

ビジネスルール管理システム (BRMS) の今後の活用

日本アクチュアリー会 IT 研究会第2グループ

【担当委員】

| | |
|-------|-----------|
| 岡村 隆之 | プルデンシャル生命 |
| 立部 博之 | 日本興亜損害 |

【メンバー】

| | |
|--------|-------------|
| 山口 真弘 | 富国生命 |
| 鈴木 千尋 | 富国生命 |
| 千野 祥平 | 富国生命 |
| 後藤 雅樹 | 野村総合研究所 |
| 菅原 慎太郎 | 共栄火災海上 |
| 井上 裕美子 | カーディフ損害 |
| 水野 一仁 | ソニー生命 |
| 市川 弘美 | アクサダイレクト生命 |
| 岩石 晃 | 三井生命 |
| 佐藤 功 | ニッセイ情報 |
| 吉田 祐章 | 太陽生命 |
| 奥田 樹 | 太陽生命 |
| 中野 雄一 | メットライフアリコ生命 |
| 川村 一央 | 損害保険ジャパン |
| 重松 利弥 | 大同生命 |
| 平田 将也 | 全共連 |

序文

近年、市場や法制度、技術趨勢（すうせい）の変化に伴い、生命保険／損害保険システムは頻繁な改修を要請されている。特に業務オペレーションの背後にある「ビジネスルール」の変化は顕著であり、保険システムにとってもビジネスルールの変化を正確に、かつ迅速に反映する仕組みの構築が課題となってきた。

こうしたなか、ビジネスルールを他システムから分離し管理するビジネスルール管理システム(以下 **BRMS**) が注目を集めている。当研究グループでは、保険業界を取巻く環境の変化に対し保険システムが迅速に対応するため、**BRMS** がどのように活用できるか、また導入に際しどのような課題があるかを検討する。

| | | |
|-----|---|------|
| 1章 | はじめに | -49- |
| 1.1 | 保険業界を取巻く環境の変化..... | -49- |
| 1.2 | 保険業務の課題について | -49- |
| 1.3 | 保険システムの課題について..... | -50- |
| 2章 | BRMS とは | -52- |
| 2.1 | ビジネスルールとは | -52- |
| 2.2 | BRMS の特徴と効果 | -54- |
| 2.3 | まとめ | -56- |
| 3章 | BRMS 導入に際しての検討事項..... | -57- |
| 3.1 | BRMS 導入による開発体制の変化 | -57- |
| 3.2 | BRMS に実装するルール作成の手順..... | -59- |
| 3.3 | ホスト（メインフレーム）環境における BRMS 利用手段 | -61- |
| 3.4 | BRMS を用いたバッチ処理の実行 | -63- |
| 3.5 | 複雑化した既存システムへの BRMS 導入 | -64- |
| 3.6 | まとめ | -65- |
| 4章 | 保険業界における BRMS の現状と今後の展望..... | -66- |
| 4.1 | 各保険会社における BRMS の導入状況 | -66- |
| 4.2 | 各 BRMS 製品の機能 | -67- |
| 4.3 | BRMS に適合する保険業務領域..... | -67- |
| 4.4 | BRMS の保険業務への導入事例..... | -68- |
| 4.5 | 導入の課題..... | -69- |
| 4.6 | 新たな業務領域への活用 | -70- |
| 4.7 | まとめ | -71- |
| 5章 | おわりに | -72- |
| | 参考文献..... | -73- |
| | 別紙「 BRMS のデモ開発検証」 | |

1章 はじめに

BRMS について論じるにあたり、まず保険業界と保険システムの歴史に触れ、時代の変化とともに発生した保険システムの課題について整理する。

1.1 保険業界を取巻く環境の変化

保険業界を取巻く環境の変化について、システム導入時期を基準とし、以下の3つのフェーズに分類する。

1) 事務プロセスの自動化を目的とした業務アプリケーションの導入

1950年代以降、業務におけるコンピュータシステムの活用が普及したことに伴い、保険業界においても汎用コンピュータを用いた業務アプリケーションを導入し始めた。導入の目的は、正確に、かつ効率的に伝票を計上することであった。この際、非常に多くの複雑な業務内容が一枚岩のシステムとして構成され、このシステムは現在に至るまで業務の重要な基盤として機能している。

2) 業務のオンライン化、双方向化

1970年代以降、業務のオンライン化に伴い増加した業務量に対応するため、システムには更なる業務の自動化が求められた。また、この頃より顧客やリスクなどをはじめとした情報の管理・活用もシステムの重要な要素の一つとなった。その結果、システムには様々な条件をもとにした判断処理（ルール）が取り込まれるようになり、急速に拡張されていった。

3) 保険業務やシステムの複雑化

1990年代から現代にかけて、保険業界を取巻く環境は劇的な変化を遂げることとなる。1990年代後半、日米保険協議の決着を受けて保険業法が全面改定され、各社の保険料に差が表れはじめた。2000年代には銀行での保険販売が開始され（窓販解禁）、更に通信販売やインターネットを用いた保険販売が開始されるなど、保険契約の販売チャネルが増加した。

これらの保険業界を取巻く環境の変化に伴い、各社、商品販売・管理システムの開発や販売チャネル増加によるシステム構築、利用するデバイスの増加によるシステム対応など、保険業界の環境変化に対応できるよう変化した。

また、保険商品を購入する顧客のニーズも時代とともに大きく変化しており、保険商品並びに保険商品を販売するための営業活動はより複雑化している。そのため、保険システムもまた複雑化の一途を辿っている。

1.2 保険業務の課題について

昨今の保険業界を取巻く環境の劇的な変化は、保険業務に対して大きな課題を突き付けた。その最たる例として、2005年頃に表面化した「保険金不払い問題」が挙げられる。

保険金不払い問題の一因として、複雑化した保険商品に対して支払い体制（あるいはシステム）が追いつくことができなかつたことが考えられている。保険商品や保険料の自由化は、競合他社との差別化を目的とした商品戦略活動を促進する契機となったが、保険商品やサービスが複雑化したことで、保険業務自体の複雑化も招くこ

ととなった。

このようなことから、保険業界が直面している課題として本稿では以下の2点を取り上げる（表1）。

表1：保険業界の課題

| | |
|-----|--|
| 課題1 | 環境の変化・顧客ニーズの変化に対して、保険業務は柔軟かつ俊敏に対応できなければならない |
| 課題2 | 保険業務が複雑であっても、作業の手順は平準化され、得られる結果の信頼性を常に担保できなければならない |

今後も保険業界を取巻く環境は劇的に変化し続けていくことが予想されるため、保険システムにはこれらの課題を解決することが強く求められている。

1.3 保険システムの課題について

他業種と比べ早期からシステム化を進めた保険業界においては、度重なるシステム改修を経た結果、様々な課題を抱えることとなった。そして表1で示した保険業界の課題解決を目的としたシステム開発も容易に行うことができない環境になってしまった。表2は、システム開発を行う上で考えられる保険システムの課題を示している。

表2：保険システムの課題

| | |
|-----|-------------------------|
| 課題1 | ユーザー部門、IT部門双方の認識が合わせにくい |
| 課題2 | 要件変更への俊敏な対応が困難である |
| 課題3 | システム改修の都度、システムが肥大化する |

1) ユーザー部門、IT部門双方の認識が合わせにくい

従来のシステム開発は、①ユーザー部門が自然言語を用いてIT部門に要件を伝える（要件定義）、②IT部門が要件を実現するための処理内容を詳細化する（各種設計）、③IT部門が詳細化した処理内容を機械言語に変換し実装する（プログラミング）、という手順を踏むことが一般的である。

保険業務は他業種の業務と比較して複雑性が高いこともあり、IT部門がユーザー部門の要件を正確に理解し、それを詳細化していくためには高度な専門知識が必要となる。専門知識が不足していた場合には、ユーザーが求めている要件が正確にIT部門へ伝わらない、また、IT部門の誤った要件解釈のもと詳細化が進められるといった弊害が発生しうる。

また、プログラミングの段階では、自然言語で詳細化された処理内容を機械言語へと変換し実装するため、変換の誤りによってユーザー部門が意図しない処理が行われる可能性も孕んでいる。

以上のように、ユーザー部門とIT部門双方が、要件に対する共通の認識を持つにあたり、それを阻害する要因が存在する。このことから、ユーザー部門とIT部門双方で認識を合わせやすい環境や手法の構築が求められている。

2) 要件変更への俊敏な対応が困難である

プログラムコード上に様々な条件分岐命令（ルール）が直接記述されているシステムは、現在においても数多く存在している。長年に渡る運用の中でシステムに埋め込まれてきたこれらルールは相当量存在し

ており、そのすべてを瞬時に把握することは非常に困難である。そのため、ルール変更の都度、ある程度の要員や対応期間を確保した上で既存システムに与える影響調査を行う必要がある。

また、現在保険業界においてシステム開発の主流となっているウォーターフォール型の開発モデルでは、下流工程に進むほどユーザーとの認識あわせや要件変更の取り込みが難しくなる。そのため、ユーザーとの認識相違や急な要件変更が発生した場合、対応に多くの時間を要することになり、ユーザーの要望に迅速に応えられない。このことから要件変更への俊敏な対応がシステム開発に求められている。

3) システム改修の都度、システムが肥大化する

保険業務がシステム化されてから今日に至るまでの長きにわたり、システムに実装された様々なルールの改修が行われてきた。度重なる改修を経て複雑化したシステムにおいては、ルールに関わる信頼性の担保が非常に困難となる。

更に保険業界におけるシステム開発では、業界を取り巻く環境変化に加え、ユーザーからの要望や法令対応、サーバー保守といった運用保守を実施しており、複数案件を同時並行で対応していることが多い。従来の開発手法ではシステム全体の最適化や効率化より案件ごとの対応を優先しがちなため、常にシステムが肥大化していく傾向となる。

このことから、保険システムの肥大化の抑制が求められている。

2章 BRMS とは

1章で示した保険システムの課題に対する解決策として、本論文ではBRMSの導入を提案する。提案の前段として、本章では「ビジネスルール」の定義と分類を示したうえで、BRMSの特徴およびBRMS導入のメリットを整理する。

2.1 ビジネスルールとは

BRMSは、「Business Rule Management System（ビジネスルール管理システム）」の略語であり、その特徴を一言で表すのであれば「業務アプリケーションからビジネスルールを切り離し独立させて一元管理することができるシステム」である。

BRMSについて触れる前に、BRMSで管理すべき対象となる「ビジネスルール」とはどのようなものか、以下に整理していく。

<1> ビジネスルールの定義と分類

ビジネスルールとは「業務上、下されるある判断（意思決定）の根拠となる基準・規程のことであり、具体的に実行可能なレベルのもの」とであると定義される。端的に表現すれば、業務上で「もし〇〇ならば（条件）、××する（アクション）」といった形式で表現されるものは、その大半がビジネスルールであるといえる。

保険業界におけるビジネスルールの一例を示せば、表3に示すようなものが考えられる。

表 3：ビジネスルールの例

| 条件 | アクション |
|--------------------|---------|
| もし被保険者年齢が45歳以上であれば | 通知を送る |
| もし被保険者年齢が45歳未満であれば | 通知を送らない |

一言でビジネスルールといっても、その分類は様々である。保険毎日新聞に掲載されたビジネスルールの分類^①を参考に、ユーザー部門において理解すべきビジネスルールの分類およびその具体例を示す（表4）。

実務の多くは、これら無数の制約やガイドラインをベースに推論もしくは計算ルールを実行し、そこから導き出される情報をもとに後続のアクションを実行する、ということを連続的に、あるいは繰り返しながら遂行される。保険会社における実務においては、他の業種と比較した場合、ビジネスルールが多岐にわたり、かつ業務の中核を担っているⁱⁱ。

ⁱ 「業務」の範囲を外れる判断——戦略レベルの判断（「国債利回りが低いため、金融系商品を投入する」）や、システムレベルの判断（「DB接続に失敗したので、ログを出す」）——はビジネスルールとは呼べない。また「実行可能」でないルール——「安全第一」など——も取扱い対象外となる^①。

ⁱⁱ 「典型的な保険会社は300のプロセスと1600ものアクティビティ」を持っておりその「70%はデシジョンを必要としている」という^②。（デシジョンの背後にあるルールが、本稿で扱うビジネスルールである）

表 4：ビジネスルールの分類

| ルールの分類 | ルールの種類 | 例 |
|----------------------|---------------------------------------|---|
| ビジネスの構造や 振る舞いの定義 | 制約(Constraint rule) | 被保険者の年齢は 20 歳以上、 60 歳以下でなければならない |
| | ガイドライン(Guideline rules) | 保険申込者は収入を 持つべきである |
| ある事実に基づく 新たな情報の生成 | 計算(Computation rules) | △△保険の場合、 契約者貸付金の上限は 解約返戻金の 80%とする |
| | 推論(Inference rules) | 被保険者の評点合計が 100 点未満なら標準下体とする |
| 具体的なアクションの導出 | アクションイネーブラー (Action enabler rules) | 被保険者が標準下体である場合 謝絶の通知を行う |

