

スプレッドシートを利用した業務アプリケーションの実装 Implementation of Business Application using the Spreadsheet

小池 賢一
Koike Kenichi

菅野 幹人†
Kanno Mikihiro

1. はじめに

業務アプリケーションを開発する手法としてスプレッドシート上にビジネスロジックを記述しておき、その内容をビジネスルールエンジンが参照しながら業務処理を行う方式が広く採用されている。この方式により、ビジネスロジックをアプリケーションから外出して、業務管理者が直接業務のルールを修正することができるようになり、業務アプリケーションの開発期間の短縮や、開発コストと運用コストを低減が実現できるという利点がある。

一方、業務の種類に応じてスプレッドシートに記述する内容は大きく異なるため、スプレッドシートのフォーマットもそれに合わせて変更すべきである。しかし、従来のビジネスルールエンジンの多くでは、スプレッドシートのフォーマットは固定されており、そのフォーマットに合わせてビジネスルールを記述する必要があった。

そこで、本稿ではフォーマットを固定しない方式を採用する。ただし、その場合、業務向け（お金に関係する計算が必要な場合）と、それ以外を区別して計算方法を切り替えるという新たな課題が発生する。本稿では両方の計算に対応するために最小限の工数でビジネスルールエンジンを改修する方法を提案する。今回は、広く利用されている Microsoft Excel のファイル形式に対応しているという理由から Apache の POI ライブラリ (<http://poi.apache.org/>) を利用した。以下の第 2 節で本稿の提案するシステムの構成とスプレッドシートの属性情報を示す。第 3 節で本稿の解決する課題を示す。第 4 節で Apache POI ライブラリの拡張方法を示す。

2. システム構成とスプレッドシートの内容

以下に本稿で提案するシステムの構成を示す。

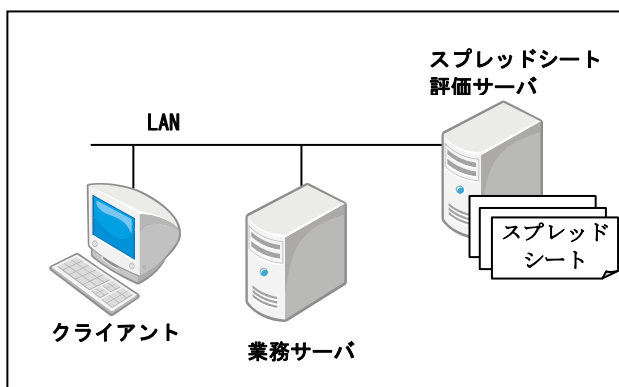


図1 システム構成

本システムでは、業務全体の処理は「業務サーバ」で行う。業務処理のうちスプレッドシートに記述されているロジックを利用する場合は、「スプレッドシート評価サーバ」に評価を要求する。「スプレッドシート評価サーバ」は、指定されたスプレッドシートの評価を行いその結果を返す Web サービスを提供する。「スプレッドシート評価サーバ」には業務に対応した複数のスプレッドシートが保存される。

スプレッドシートを評価するライブラリとして Apache の POI を利用する。POI ライブラリは、Microsoft Excel のスプレッドシートを扱うことができ、セルに記述された文字や数値や計算式を評価することができる。また、他のセルの参照やシートに記述された表から値を検索する VLOOKUP などの関数を利用することもできる。

また、スプレッドシートが業務向けか一般向けかを区別するために、スプレッドシート本体とは別に、属性情報を定義する。属性情報はスプレッドシートと紐付けて管理する。以下に属性情報の一覧を示す。

表 1 属性情報一覧

属性名	内容
シート ID	シートを特定するための一意の ID
有効期限開始日	スプレッドシートの有効期限の開始日
有効期限終了日	スプレッドシートの有効期限の終了日
シート種別	「業務向け」/「一般向け」の種別
入力セル位置	入力セルの位置のリスト
出力セル位置	出力セルの位置のリスト

シート ID はスプレッドシートを識別するために使用する。有効期限はスプレッドシートの有効期限を表し、スプレッドシートに複数の世代が存在する場合に、どのスプレッドシートを適用すべきかを日付で判断するために使用する。シート種別属性は、「業務向け」の場合は計算式の処理の際に厳密な計算を行い、「一般向け」の場合は通常の浮動少数点数の演算を行う。入力セルの位置属性は、スプレッドシートの評価を実行する前に、スプレッドシートにパラメータを設定する必要がある場合に、Web サービスのリクエストから取得したパラメータを設定するセルを指定するために使用する。出力セル位置属性は、Web サービスのレスポンスとして返す値の取得先のセルを指定する。以下に属性情報の例を示す。

表 2 属性情報の例

属性名	値
シート ID	A0001
有効期限開始日	2012/01/01
有効期限終了日	2012/12/31
シート種別	業務向け
入力セル位置	A2,A5
出力セル位置	B2,B3

†三菱電機情報技術総合研究所, Information Technology R&D Center, Mitsubishi Electric Corporation

3. スプレッドシートの評価における課題

Microsoft Excel や Apache の POI ライブラリでは、スプレッドシートに記述された計算式を正確に計算することができないという問題がある。実際に Excel のファイルに以下の計算式を入力すると誤差が発生する。

