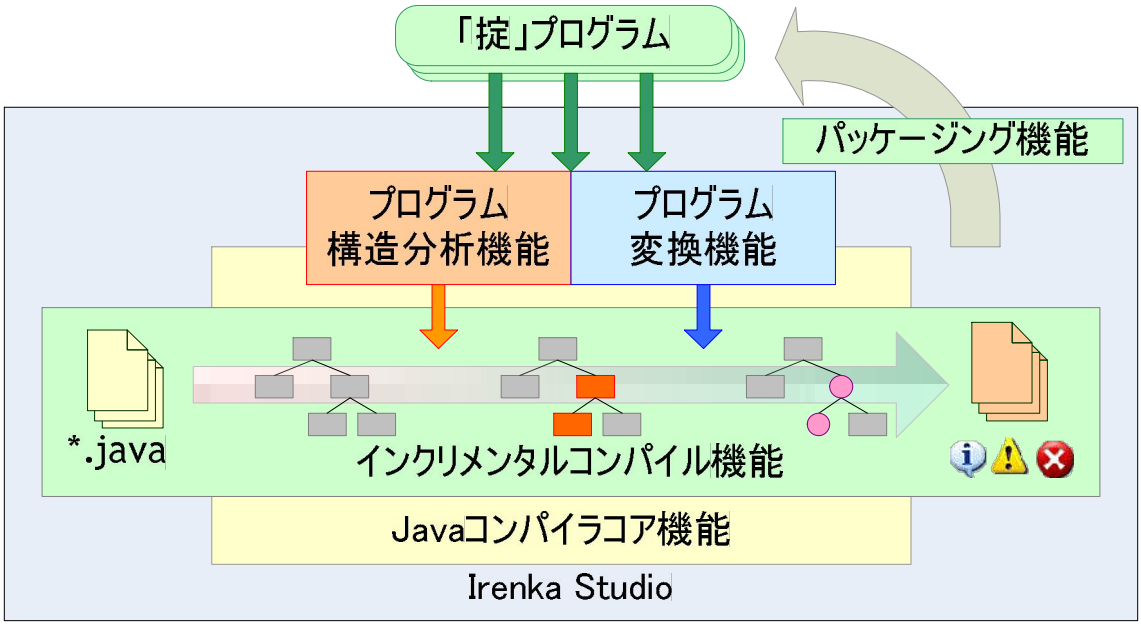

図 2. Irenka および Irenka Studio の機能

Java コンパイラコア機能

Irenka は、プログラミング言語 Java で記述されたテキストファイルを読み込み、Java プログラムとしての意味を表現した構造 - Irenka Document Object Model (Irenka DOM)を生成する機能を有する。Irenka DOM は通常のコンパイラが利用するソースプログラムの中間表現のようなもので、プログラム中に存在するシンボルが全て解決された状態で提供される。

プログラム構造分析機能

プログラム構造分析機能は競合プロダクトにあまり実装されていない Irenka の特徴的な機能で、提供する Irenka Search Query というクエリ言語を用いて特定のプログラム片を検索することができる。多くの競合プロダクトは特定のプログラム片

を検出する場合、プロダクトに組み込まれた定型的な検索機構を用いるか、またはユーザがプログラムのオブジェクトモデル（Irenka DOM のようなもの)をその構造をたどりながら分析する必要があった。前者では分析機能が限定的になってしまい、後者ではコンパイラの知識を必要とするため、いずれも利用範囲が限られていた。この問題を解決するため、Irenka では特定の構造がコンパイル対象に含まれているか問い合わせるためのクエリ言語を提供し、複雑な構造を単純でわかりやすい表現で検出することが可能である。

プログラム変換機能

プログラム変換機能は、Irenka の Java コンパイラコア機能によって生成された Irenka DOM をユーザが変更した際に、その変更を適切に書き戻すための機能である。同 DOM は簡素な方法でその構造を変更することができ、Irenka に組み込まれたデコンパイル機能によりDOMへの変更が反映されたソースプログラムを生成することが可能である。

インクリメンタルコンパイル機能

Irenka Studio には、プロジェクトをインクリメンタルにビルドする機能を有する Irenka Builder が搭載されている。これは、Eclipse 上でソースプログラムを変更した際に、その変更の範囲に絞ってIrenka DOMを再構成するための機能である。この機能によって、プロジェクト全体を毎回ビルドすることなく、短い時間での「掟」の履行を実現している。