< 2 > ビジネスルールの所在

保険会社においてビジネスルールがいかに重要な要素であるかについては前述のとおりであるが、ビジネスルールはどこで定義されているのであろうか。保険会社におけるビジネスルールの所在と表現方法を整理すると、表 5 のようになる。

表 5：ビジネスルールの所在と表現方法

| ビジネスルールの所在 | |
|--------------------------|------------------------------------|
| 法令等 | (保険業法、保険法、金融庁の監督指針、生命保険協会ガイドライン 等) |
| 基礎書類 | (普通保険約款、事業方法書、算出方法書 等) |
| ↓ 表現方法 | |
| 社内規定・事務処理要領 | |
| システム設計書、業務フロー・システムフロー | |
| 各業務の所管部署で作成される非公式なドキュメント | |
| 有識者のナレッジ (非体系化) | |

このように、ビジネスルールが存在するエリアは非常に多岐に渡っており、ビジネスルールは保険業務を遂行する上で非常に重要な要素である。

保険業界においてはビジネスルールの追加・削除・更新といった変更の頻度が多く、特に各業務の所管部署で作成される非公式なドキュメントは最新の情報で管理できていない場合が多いのではないかと考えられる。ビジ

ネスルールの管理が適切にできていない状態は、ビジネスルールの変更に伴うシステム対応の要望を受けてから対応完了までにかかる費用や時間に、大きな影響を与えることになる。

変化の激しい保険業界においては、ビジネスルールの変更に対して、早く、安く、正確に対応することを求められていることから、ビジネスルールを管理するためのシステム（BRMS）が近年注目を集めていると考えられる。

2.2 BRMSの特徴と効果

前節ではBRMSにて管理するビジネスルールを整理した。ここからはBRMSの機能やその特徴について整理していくことにする。

BRMSとは、「業務上、下されるある判断（意思決定）の背後にあるビジネスルールを、他のシステム要素と分離して一元管理するためのシステム」であると定義される。ビジネスルールを一元管理することで、前節に述べたビジネスルールの所在を明確化することができ、適切に管理することが容易になる。BRMSを用いてビジネスルールを一元管理することには、現在の保険業務の在り方やユーザー部門とIT部門の在り方に大きな変革を与えうる可能性がある、私たちは考えている。

< 1 > BRMSの特徴

BRMSが持つ特徴を整理すると、主として表6に示す3点が挙げられる。

表6：BRMSの特徴

| | |
|-----|---|
| 特徴1 | ユーザー部門とIT部門双方が理解できる「共通言語」を用いてビジネスルールを定義可能 |
| 特徴2 | ルールの容易な変更が可能 |
| 特徴3 | ルールを他要素から分離して定義可能 |

1) ユーザー部門とIT部門双方が理解できる「共通言語」を用いてビジネスルールを定義可能

BRMSを用いると、ビジネスルールを、ユーザー部門とIT部門双方にとって直感的に理解しやすい表現形式のままシステムに配備することが可能となる。

具体的な表現形式は製品により様々であるが、代表的な例として「自然言語」「決定表」「決定木」によるビジネスルールの表現例を図1に示す。いずれの表現形式を用いるかは、ルールのもととなるドキュメントの形式やルールの複雑性などから判断すべきであるが、いずれの形式でも、条件・アクションともに保険業務で日常的に利用される用語によって記述されていることが分かる。

なお、自然言語形式で使用可能な文法は製品によって異なる。例えば、ドメイン特化言語(DSL:Domain Specific Language)ⁱⁱⁱといった形式がある。

ⁱⁱⁱ コンピュータに指示を与える言語で、より対象ドメイン（システム／業務領域）に合った形での記述が可能なもの。C言語などの汎用プログラミング言語と対比される。DBクエリ用のSQL言語もDSLの一例である。

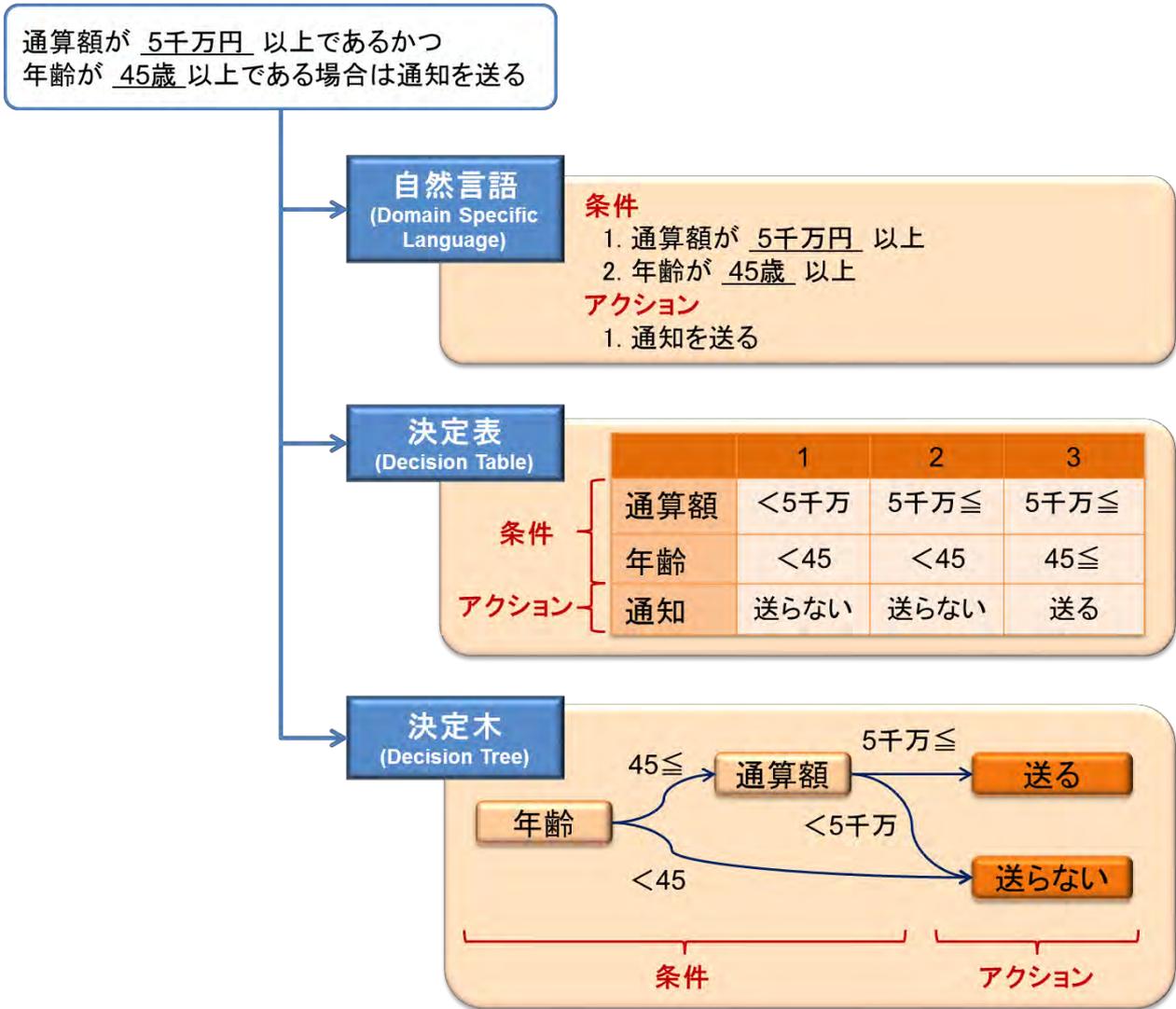


図 1：「自然言語」「決定表」「決定木」によるビジネスルールの表現例

2) ルールの容易な変更が可能

ビジネスルールに変更を加える場合、使い慣れた Excel などを使用できるため、BRMS を用いると迅速な対応が可能となる。

また、BRMS 製品の多くにはルールに対してのテスト機能が備わっているため、ビジネスルールの変更のみであれば、BRMS 上で修正から動作確認（テスト）までを一貫して行うことが可能である。

3) ルールを他要素から分離して定義可能

現行システムにおいては、ビジネスルールに該当する部分とその他の処理部分が渾然一体となっているケースが多いと考えられる。BRMS を用いれば、これらの要素からビジネスルールに該当する部分のみを分離して一元管理することが可能となるため、システムのスリム化が実現できる。

< 2 > **BRMS の効果**

前項の BRMS の特徴は、1 章にて提示した「保険システムの課題」とリンクさせることで、その解決策としての効果を見出すことができる（表 7）。私たちはこれらの 3 つの効果から、BRMS の活用は、保険システムが抱えている課題に有効な解決手段として期待できると推察する。

表 7：BRMS の特徴と効果

| BRMS の特徴 | BRMS の効果 |
|---|-----------------------------------|
| (1) ユーザー部門と IT 部門双方が理解できる 「共通言語」を用いてビジネスルールを定義可能 | (1) ユーザー部門と IT 部門の認識を 合わせやすくなる |
| (2) ルールの容易な変更が可能 | (2) 要件変更に対する俊敏な対応が可能となる |
| (3) ルールを他要素から分離して定義可能 | (3) システムの肥大化を抑制する |

1) ユーザー部門と IT 部門の認識を合わせやすくなる

ユーザー部門と IT 部門双方が要件に対する共通の認識を持つにあたり、それを阻害する要因として、認識齟齬、誤変換といったリスクが存在していることは前述のとおりである。

BRMS を導入することによるビジネスルールの見える化によって、①ユーザー部門と IT 部門間のコミュニケーション円滑化を促し、認識齟齬のリスクを低減する効果、②プログラムコードを直接記述することなく、定義したルールからプログラムコードを自動生成することで、誤変換のリスクを低減する効果、という二重の効果が期待できる。

2) 要件変更に対する俊敏な対応が可能となる

従来のシステム開発においては、終盤の工程に差し掛かる程に、要件変更が発生した場合の手戻りリスク（手戻り発生時の影響）は当然大きくなってしまふ。手戻りのリスクの大きさ次第では、リリースのタイミングに影響を与えかねない事態にも発展しうる。

BRMS を導入した場合、ビジネスルールの変更のみに関わる要件変更であれば、BRMS 上での作業のみで修正からテストまでを実現できるため、比較的短期間で対応が可能となる。例えば、頻繁に発生する料率改定のような改修は、柔軟かつ低リスクのうちに対応が可能となるだろう。

BRMS がユーザー部門と IT 部門の認識を合わせやすくする効果を持つことは前述のとおりであるが、上流工程の段階における認識誤りを減少させることで、手戻り発生の確率を下げる効果も期待できる。

3) システムの肥大化を抑制する

1 章 3 節で述べたとおり、従来のシステムにおいては、ビジネスルールの度重なる改修がシステムの肥大化に大きな影響を与えていると推察できる。

また、異なるサブシステム間で同一のチェックを行う場合も、共通の資産を用いるのではなく、独自で開発したチェック機能を用いている場合も少なくないため、システムの肥大化につながる。

BRMS を用いてビジネスルールを一元管理すれば、システム肥大化を抑制する効果を得られる。

2.3 まとめ

本章では BRMS に関する用語の定義を確認し、保険システムへの導入効果を示した。

2 章 1 節ではまずビジネスルールの一般的な定義を示したのち、分類を概観し、保険業界において、多様な情報源からビジネスルールが取得されることを示した。

2 章 2 節では、ビジネスルールを管理・実行するシステムである BRMS が提供する機能を示し、その一般的な導入効果——①共通言語によるルール定義、②ルールの容易な変更、③ルールの他要素からの分離——により 1 章で示した保険システム特有の課題がどのように解決されるかを示した。

3章 BRMS 導入に際しての検討事項

本章では、実際に BRMS を導入するにあたり、IT 部門として検討すべき事項について考察する。

具体的には、システム開発体制やルール定義プロセスといった BRMS 導入時の開発環境の一例と、ホストシステムや複雑化したシステムへの BRMS 適用方法、バッチ処理プロセスといった、BRMS の稼働環境について述べる。

3.1 BRMS 導入による開発体制の変化

現在の保険システム開発では、ユーザー部門のメンバーが開発に加わるのはユーザーテストなど、開発の終盤であることが多く、IT 部門との認識相違があったとしても、その発見が遅れてしまっている。

BRMS はユーザー部門にも分かりやすいインターフェースが用意されていることが特徴であるが、現在の保険システム開発では、ユーザー部門が検証に参加するのが開発の終盤であることから、製品に備わっている機能を十分に活用できるとは言いがたい。