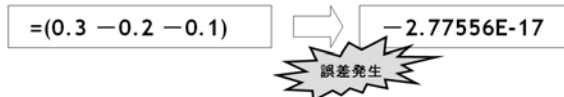


図2 誤差が発生する例

この問題は、0.1 を 2 進数で表現するときに、循環小数となり正確な値を保持できないことから発生する。同様の問題は Apache の POI ライブラリでも発生する。お金の計算では計算結果を最後に円の単位にするために四捨五入が行われる。このときの境界である 0.5 が誤差のために 0.5 より小さい値になると、計算結果が誤った値になることがある。以下に Microsoft Excel での例を示す。

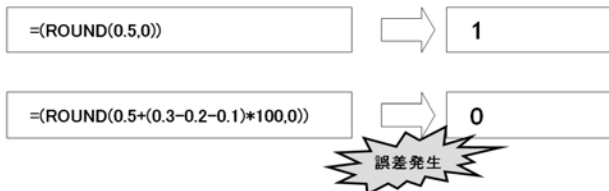


図3 誤差が計算結果に影響する例

上の計算式では、0.5 を四捨五入して正しい結果が出力される。一方、下の計算式では 0.5 から誤差の値の 100 倍が加えられているため、0.5 よりも僅かに小さい値となり、四捨五入した結果が「0」になる。お金を扱う業務ではこの誤差の発生は許されない。一方、科学技術計算では四捨五入を行うことはないので問題とはならない。

4. Apache POI ライブラリの拡張方法

Apache の POI ライブラリは、Java で記述された Excel ファイルや Word ファイルを扱うためのライブラリであり、計算式の処理は Java のプリミティブ型である double 型を使用して実行される。double 型では内部表現が 2 進数となるため、3 節で示したものと同一誤差が発生する。一般に、Java でお金の正確な計算を行う場合は、10 進数を扱うことができる java.math.BigDecimal クラスが使用される。そこで、Apache の POI ライブラリを拡張して、業務向けのスプレッドシートの評価を行う場合は、double 型で計算するのではなく、BigDecimal クラスで計算すればよい。一方、計算速度の点では、double 型で計算する方が速いという理由から「一般向け」のシートでは従来通り double 型で計算した方が有利である。そこで、「業務向け」と「一般向け」を区別して、計算方法を切り変える方式が考えられる。しかし、double 型と BigDecimal クラスでは扱える数値の範囲において以下の様な違いがある。

表3 double と BigDecimal の違い

	非数(NaN)への対応	無限への対応
double 型	○	○
BigDecimal クラス	×	×

Apache の POI ライブラリは、double 型が非数や無限を扱えることを前提として設計されているために、double 型を BigDecimal クラスに置き換えることが難しい。そこで、新たに double 型と同様に NaN や無限を扱える「十進数演算機能」を開発した。この機能は内部で double 型と BigDecimal クラスの両方を利用して実現する。以下にその処理の手順を示す。

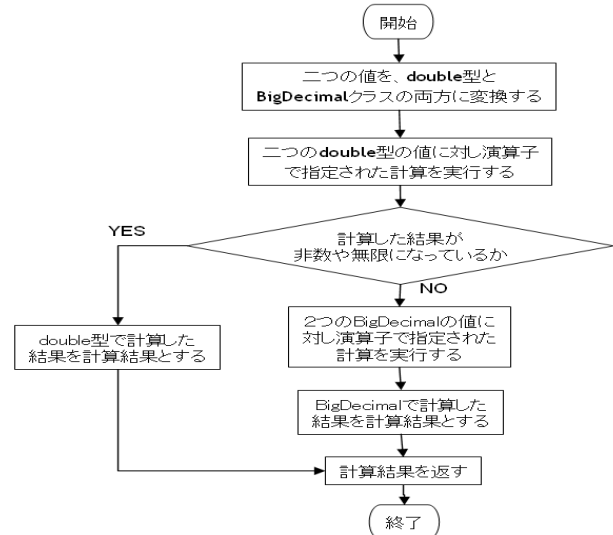


図4 十進数演算機能の処理手順

最初に、計算対象の値を文字列型から double 型と BigDecimal クラスのインスタンスの両方に変換する。次に、double 型で計算を実行して結果が非数や無限になっているか判定する。非数や無限になっている場合はそれを計算結果として返す。非数や無限になっていない場合は、BigDecimal クラスのインスタンスで計算を実行してそれを計算結果として返す。

以上のようにすることで、計算結果が非数や無限になる場合はその値を返し、それ以外の場合は正確な値を計算して返すことができる。

この「十進数演算機能」を利用することで、Apache の POI ライブラリを拡張して、業務向けと一般向けの計算の両方に対応することが可能となった。

5. おわりに

本稿では厳密な計算を必要とする業務にも利用できるスプレッドシートを利用した業務アプリケーションの実装方法について論じた。従来の方式との違いは、スプレッドシートのフォーマットを固定しないものとし、厳密な計算が必要な場合と、それ以外を区別して計算方法を切り替える方式を採用した点である。さらに、新たに「十進数演算機能」を開発することで既存のライブラリの改修範囲を最小限に留めながら両方の計算に対応できることを示した。今後はこの方式を実システムに適用してその有効性の評価を行う。

参考文献

- [1]川口正高, 塩尻綾子, 浅見可津志, 原田雅史, 佐藤啓紀:オープン環境のシステム構築を高品質・短納期で実現する Web システム開発標準 “MIWESTA”三菱電機技報 81(7), 489-492, 2007-07