パッケージング機能

Irenka Studio で作成した「掟」は、パッケージング機能を利用して配布及び再利用することが可能である。これにより、プロジェクトメンバー全員で同一の掟を利用したり、過去のプロジェクトで利用した掟を再利用したりすることができる。

開発環境および実行環境を下記に示す。ただしこれは確認の取れた環境のみを列挙しており、Eclipse 3.3 系 Java 開発環境の多くで動作するように開発を行っている。

プログラミング言語　　: Java (J2SE 5.0, JavaSE 6.0)
実行環境　　: Eclipse 3.3.1 Classic, Eclipse 3.3.2 Classic
オペレーティングシステム　　: Windows XP SP2

４．従来の技術（または機能）との相違

Irenka および Irenka Studio では、ソフトウェア開発における「掟」を容易に実現できるように注意深く設計を行った。まず、プログラムが「掟」を守っているか否かを検査するために、プログラム構造分析機能を有する。この機能は Irenka Search Query というクエリ言語を通して利用することができ、特定のプログラム構造が持つ特徴を同言語によって記述し、その特徴を有するプログラム片をコンパイル対象から検出することが可能である。このようにプログラムの監査が可能であるプロダクトとして Checkstyle というものが既に存在するが、これによって同等の機能を実現するにはコンパイラの知識や、プログラム構造を分析するための複雑な手順が必要である。

同様に、Irenka ではプログラムが「掟」を守っていない場合に、そのプログラムを自動的に改変する事が可能である。この場合、プログラム構造分析機能によって検

出された該当箇所をそのまま変更すればよい。同機能によって検出されたプログラム片は、Java プログラムのオブジェクトモデルである Irenka DOM として表現される。Irenka DOM は直接操作することができ、それによって加えられた変更はプログラム変更機能によって、ソースプログラム等に反映させることができる。Irenka DOM は Java プログラムの意味構造を基として構築されており、その変更によるプログラムの改変は、テキストを基とした変更に比べて最終的なソースプログラムの安全性が高いと考えられる。たとえば、スクリプト言語等を利用すればテキストの置換によるプログラムの改修は可能であるが、それがコンパイル可能かどうかは実際にコンパイラを利用しなければ検証が難しい。

５．期待される効果

開発したソフトウェア一式により、多くの利用者がいる Java 統合開発環境 Eclipse 上で「掟」を Hack として作成し、それらを同環境上に適用することができる。これらの「掟」は Eclipse 上でコンパイルが実行されるたびに併せて実行され、「掟」を守らないプログラムを発見するたびに警告を発したり、自動的に改修を行ったりすることが可能となった。また、これらの機能は Eclipse 上から透過的に利用され、利用者は Irenka の存在を意識せずに「掟」をプログラムに対して適用することができる。これらの「掟」は Irenka により機械化されるため、その学習、実施、監査に掛かるコストを軽減することが可能となった。これにより、開発者はプロジェクト中に守るべき掟を破ってしまった際にも Irenka により自動的に通知され、また Irenka の自動的なサポートにより実施のコストも低減させることができる。この Irenka によって守られた成果物は品質に一定の担保がある状態であり、品質保証の担当者も Irenka で検出できないようなより高度な品質の管理に注力することができるようになると考えられる。

６．普及（または活用）の見通し

Irenka および Irenka Studio は OSS として開発されており、本プロジェクトの採択期間中に協力者が 10 人以上加わった。彼らによって Irenka Studio の周辺機能の一部が開発されており、今後もそのような場を通じて Irenka に関する情報を発信し、より魅力的なプロダクトに磨き上げていくつもりである。

また、事業展開の方向性として、Irenka を用いた開発コンサルティング、および開発に関する教育コンサルティングを考えている。Irenka の最大の目標は、「掟」の機械化と物質化、つまり自動的に「掟」を履行し、かつそれを蓄積可能な成果とすることであった。これらの成果を利用し、開発の監査を低コストで行ったり、教育用の「掟」を準備することにより、Java プログラミングに関する教育を一定水準まで機械化したりすることができると考える。

７．開発者名（所属）

荒川 傑(株式会社グルージェント 開発部)

(参考)開発者URL

Irenka Project Site: http://irenka.ashikunep.org/