そこで、BRMS を活用する方策の 1 つとして、近年システム開発の世界で注目されている「スクラム開発」の導入を提言したい。

< 1 > スクラム開発とは

スクラム開発とは、チームを組み一丸となって開発を進めていく、プロジェクト運営のフレームワークの 1 つである^{iv)}。スクラム開発は、アジャイル^{iv)}など反復型の開発に適用しやすい手法であり、短期間で試行錯誤しながら開発を進めていくことが特徴である。また、メンバー構成、開発工程は以下の形式とするのが一般的である。

1) メンバー構成

スクラム開発では、各メンバーが以下の役割を担う。

- ・開発チーム (=担当者)
⇒要求分析、設計、コーディング、ドキュメント作成など、ソフトウェアを作るためのすべての作業を行う。数人でチームを組むことが多い。
- ・プロダクトオーナー (=承認者、意思決定者)
⇒作成されたドキュメントやソフトウェアについての責任者。開発チームを活用して、ドキュメントやソフトウェアが生み出す価値を最大化する責任がある。
- ・スクラムマスター (=プロジェクトマネージャー)
⇒スクラムのルールや成果物、進め方をプロダクトオーナーや開発チームに理解させ、効果的な実践を促し、スクラムの外にいる人からの妨害や割り込みからプロダクトオーナーや開発チームを守る。

^{iv)}要求仕様の変更などに対して、機敏かつ柔軟に対応するためのソフトウェア開発手法のことをいう。アジャイルでは、仕様や設計の変更があることを前提に開発を進めていき、徐々にすり合わせや検証を重ねていくというアプローチをとる^[4]。

2) 開発工程

スクラム開発では、「スプリント」と呼ばれるソフトウェア開発工程を用いる。スプリントは、開発、まとめ、レビュー、調整を繰り返す工程である。スプリントでは、開発、まとめ、レビュー、調整の各プロセスの順序は問わない。

スプリント期間中、チームは毎日スクラム会議を開催する。スクラム会議は決まった時間に行い、前回の会議以降に実施したこと、課題、今後の予定について、スクラムマスターが質問する。会話はこれらの質問への応答に限定され、議論が必要な場合は別の打ち合わせを開催する。

スプリントの後にはスプリントレビューが行われる。スプリントレビューにはプロダクトオーナーも参加する。スプリントとスプリントレビューは、機能や品質が十分と判断されるまで繰り返し行う。

< 2 > BRMS を用いたスクラム開発のプロセス

BRMS を用いた開発プロセスにスクラム開発の考え方を適用することで、ウォーターフォール型のシステム開発で課題となっていた、IT 部門とユーザー部門の橋渡し（仕様理解と要件理解の効率化）が可能になる。

本節で提案しているスクラム開発では、IT 部門とユーザー部の双方のメンバーがチームを組んで開発を行うことが特徴である。また多くの BRMS 製品には、システム開発に不慣れなメンバーでも容易に開発可能な、決定表などのインターフェースが用意されている。このことから、BRMS の導入とスクラム開発の適用によって、ユーザー部門と IT 部門が同じチームとして、要件定義・ルール設計・実装・テスト・検証の各プロセスを対応することが可能になる。

このように、上流工程の段階で IT 部門のメンバーとユーザー部門のメンバーがチームを組み、ビジネスルールの実装・検証を進めていくことで、ユーザー部門のメンバーはシステム化した場合のイメージをつかむことができる。

< 3 > スクラム開発導入による効果

図 2 は、BRMS を用いたシステム開発において、スクラム開発を導入する効果の一例である。スクラム開発を適用してビジネスルールの実装・検証を進めていくことで、従来は開発工程終盤のユーザーテストから検証に参加していたユーザー部門が初期段階から開発に参加できる。また、ユーザー部門のメンバーが設計初期段階から開発に参加することで、上流工程でルール開発（設計～テスト）を行うことができる。このことから、開発工程終盤で発覚していたバグを初期段階で除去できる可能性が高くなるため、下流工程で手戻りが発生する可能性が減少する。

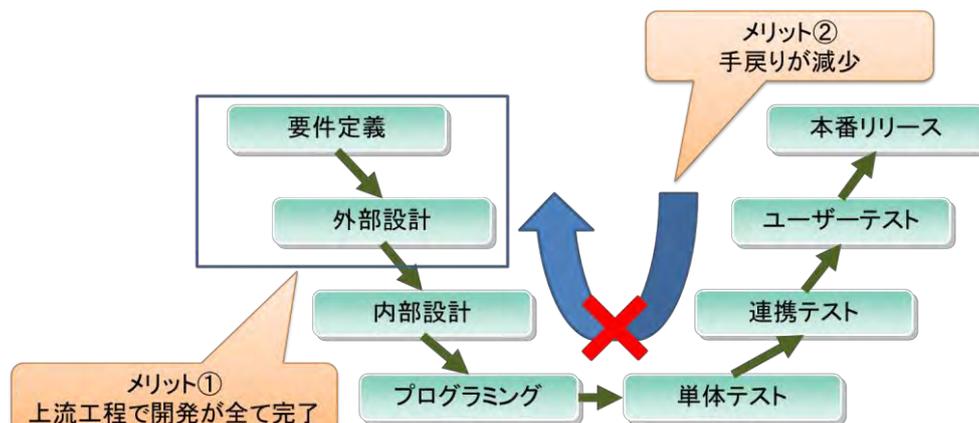


図 2：スクラム開発導入のメリット

3.2 BRMS に実装するルール作成の手順

本節では、BRMS の導入において重要となるルール実装までの手順を説明する。

図 3 は、BRMS を用いたルール開発プロセスの一例を示している。

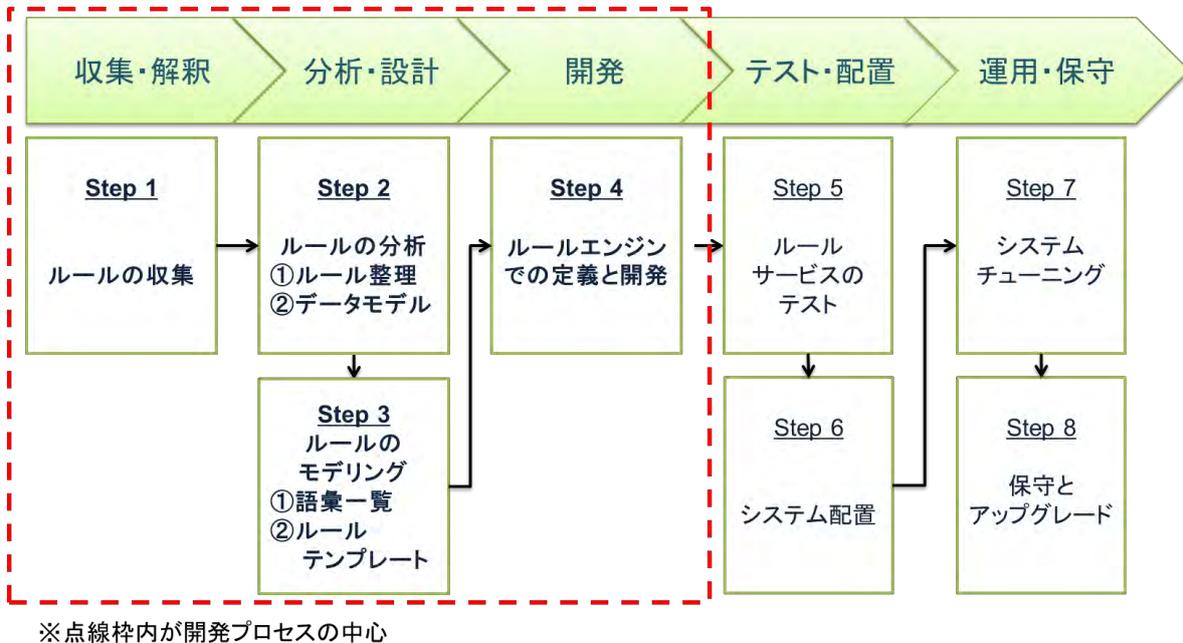


図 3 : ルール開発プロセス例

各 Step において行う具体的な作業は、次のとおりである。

< 1 > Step1 : ルールの収集

ルール開発における最初の作業はビジネスルールの収集となる。

ルールの収集元は開発するアプリケーションの種類や案件内容によって異なるが、大きく分けて以下の 4 つに分類できる。

1) 約款、事業方法書、運用マニュアルなどの資料

約款、事業方法書、運用マニュアルなどの資料には、「もし〇〇ならば (条件)、××する (アクション)」といったビジネスルールが多く盛り込まれている。例えば、新商品を BRMS に搭載する場合、ルールの抽出元は約款や事業方法書になる。

2) 業務プロセス

業務がシステム化されていない場合は、業務プロセスそのものをビジネスルールとして定義し、システムに取り込む必要がある。

3) 部門エキスパートの知見

医務査定など、ユーザー部門の担当者が自身の知識をもとにオペレーションを行っているケースでは、ビジネスルールがドキュメント化されていないことが多い。そのため、エンドユーザーからヒアリングを行い、ビジネスルールを抽象化する必要がある。

4) 現行システムのアプリケーションコード

BRMS により現行システムのリプレースを行う場合は、現行システムのアプリケーションコードよりルールを収集する。

< 2 > Step2 : ルールの分析

ルールの分析では「ルールの整理」のほかに「データモデルの作成」を行う。

ルールの整理では重複するルールや使われていないルールを削除し、欠落しているルールを追加する。その後、BRMS でパラメータとして使う語彙の設計のために、業務プロセス図、画面仕様書、帳票、ユースケースなどを分析し、整理したルールからデータモデルを作成する (図 4)。

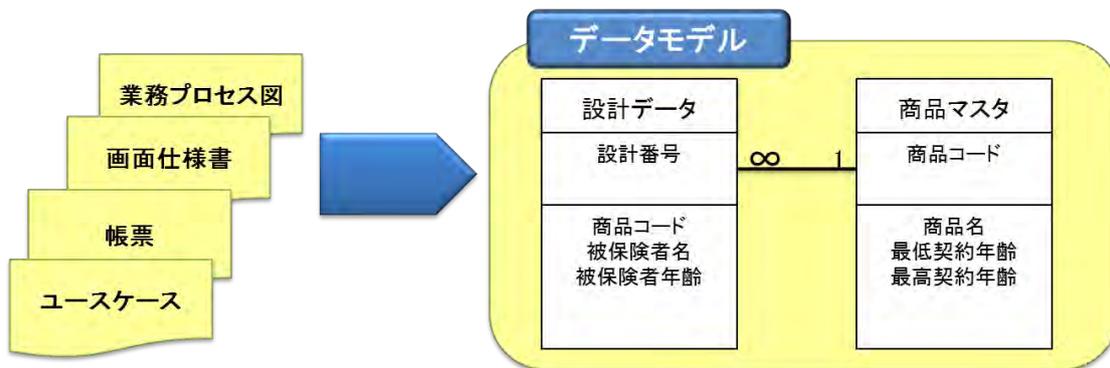


図 4 : データモデルの作成イメージ

< 3 > Step3 : ルールのモデリング

ルールのモデリングでは、Step2 で整理したデータモデルから「語彙一覧」と「ルールテンプレート」を作成する (図 5)。

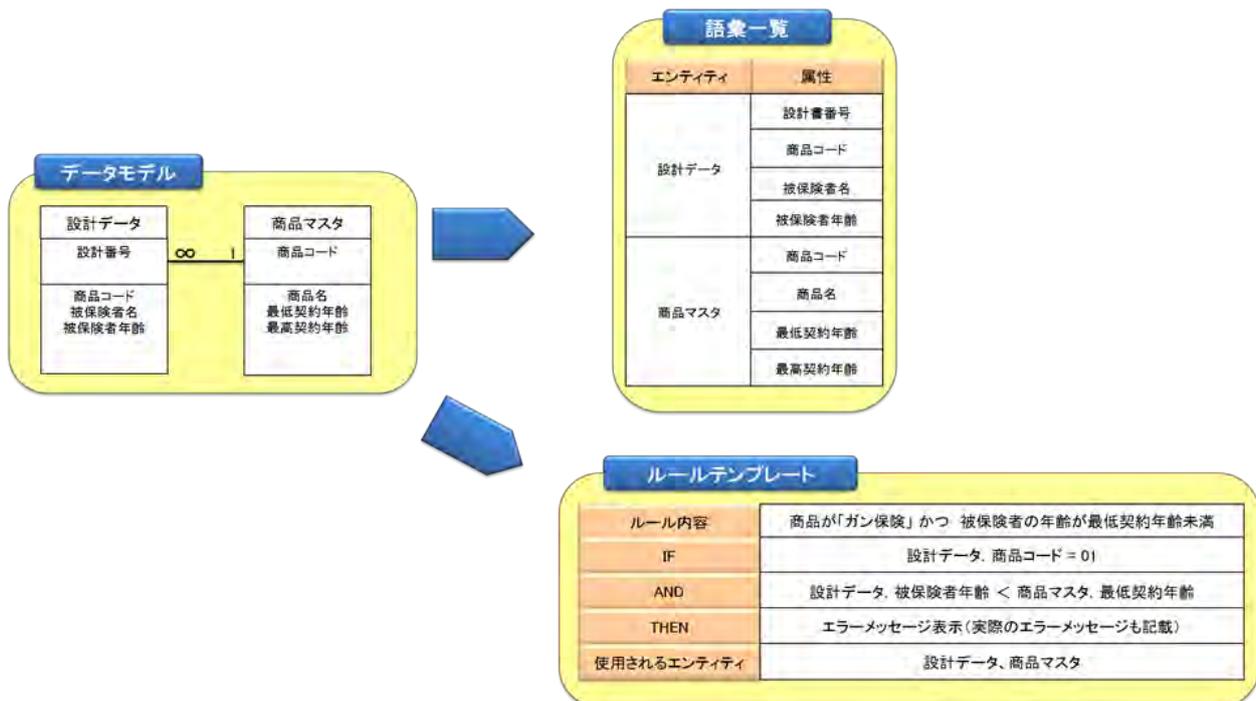


図 5 : 語彙一覧とルールテンプレートの作成イメージ

ルールテンプレートは、ルールを記載するフォーマットである。各担当者がばらばらなフォーマットを使用すると、システム設計書の可読性や保守性が低下するため、あらかじめ決められた統一フォーマットを使用するのが望ましい。ルールテンプレートは、製品に依存するものではなく、各企業、各システム個別で作成する。ルールテンプレートを活用することで、あらかじめ決められたフォーマットに語彙を当てはめるだけでルール定義が可能になるため、一からルール作成を行うことなく、短期間でのシステム開発が可能になる。

定義したルールは、ユーザー部門が確認作業を行うことで、要件の漏れや誤りを設計時点で検出できるというメリットがある。ユーザー部門による確認作業やルールの保守性を考慮し、ルールテンプレートは決められたフォーマットで作成することが重要となる。

< 4 > Step4 : ルールの実装

ルールテンプレートと語彙一覧をもとに、ルールの実装を行う。

製品ごとにルールの実装方法は異なるが、各社ともにプログラムの知識がなくても実装可能なユーザーインターフェースが提供されている。代表的なものとしては、2章にて述べた「自然言語」「決定表」「決定木」がある。

3.3 ホスト（メインフレーム）環境における BRMS 利用手段

保険会社は業務の特性上、お客様の契約情報など大量のデータを長期間にわたり管理・処理する必要があり、現在も多くのシステムがホスト環境で稼働している。

そこで本節では、BRMS をより多くのシステムで活用するため、ホスト系システムにおける BRMS の利用手段を考察する。

< 1 > ホスト系システムにおける BRMS 導入の現状

オープン系システムに比べ、ホスト系システムは BRMS の導入に不向きといえる。そもそも ホスト系システム（COBOL 言語）で直接動作するメジャーな BRMS 製品がない状況である。このため、ホスト系システムから BRMS を呼び出す場合には、システム間の通信が発生する。

オープン系システムであれば、Java のアプリケーションから直接 BRMS を呼ぶことが可能であるが、ホスト系システムの場合、OS や筐体（きょうたい）を超えた通信が発生しオーバーヘッドが大きくなってしま（図 6）。

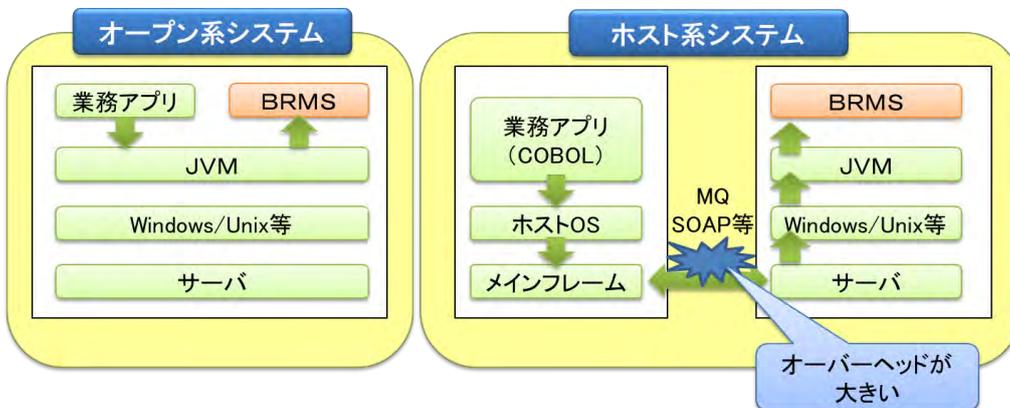


図 6 : オープン/ホストシステムへの BRMS 導入例

< 2 > ホスト系システムにおける導入方法

前項で説明したオーバーヘッドを抑える方法はいくつか存在する。ここでは具体的な導入方法を2つ紹介する。

1) ホスト OS 上での BRMS の稼働させる

IBM 製品である ODM(Operational Decision Management)^vは、ホスト上の JVM(Java Virtual Machine)で BRMS を稼働させることができる。同一の筐体・OS 上で業務アプリと BRMS を稼働することにより、OS、筐体間の通信を行うことがなくなり、オーバーヘッドを抑えることができる (図 7)。

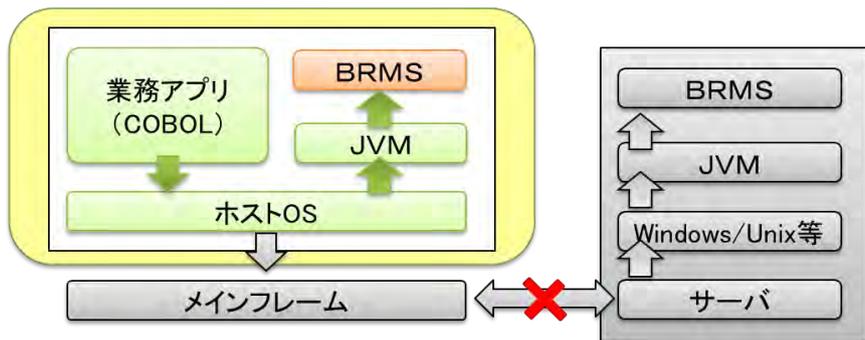


図 7: ホスト+非 COBOL にて BRMS を稼働させる例

2) ビジネスルールを COBOL として稼働させる

もうひとつの方法として、日本プログレス社の Corticon^{vi}を導入した国内損保の例がある。この例では、ビジネスルールを COBOL に変換することで、ホスト OS より業務アプリケーションと同様に COBOL として利用することを可能とした(図 8)。

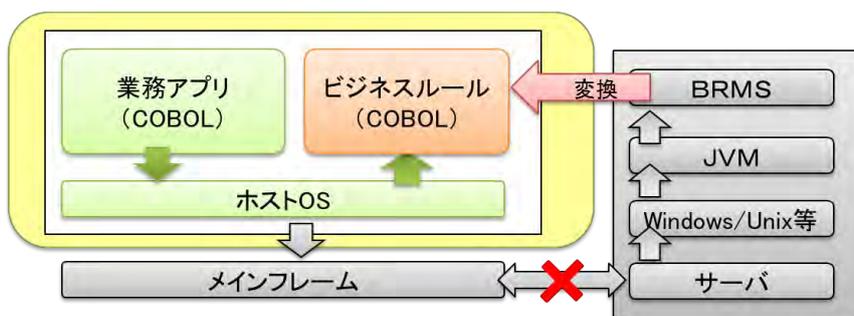


図 8: BRMS のルールを COBOL として稼働させる例

具体的な実装方法を図 9 に示す。ユーザー部門においてサーバーで稼働する BRMS を用いてルールを入力する。一方、業務アプリケーションはホスト系システム上にあるため、入力されたルールから COBOL ソースを自動生成 (変換) する機能を構築し、COBOL としてビジネスルールを利用した。この例はスクラッチ開発^{vii}であったが、変換は容易ではなく業務アプリケーションとルールのインターフェース、語彙を揃える必要があった。

^v製品の詳細説明は 4 章に記載する

^{vi}製品の詳細説明は 4 章に記載する

^{vii}特定のパッケージ製品のカスタマイズや機能追加などによらず、すべての要素を個別に最初から開発すること

そこで BRMS で使用する語彙ファイルを業務アプリケーションのレイアウト定義書 (COPY 句) から自動生成することで語彙ファイルの作成を容易にした。

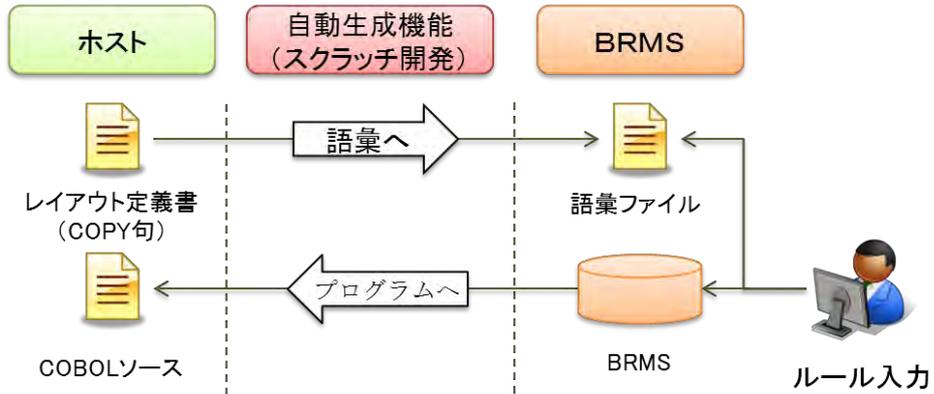


図 9 : COBOL 変換による実装例

このように、ホスト系システムにおける BRMS の導入は容易とはいえないが利用可能といえる。

3.4 BRMS を用いたバッチ処理の実行

保険会社のシステムではデータ毎に大量のビジネスルールによるチェックや計算を実施することが多くあるため、ここではバッチ処理時の BRMS 実行方法を考察する。

< 1 > バッチ処理のシステム構成例

業務システムによくあるシステム構成として、中心となる制御ロジックが入金データなどを読み込み、大量のチェックや計算を行ったうえでマスタや帳票などへの入出力を行うシステムがある (図 10)。このチェックや計算を BRMS にて処理する場合、処理を行う都度 BRMS を呼び出ししていると通信にかかるオーバーヘッドが非常に大きくなるといえる。

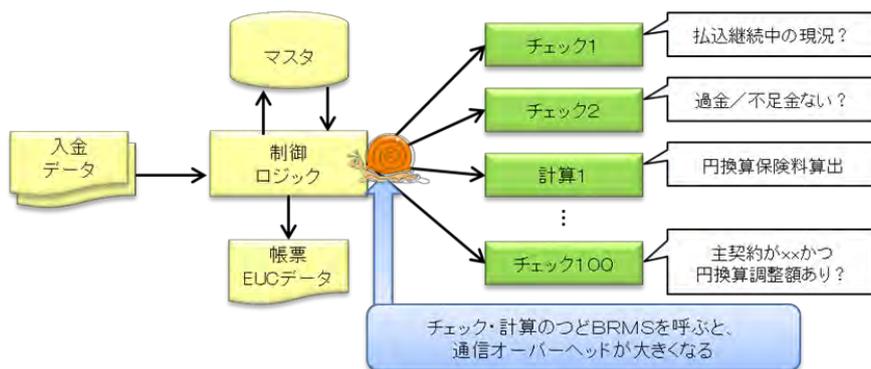


図 10 : バッチ処理のシステム構成例

< 2 > バッチ処理においてインターフェースに要する時間

具体的に「1 回の入力に 100 チェック、オンラインは毎秒 1 データ、バッチは毎秒 30 データ」と規模を想定した際、1 チェックに掛けられる時間は、それぞれ下記のとおりとなる。

- ・オンライン … 早いレスポンスが要求されるが一度に扱うデータも少ないため、10ms のオーダーで時間を与える
- ・バッチ … 大量データを扱うため、0.1ms と非常に厳しいオーダーを要求される

この規模では、BRMS 呼び出しのインターフェースに要する時間は無視できない。BRMS のチェック自体は(チューニングによるが) 要求されるオーダーに対応できるが、一度通信が発生すると、少なく見ても 1 桁大きな時間を奪われてしまうことになる。

< 3 > 処理遅延を防ぐシステム設計方法

処理遅延を防ぐため、インターフェース回数を減らすことで通信のオーバーヘッドを小さくすることを考える(図 11)。具体的な手順は次のとおりである。

- 1) 制御ロジックは、入力データをもとにチェックや計算に必要な情報をマスタから一括して収集する
- 2) 収集した情報をファイルにまとめて、データ転送、FTP などで BRMS サーバーに送信する
- 3) ファイルを受け取った BRMS サーバーでは、自前のバッチや BRMS のバッチモードでまとめてチェックを行う
- 4) 再度返送されたチェックや結果の結果をもとに、制御ロジックでマスタ編集や帳票出力を行う

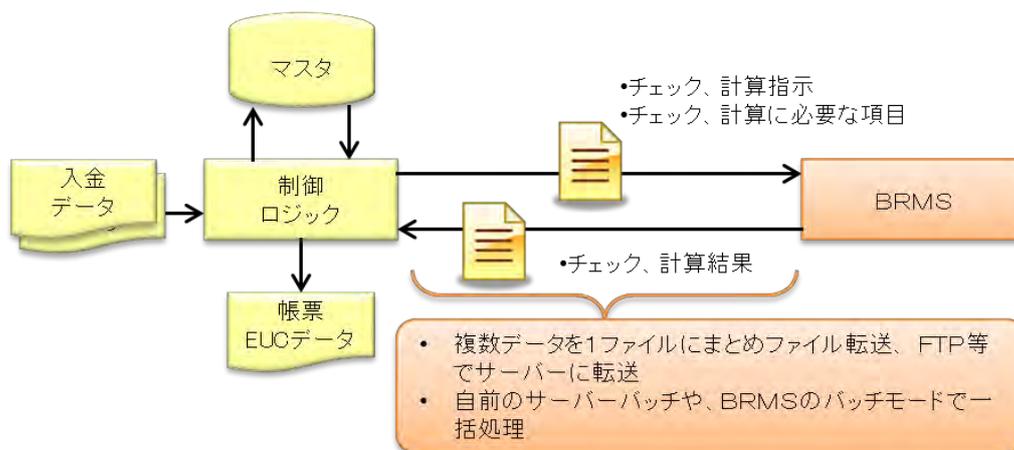


図 11：処理遅延を防ぐシステム設計方法

3.5 複雑化した既存システムへの BRMS 導入

最後に複雑化した既存システムへの BRMS 導入について考察する。

< 1 > 既存システムへの導入における問題

改修ごとの部分最適化を重ねるうちに、「プロセス」「機能・処理」と「ビジネスルール」が一体化し、容易に分離できない場合がある(図 12)。3 者の分割が明確であれば、業務のルール部分のみを切り出して BRMS に外部化することが容易だが、3 者が渾然一体となっている場合、そもそもの部分をビジネスルールとして外部化すべきか、ビジネスルールをいかに既存処理に影響を与えず外部化するか、といった困難に突き当たる。

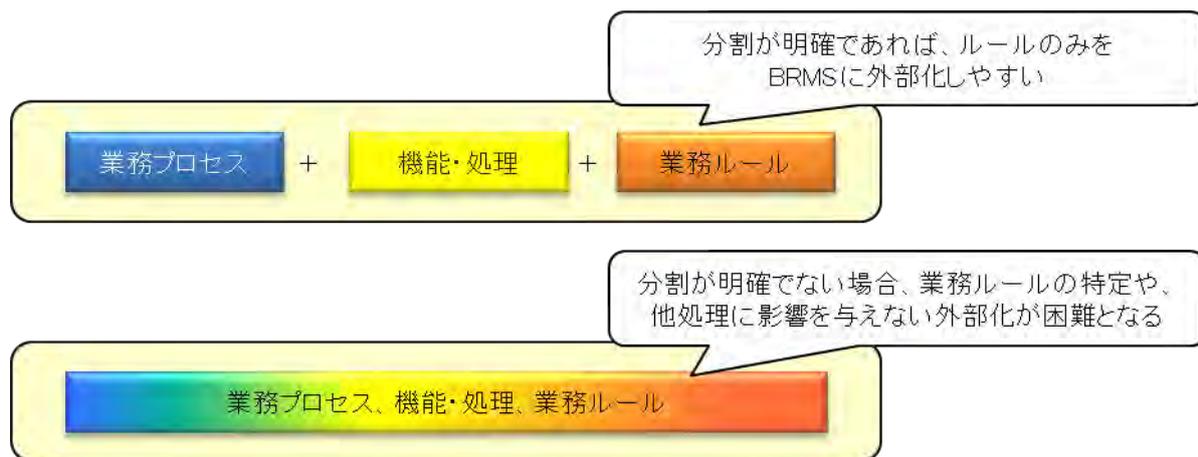


図 12：既存システムのイメージ

< 2 > 複雑化したシステムへの導入方法

複雑化したシステムに対する万能な解決策は無い。ベンダーレクチャーでは「日本の保険会社はホスト一枚岩でできており、ホスト内で部分最適化されている」といった意見があり、アンケートでは「既存アプリケーションの改修規模が大き過ぎた(全面組み換え)」などの意見があった。

BRMS を導入する場合は、ルール分離のコストと既存のままでの保守コストを比較し投資効果の高い部分から外部化していく方針とすることが望ましいと考える。

3.6 まとめ

本章では、実際に BRMS を導入するにあたり、IT 部門が検討すべき事項として、システム開発体制、ルール定義の作成方法、ホストシステムや複雑化したシステムへの BRMS 適用方法、バッチ処理プロセスといった BRMS の稼働環境を挙げ、その進め方や解決策を示した。

BRMS による効果を具体的に検証するため、当研究グループでは簡易的な「Web 保険申込システム」に BRMS を組み込んだデモ開発を行い、我々の考察を帰納的に証明することもできた。(詳細は別紙「考察：デモ開発検証」を参照)

現状、ホスト系システムへの導入は容易とはいえないが、ホスト OS で稼働する製品も出てきており、導入事例も増えてきていることから、今後は導入が増えていくものと考えられる。

4章 保険業界における BRMS の現状と今後の展望

BRMS は、業務プロセスやビジネスルールが複雑であり、また、変更頻度が多い業務領域に対して適用することで高い効果を発揮することができると考えられる。本章では、BRMS の導入状況について日本アクチュアリー会賛助会員企業に対して行ったアンケート結果をもとに、保険業界における BRMS の現状と導入における今後の展望について考察する。

4.1 各保険会社における BRMS の導入状況

日本アクチュアリー会賛助会員企業である保険会社約 40 社に対して行ったアンケートについて、『BRMS の導入・検討状況』の問いについての回答結果を図 13 に、また『導入済みの場合、導入している製品』の問いについての回答結果を図 14 に示す。

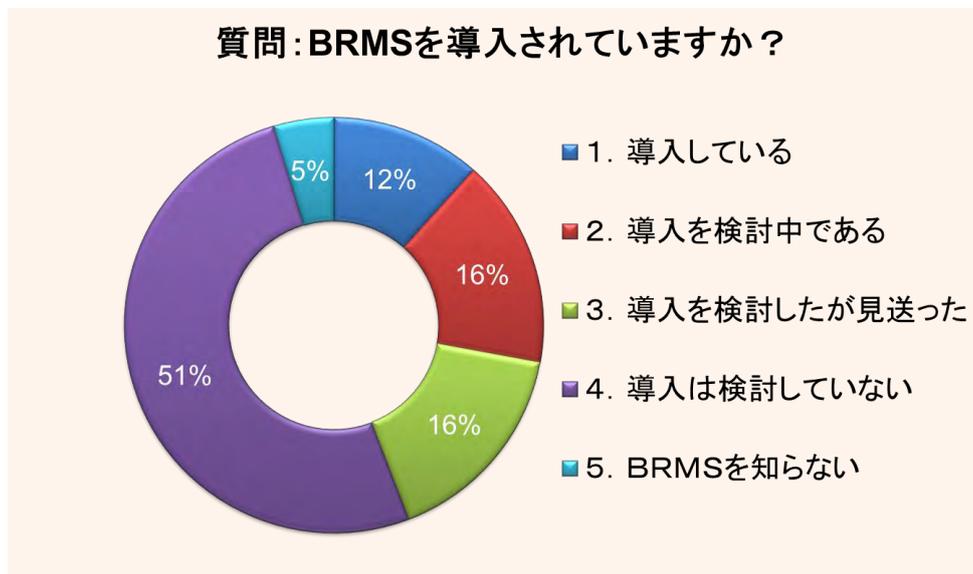


図 13: 保険会社の BRMS 導入状況

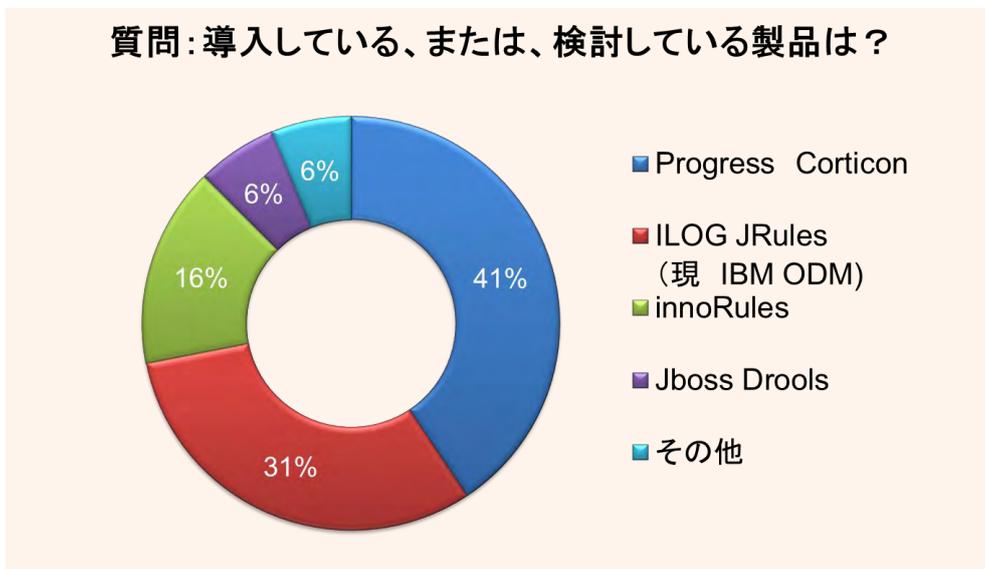


図 14: BRMS 導入企業の採用製品

図 13、図 14 より、日本アクチュアリー会賛助会員企業の約半数に及ぶ 44%が『BRMS について導入もしくは導入の検討を行った』と回答しており、『導入している、または、検討している製品』については IlogJRules、ProgressCorticon、JbossDrools、innoRules の 4 製品で全体の 90%以上を占めていた。

4.2 各 BRMS 製品の機能

上記で述べた BRMS 製品 4 種についてそれぞれの機能を纏めたものを表 8 に示す。

表 8：BRMS 製品の特徴

| 製品名 | 提供会社 | 特徴 | 主な適用領域 | ルール定義方式 |
|----------------------------|---|--|--|---|
| ILOG JRules (現 IBM ODM) | IBM | <ul style="list-style-type: none"> ・ルール作成方法が複数用意されている。そのため開発担当者、業務担当者それぞれのスキルにあったルールの作成が可能。 | <ul style="list-style-type: none"> ・支払 ・申込み(引き受け査定) ・保険募集の報酬 ・契約内容の変更 | <ul style="list-style-type: none"> ・デシジョンテーブル ・デシジョンツリー ・デシジョンフロー ・自然言語 |
| Progress Corticon | 海外: Progress Software 国内: アシスト | <ul style="list-style-type: none"> ・独自の演算子をJavaのClassとして開発可能。 ・100%コーディングレスでビジネスルールの変更/追加が展開可能。 ・ルールに過不足や矛盾性・曖昧性がないか自動検証し、最適化できる。 | <ul style="list-style-type: none"> ・支払 ・申込み(引き受け査定) ・コンプライアンス ・契約管理 ・商品設計 | <ul style="list-style-type: none"> ・デシジョンテーブル ・デシジョンフロー |
| Jboss BRMS | レッドハット | <ul style="list-style-type: none"> ・Excelでルール記述が可能。 ・BPM、CEPを包含した機能セットを持つ。 ・オープンソースソフトウェアのため、低価格(保守サービスに相当するサブスクリプション費用のみ) | <ul style="list-style-type: none"> ・支払 ・申込み(引き受け査定) ・補償 ・マーケティング ・顧客管理 | <ul style="list-style-type: none"> ・デシジョンテーブル ・デシジョンフロー ・ルール言語 |
| innoRules | アーネストビジネスソリューションズ | <ul style="list-style-type: none"> ・ルール作成方法が複数用意されている。 ・テストデータのインポート、テスト結果のエクスポートがExcel形式で可能。 ・テストケースの自動生成が可能。 ・標準関数が80種類、Javaでユーザ独自の関数も開発可能。 ・ルール処理関連図によるルール処理構造全体の可視化が可能。 | <ul style="list-style-type: none"> ・支払 ・申込み(引き受け査定) ・商品管理 ・保全 ・顧客管理 ・営業成績評価 (手当、手数料など) | <ul style="list-style-type: none"> ・デシジョンテーブル ・デシジョンフロー ・テーブル形式 |

表 8 より、各製品ともに支払・申込み(引受査定)業務に適用されていることがわかる。

4.3 BRMS に適合する保険業務領域

BRMS について導入済み、または導入を検討している企業について、導入対象の業務領域を集計した結果を図 15 に示す。



図 15 : BRMS の導入領域

BRMS を契約・加入の分野に導入、もしくは導入の検討を行った企業は 75% となっており、この領域において BRMS が高い効果を発揮すると期待されていることがわかる。これは、冒頭で述べた、業務プロセスやルールが複雑であり、また変更頻度が高い領域に対して適用することで高い効果を上げることができる、という考察とも一致している。また、『人による複雑な判断を要する業務、各種データを分析することを主としている業務（人事・資産運用・業績管理などがそれらの業務に合致すると考えられる）』については、BRMS の導入検討が行われておらず、導入効果が薄いと考えられているものと考察できる。

4.4 BRMS の保険業務への導入事例

BRMS を実際の保険業務へ導入した例として M 社の事例を挙げる。

< 1 > 導入箇所

M 社ではテレフォンオペレーター向けの保険設計書作成システムの新システムをスクラッチ開発する際、その新システムの一部機能に BRMS(Progress Corticon) を導入した。

このシステムは、テレビコマーシャルなどを契機とした問い合わせに対して、テレフォンオペレーターが会話をしながら保険設計をスムーズに行うことを目的としたシステムであり、以下の 2 つの機能に BRMS を導入している。

1) 保険設計書作成時の商品の規程チェック

保険期間、払込期間に応じた年齢や保険金額の規程チェックを BRMS にて行う。

2) 帳票の表示制御

保険設計書に印字する「しくみ図」の組み合わせを BRMS にて制御する。なお、定めたルールは Excel でマトリックス化して、そのまま BRMS へ反映する。

< 2 > 導入効果

BRMS を導入したシステム開発では、決定表を用いてルールの実装を行うことで、コーディングをせずに製造工程を行うことが可能となり、プログラミングの知識や経験が無いユーザーでもルールの作成が行えるようになった。また、BRMS のテスト機能を利用することで、ルールの動作確認が BRMS 上で可能である。さらに、ルールの論理的エラーを自動的に検出する機能が備わっている。これらの機能により作業の効率化を図れるようになり、M 社の新システムの開発においても、BRMS を導入せずにすべてスクラッチで開発した既存のシステムと比較して工数が 2~3 割削減された。

また、BRMS は品質の向上についての効果も見られ、M 社の新システムでは本稼動後 45 ヶ月で約 25 万件の保険設計書出力を行っているが、顧客影響を伴う障害の発生件数は 0 件である。

以上から、BRMS を用いてビジネスルールを実装し、システム開発を行うことで開発の効率化および品質向上の効果を期待できる。

4.5 導入の課題

一方で図 13 からは、BRMS の導入を検討したがものの、最終的に見送った企業が 16%あったことも分かる。

< 1 > 導入見送りの理由

見送りの理由としては以下のものが挙げられた。

- 1) 費用が高額となり費用対効果が得られない
- 2) 既存システムへ組み込む場合の改修規模が大きい
- 3) ユーザーが使いこなせない
- 4) 他社の導入事例が無い、もしくは少ない

< 2 > 見送り理由の改善

上記の見送りの理由は、BRMS 製品の機能向上や市場の成熟により解消されていくものとする。理由は、以下のとおりである。(番号は前項の導入見送りの理由の番号と対応している)

- 1) JbossDrools に代表するオープンソースソフトウェアの登場により低価格化が進み、BRMS の需要が高まるにつれてシェアの確保のため価格競争も激しくなっていくと考えられる
- 2) ILOG JRules の COBOL 用アドオンである「ILOG Rules for COBOL」など、既存システムへの組み込みを考慮した導入方法があり、前章でも述べたとおり COBOL と連携した事例が増えていくことにより、今後は更に発展していくと考えられる
- 3) よりユーザーが使いやすくなるためのユーザーインターフェース向上による操作性問題の解消
- 4) BRMS の導入事例は徐々に増えつつあり、導入事例が無いことに関しては時間が解決していくと推測される

4.6 新たな業務領域への活用

ここまでは BRMS が適するとされる領域について述べてきたが、ここからは、BRMS の特性を鑑みて、今まで BRMS の適用が検討されていなかった領域について検討する。一案として、営業支援領域への BRMS の導入について述べる。

営業活動に有効な各種の情報やイベントを BRMS にて処理し、リアルタイムに活用することができるようになれば、情報鮮度の高い情報を営業担当に伝達することが可能となる。その結果、営業活動の効率を向上させることができると考えられる。この場合、複雑なルールの設定とその変更を容易かつ柔軟に行えるという BRMS の利点が営業活動への効果を発揮し、顧客のニーズを的確に捉えることが可能になる。

例として、Web 上であらかじめ定めた特定のイベントの組み合わせ（複合イベント）を検知し、契約状況や営業活動と照らし合わせた結果、解約予兆がある顧客と判断した場合に、営業担当へフォロー活動を行うようにリアルタイムで指示を出す営業支援システムを考えた。

このシステムは、1) 複合イベント処理^{viii}検知、2) 条件判定、3) 対応アクションの 3 つのフェーズからなっている（図 16）。



図 16 : BRMS を用いた営業支援システム

各フェーズの内容を以下に示す

1) 複合イベント処理検知

顧客が Web マイページ上で操作したアクションをリアルタイムでチェックし、次のイベントを検知した場合に条件判定へと移行する。

- ① 契約内容照会が行われる
- ② 上記①から 10 分以内に減額試算を実施する

^{viii}直近の複数データをメモリなどの高速な入出力が可能な媒体に保管し、それらに対して分析を行うことで高速な処理が可能となる方式。

2) 条件判定

上記①、②のイベントを検知すると顧客情報を蓄積しているマスタから各種情報を取得し、その中から下記の条件を満たしているかをチェックする。

- ③ 年齢が 20～40 歳
- ④ 配偶者がいない
- ⑤ 営業担当による訪問が過去 2 年間行われていない

3) 対応アクション

上記③～⑤の条件を満たしている場合、解約予兆ありとして顧客対応のための人員を派遣し、訪問活動をとおして、保険契約の見直しなどの保全活動を行う。

BRMS を用いることで条件やアクションを柔軟に変更できるため、様々な状況に応じた営業活動を効果的に支援することが可能となる。

4.7 まとめ

現在、BRMS を保険業界に導入する事例はそれほど多くはないが、今後 BRMS が認知されていくに従い導入事例も増加し、中でも期待度が高い契約・加入業務などの領域では特にその効果を発揮していくものと思われる。また、BRMS の特性を最大限活かすことにより、上記以外の適用領域を見出し、新たな利用価値を生み出していくであろうと考えられる。

5章 おわりに

当研究グループでは、保険業界を取巻く環境の変化に対し保険システムが迅速に対応するため、BRMS がどのように活用できるか、また導入に際しどのような課題があるかを検討した。

1章、2章では、保険システムが抱える課題の解決策として、BRMS の機能が合致することを示した。具体的には、①BRMS ではユーザー部門と IT 部門双方が理解可能な「共通言語」でルールを表現できるため、両部門の認識が合わせやすくなる、②BRMS ではルールの容易な変更手順が提供されているため、要件変更への柔軟な対応が可能となる、③BRMS ではルールを他要素から分離して定義可能なため、システムの肥大化を抑制できる、という3点に着目した。

3章では、BRMS 導入に際して検討すべき事項として、まず、体制の観点からスクラム開発の有効性およびルール定義の具体的なプロセスを示した。次に、システムの観点からはホストでの BRMS 稼働に適したアーキテクチャとして、ホスト上の JVM で BRMS を稼働させる方式や、ルールを COBOL 変換する方式を紹介した。また、バッチ処理の高速化のために一括処理が重要であることを示した。

4章の前半では、賛助会員企業の約半数が BRMS について導入もしくは導入の検討を行っていることや、導入対象領域としては「契約・加入領域」が多数挙げられていることをアンケートにもとづき紹介した。後半では、導入実績の少ない「営業支援」領域について、複合イベントに対する条件判定と対応アクションを BRMS で扱うことにより、効果的な営業活動を行うことを提案した。

当研究成果が、各保険会社にて BRMS 導入を検討するきっかけとして、また導入時に直面する課題の解決指針として活用されることを期待する。

謝辞

当研究活動を行うにあたって、アンケート調査にご協力頂いた各社、製品レクチャーを実施頂いた日本プログレス株式会社様、株式会社日本アイ・ビー・エム様、株式会社日立ソリューションズ様、株式会社アーネスト・ビジネス・ソリューションズ様、レッドハット株式会社様（レクチャー実施順）、および関係者の方々に、この場を借りて深く御礼申し上げます。

参考文献

- [1] 「Are all Rules Business Rule? Not!」 < <http://www.brcommunity.com/b345.php> >(2014年2月1日アクセス)
- [2] 『保険毎日新聞』2012年8月16日付 「ビジネスルール管理システムの正しい理解で変革を目指す」
- [3] 西村 直人,永瀬 美穂,吉羽 龍太郎 (2013) 『SCRUM BOOT CAMP THE BOOK』(翔泳社)
- [4] ASCII.jp デジタル用語辞典 < <http://yougo.ascii.jp/caltar> >(2014年2月1日アクセス)

考察：BRMS のデモ開発検証

BRMS による効果を具体的に検証するため、当研究グループでは簡易的な「Web 保険申込システム」に BRMS を組み込んだデモ開発を行い、我々の考察を帰納的に証明することを目的とする。

A. 1 デモ開発の目的と背景について

ソーシャルネットワーキングサービス (SNS) やスマホ・タブレット等の各種デバイスの浸透によりインターネットのユーザー層は目紛しく拡大しており、ネット専従生命保険会社の登場やネットを活用したマーケティング施策（保険料お見積もり、資料請求、お申し込み、比較サイト等）を進めている保険会社が増えている。

しかし、Web によるダイレクトチャネルでのマーケティング活動は従来の営業職員による丁寧な顧客対応と比較すると現状は十分な対応とは言えない。ここにサービスレベルの向上による新たなビジネスチャンス創出の可能性があると思われる。

目まぐるしく変化するビジネスニーズにシステムが迅速に対応していくには、保険会社で多く採用されているウォーターフォールモデルの開発ではニーズの変化を捉えた迅速な対応がしづらい。ここに本稿で研究対象とした「ビジネスルールマネジメントシステム (BRMS)」の適用機会があると考えた。

そこで本章では、ダイレクトチャネルへの BRMS 適用について実装面での実現可能性・課題を探るとともに、本稿で示した BRMS の恩恵・課題について、より具体的な例を用いて整理すべく「Web 保険申込システム」のデモ開発を行い、その効果を検証する。

A. 2 「Web 保険申込システム」のデモ開発について

1) 「Web 保険申込システム」の概要

下図のとおり「保険料の見積もりから、そのまま保険の申込までを行える Web システム」（以下、「Web 保険申込システム」）としてデモ開発を行った。

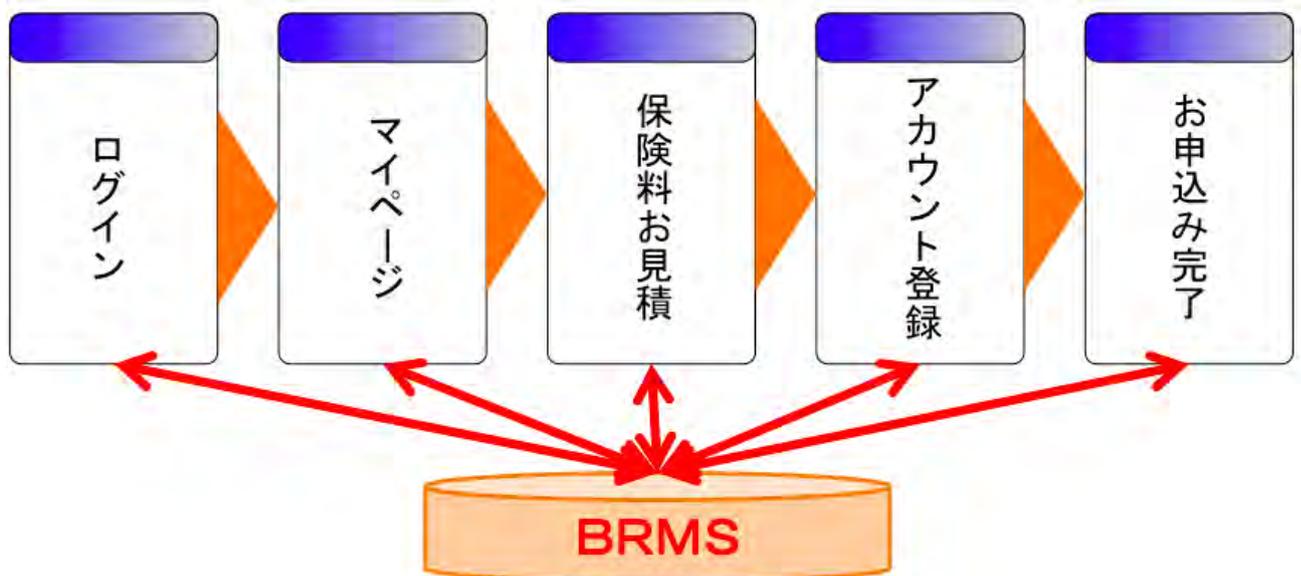


図. 1 「Web 申込システム」の申込フロー

実装に用いる BRMS および各種ミドルウェアは、環境の汎用性を考慮し、JBoss コミュニティの BRMS 「Drools」をはじめとした以下のオープンソースソフトウェア（以下、「OSS」）を採用した。

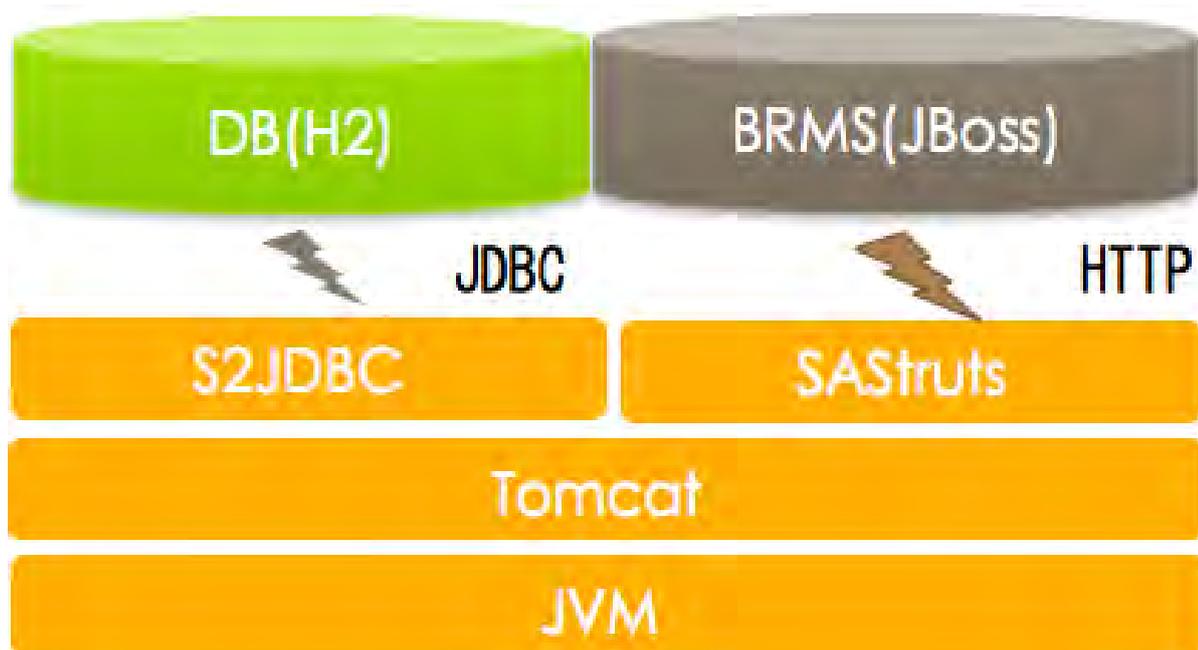


図. 2 「Web 保険申込システム」のシステム構成概略図

個々の構成要素の詳細については、下表の通り。

表. 1 システム構成要素一覧（概要）

| No | 構成要素 | 概要 |
|----|--------------|--------------------------------------|
| 1 | JVM | Java Virtual Machine (Java 実行環境) |
| 2 | Tomcat | JSP/Servlet のサーブレットコンテナサーバ |
| 3 | SAstruts | Web 開発を高速に行なうためのフレームワーク |
| 4 | S2JDBC | SQL とオブジェクトとの紐付け (O/R マッピング) フレームワーク |
| 5 | H2 | JVM 上で動作する DBMS |
| 6 | JBoss Drools | JBoss コミュニティの BRMS |

2) BRMS によるビジネスルール定義

「Web 保険申込システム」の保険料算出要件を以下のステップでビジネスルールとして実装した。

(ア) ステップ 1：保険料算出要件の定義

要件を以下の通り、簡便に対応できるように定義とした。

表. 2 保険料算出のビジネスルール要件

| ビジネス要件 | |
|---|---------------------------------|
| 保険料の算出は以下の数式で行う。 | |
| $\text{保険料} = \text{「年齢 (歳)」} \times \text{「保障額 (万円)」} \times \text{「保険料率 (%)」} \div 100$ | |
| ① | 60 歳以上の男女ならば、保険料率を 200% とする。 |
| ② | 59 歳～20 歳の男性ならば、保険料率を 100% とする。 |
| ③ | 59 歳～20 歳の女性ならば、保険料率を 100% とする。 |

(イ) ステップ 2 : ビジネスルールの一般化

ビジネスルールから共通項を抜き出して、以下のように一般化する。

表. 3 保険料算出のビジネスルールの一般化

| ビジネスルールの一般化 | |
|---|--|
| お客さまの年齢が「〇〇～●●」かつ性別が「△△」ならば、保険料率を「××」とする。 | |
| ※ 「〇〇」・「●●」・「△△」・「××」にはシナリオごとの定義値が設定される。 | |

(ウ) ステップ 3 : 決定表の対象項目の定義

ビジネスルールの雛形から条件項目を「年齢（下限、上限）」および「性別」、条件によって確定する項目「保険料率（%）」として、下図の「保険料率算出決定テーブル」（項目のみ定義）を定義する。

| ルール決定表名 | RuleTable 保険料率算出決定テーブル | | ACTION |
|-------------------|------------------------|-----------|-----------------|
| 制御行 | CONDITION | CONDITION | ACTION |
| 画面・PDF等のオブジェクトの宣言 | | | |
| フィールドの宣言および条件 | | | |
| ケース内容 | 年齢 (下限,上限) | 性別 | 保険料率(%) |
| | 条件用項目を列挙 | | 実行用項目を列挙 |

図. 3 保険料率算出決定テーブル（項目列挙のみ）

(エ) ステップ 4 : 具体的なロジック実装

ビジネスルールの雛形から下図の通りに具体的なロジックを定義する。

| ルール決定表名 | RuleTable 保険料率算出決定テーブル | | ACTION |
|-------------------|---|-----------|--|
| 制御行 | CONDITION | CONDITION | ACTION |
| 画面・PDF等のオブジェクトの宣言 | | | |
| フィールドの宣言および条件 | 年齢:age >= \$1, age <= \$2, 保障額:sumInsured, 保険料:premium | 性別 : sex | modify(保険料お見積もり画面){ setPremium(年齢 * 保障額 * \$1 / 100); } update(保険料お見積もり画面); |
| ケース内容 | 年齢 (下限,上限) | 性別 | 保険料率(%) |
| | 詳細なロジックを定義 (Excel式レベル) | | |

図. 4 保険料率算出のビジネスルール（ロジック追加）

(オ) ステップ 5 : ビジネスシナリオの定義

シナリオごとの年齢（下限、上限）・性別および保険料率（%）を定義する。

| ルール決定表名 | RuleTable 保険料率算出決定テーブル | | |
|-------------------|---|-----------|---|
| 制御行 | CONDITION | CONDITION | ACTION |
| 画面・PDF等のオブジェクトの宣言 | 保険料お見積もり画面:SimulationForm | | |
| フィールドの宣言および条件 | 年齢:age >= \$1, age <= \$2, 保障額:sumInsu 保険料:premium | | modify(保険料お見積もり画面){ 年齢 * 保障額 * \$1 / 100}; |
| ケース内容 | 年齢 (下限,上限) | | 保険料率(%) |
| 高齢者用割増パターン | 50,60 | | 200 |
| 男性用保険料パターン | 20,49 | 男 | 100 |
| 女性用保険料パターン | 20,49 | 女 | 100 |

シナリオに応じた
定義値を設定

図. 5 保険料率算出のビジネスルール (シナリオ追加)

A. 3 保険システムが抱える課題に対する BRMS の効果の考察

保険ビジネスの迅速化を阻む課題として以下の3点を示し、BRMSによる改善効果を検討した。

- ✓ 課題1：ユーザー部門、IT部門双方の認識が合わせにくい
- ✓ 課題2：要件変更への柔軟な対応が困難
- ✓ 課題3：保険システムの肥大化

1) 「課題1：ユーザー部門、IT部門双方の認識が合わせにくい」に対する効果について

保険料率算出のビジネスルール (Excel) は自然言語形式ではないが、ルールをマトリックスとして定義できることから、直感的にユーザー部門にも理解しやすいフォーマットになっており、以下の通りIT部門とユーザー部門による分業化が可能である。

| ルール決定表名 | RuleTable 保険料率算出決定テーブル | | |
|-------------------|---|-----------|---|
| 制御行 | CONDITION | CONDITION | ACTION |
| 画面・PDF等のオブジェクトの宣言 | 保険料お見積もり画面:SimulationForm | | |
| フィールドの宣言および条件 | 年齢:age >= \$1, age <= \$2, 保障額:sumInsured, 保険料:premium | 性別:sex | modify(保険料お見積もり画面){ setPremium(年齢 * 保障額 * \$1 / 100); update(保険料お見積もり画面); |
| ケース内容 | 年齢 (下限,上限) | 性別 | 保険料率(%) |
| 高齢者用割増パターン | 50,60 | | 200 |
| 男性用保険料パターン | 20,49 | 男 | 100 |
| 女性用保険料パターン | 20,49 | 女 | 100 |

ユーザー部門は
シナリオの入力だけ

IT部門はルール
テンプレートを作成

図. 6 保険料率算出のビジネスルール (分担イメージ)

また、保険料率算出のビジネスルール (Excel) はシステムから切り離して Excel ファイルとして構成していることから、以下の通りユーザー部門が直接ビジネスルールを参照および編集することができる。

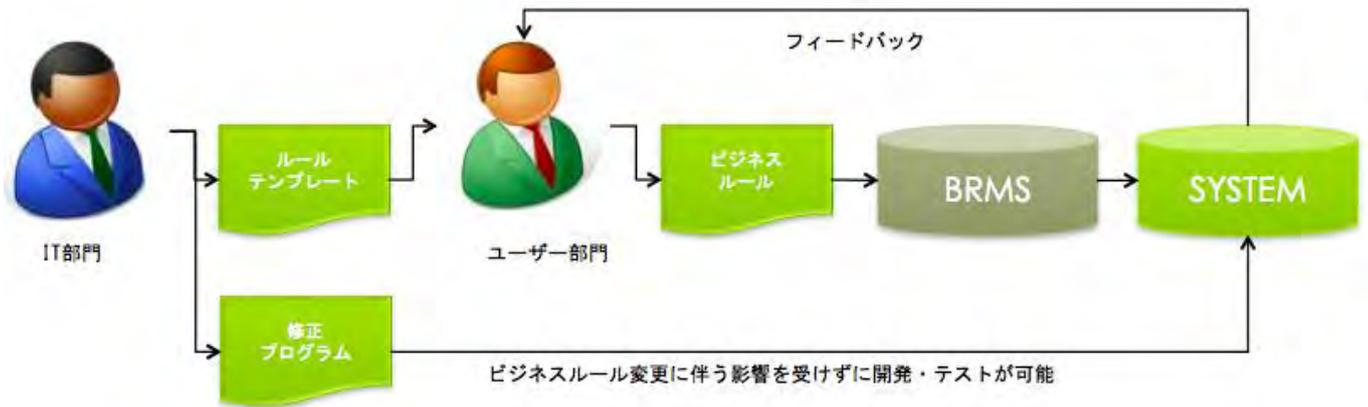


図.7 ビジネスルールの開発分業フローイメージ

このことから、プログラミング経験のないユーザー部門でも、ビジネスルールをシステムに反映しながら要件を調整できるので、従来の上流工程と比べて両部門の認識を合わせやすいと考えられる。

2) 「課題2：変更要件への柔軟な対応が困難」に対する効果について

保険料率算出のビジネスルール (Excel) の入力パラメーター (性別・年齢・保障額) に変更がなければ、システムを改修することなく、Excel の修正のみで変更内容をシステムに反映することが可能である。例えば、保険料率算出のビジネスルール (Excel) に対して変更要件「女性の保険料を半額にする」を反映する場合、下図の吹き出しのように保険料率の変更およびビジネスルールシナリオを追加するだけで、システムを変更すること無く、システムに変更要件を反映することができる。

| ルール決定表名 | RuleTable 保険料率算出決定テーブル | | |
|-------------------|---|-----------|---|
| 制御行 | CONDITION | CONDITION | ACTION |
| 画面・PDF等のオブジェクトの宣言 | 保険料お見積り画面:SimulationForm | | |
| フィ | 年齢:age >= \$1, age <= \$2, 保障額:sumInsured, 保険料:premium | 性別: sex | modify(保険料お見積り画面) { setPremium(年齢 * 保障額 * \$1 / 100); } update(保険料お見積り画面); |
| シナリオを2つに分離 | | | |
| ケース内容 | 年齢 (下限 上限) | 性別 | 保険料率(%) |
| 高齢者用割増パターン(男) | 50,60 | 男 | 200 |
| 高齢者用割増パターン(女) | 60,20 | 女 | 100 |
| 男性用保険料パターン | | | 100 |
| 女性用保険料パターン | | | 50 |

図.8 女性の保険料を半額にした保険料率算出のビジネスルール

さらに、BRMS ではビジネスルールの構文をチェックするために、静的チェック (文法およびルール間チェック) 機能があり、ビジネスルールをインポートする際にケアレスミスや考慮漏れを即座にチェックできる。



図. 9 BRMS の静的チェック（文法・ルール間チェック）

また、BRMS にはビジネスルール単体でのテスト機能が備わっており、下図のようにビジネスルール変更に伴う影響を視覚的に確認できるため、漏れなく調査・動作検証をサポートできる。

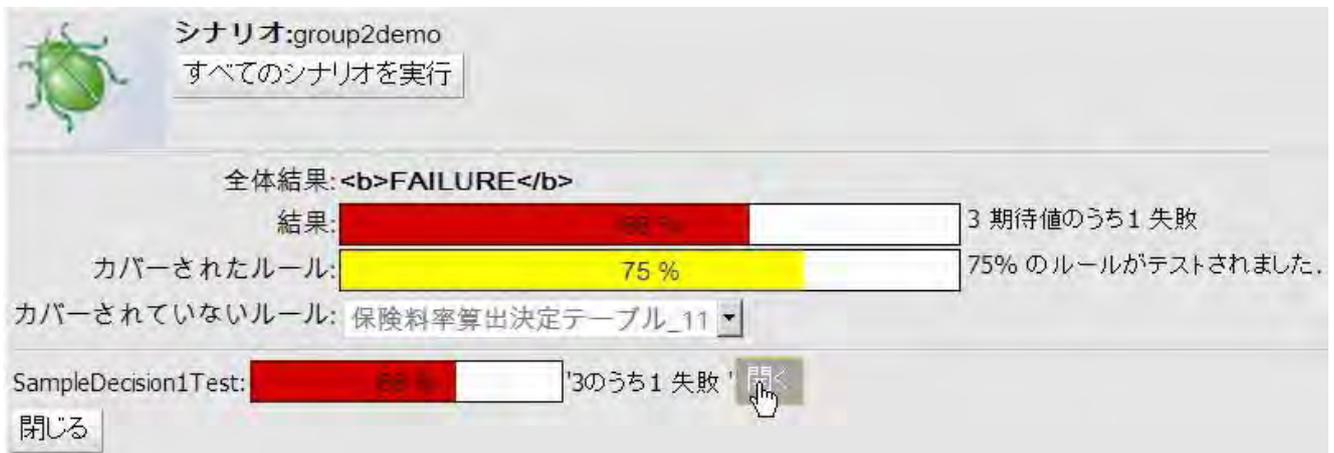


図. 10 BRMS の単体テスト結果（概要）

なお、単体テストの詳細なシナリオは、下図のように入力データとそれに対する期待値を定義して、テスト結果と照らし合わせることができるため、ユーザー部門でも十分に活用しやすいと思われる。

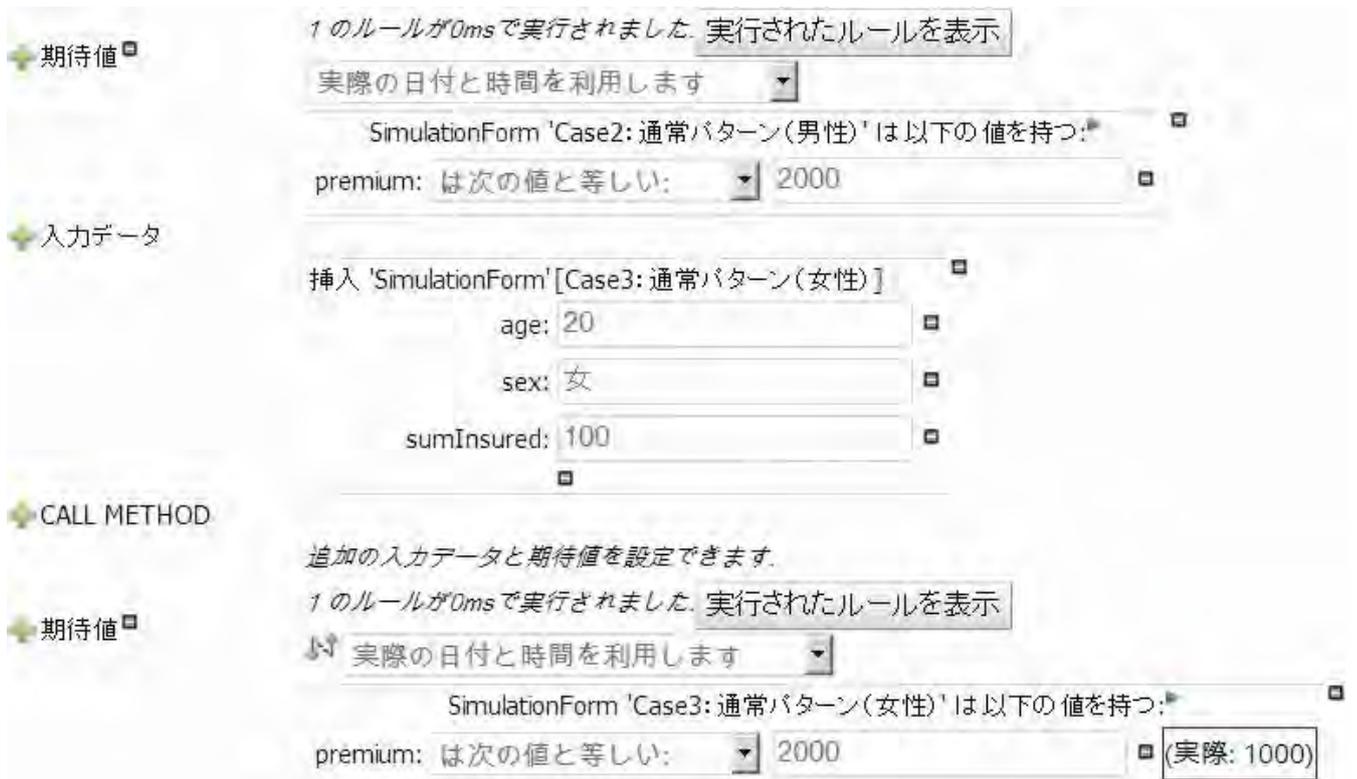


図. 11 BRMS の単体テスト結果 (詳細)

3) 「課題 3 : 保険システムの肥大化」に対する効果について

保険料率算出のビジネスルール (Excel) は「Web 保険申込システム」から切り離して BRMS に移行し、システムと BRMS の HTTP 通信で連携したことで、両者はインタフェースパラメーターのみに依存する疎結合の関係と言える。今後、他システム開発を行う上で、BRMS に定義されたビジネスルールは再利用可能なモジュールとして扱いやすく、ビジネスルールの肥大化抑止に効果的である。

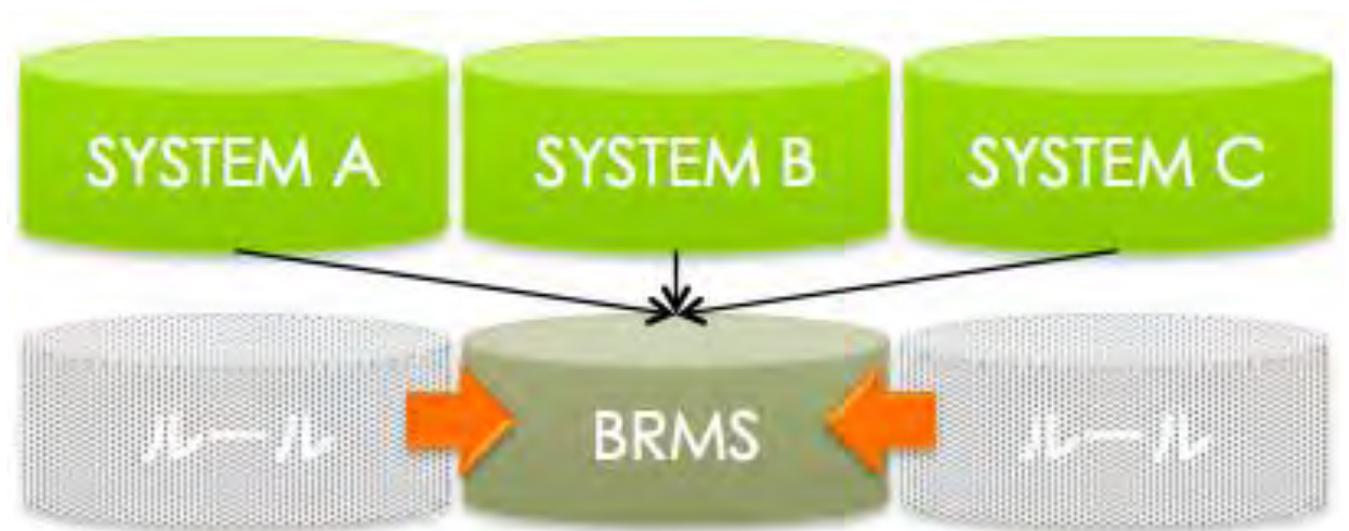


図. 12 BRMS によるビジネスルールの集約化

また、保険料率算出のビジネスルール (Excel) は以下の十数ステップのロジックを Excel の 1 行に集約できることから、ビジネスルールをスリムに管理することができる。

```

1. | #generated from Decision Table
2. | #From row number: 10
3. | rule "保険料率算出決定テーブル_10"
4. |
5. |   no-loop true
6. |   when
7. |     保険料お見積もり画面:SimulationForm(年齢:age >= 50,
8. | age <= 60,
9. | 保障額:sumInsured,
10. | 保険料:premium)
11. |   then
12. |     modify( 保険料お見積もり画面 ) {
13. | setPremium(年齢 * 保障額 * 200 / 100);
14. | }
15. | update(保険料お見積もり画面);
16. | end

```

図. 13 保険料率算出のビジネスルールのルール言語化（抜粋）

一方、BRMS を呼び出すシステム側は単に BRMS とシステムのパラメーターと実行結果を連携するだけのシンプルな設計（下図参照）となり、システム上の複雑なロジックがなくなることで保守しやすくなる。

```

// BRMS の呼び出し準備 ↓
KnowledgeBase kbase = RuleManager.createKnowledgeBase

// BRMS にパラメータを設定 ↓
FactType simulationType = kbase.getFactType("group2demo", "SimulationForm");
Object simulationBean = simulationType.newInstance();
simulationType.set(simulationBean, "age", simulationForm.age);
simulationType.set(simulationBean, "sumInsured", simulationForm.sumInsured);
simulationType.set(simulationBean, "sex", simulationForm.sex);

// BRMS を実行 ↓
ksession = kbase.newStatefulKnowledgeSession();
ksession.insert(simulationBean);
ksession.fireAllRules();

// BRMS の判定結果を取得 ↓
simulationForm.premium = (Integer) simulationType.get(simulationBean, "premium");

```

①パラメータをセット

②BRMSを実行

③実行結果を取得

図. 14 BRMS を呼び出すプログラム

以上3点から、BRMS を導入して正しくビジネスルールを設計することで、保険システムが抱える3つの課題に対して期待できる改善効果をもたらすと考える。