

ビジネスプロセスモデルを活用した BAM システム構築の提案

細川 武彦†, 上野 浩一郎†, 馬場 昭宏†, 砂田 英之†,
北山 泰英†, 相浦 利治†, 茂木 強†
†三菱電機(株) 情報技術総合研究所

近年、業務環境の変化速度の増大や、IT システムの複雑化により、実業務と IT システムの乖離が大きくなってきている。実業務に IT システムを追従させる技術として、BPM(Business Process Management)/BAM(Business Activity Monitoring)技術が注目されている。本稿では、ビジネスプロセスモデルを活用した BAM システムを構築する方法について提案する。また、提案した方法のフィージビリティを確認するためのプロトタイプシステムを開発し、実システムへ適用するための課題について整理した結果を報告する。

Proposal of BAM system construction that effectively uses business process models

Takehiko HOSOKAWA †, Koichiro UENO †, Akihiro BABA †, Hideyuki SUNADA †
Yasuhide KITAYAMA †, Toshiharu AIURA † and Tsuyoshi MOTEGI †

† Information Technology R & D Center, Mitsubishi Electric Corporation

Recently, the unbridgeable gap between real business and IT system is growing, because of the complication of the IT system and the increase of the change speed of the business environment. BPM(Business Process Management)/BAM(Business Activity Monitoring) technology is paid to attention as a technology that fills in this gap. In this paper, we propose the method of BAM system construction that effectively uses business process models. And, it reports on the result of arranging the problem to apply to the real system.

1. はじめに

今日、企業の経営活動は情報システムなくしては遂行できない状況となっている。一方、業務環境の変化速度の増大や、IT システムの複雑化により、実業務と IT システムの乖離が大きくなってきており、大きな問題となっている。そこで、IT システムを変化の激しい業務環境へ追従させるための技術として BPM(Business Process Management)や BAM (Business Activity Monitoring)といった技術が注目されている(参考文献[1]-[4])。

これら技術に関する標準化については BPMI, OASIS, OMG といった標準化団体により進められており、一部実用段階に入っている。しかし、これら標準はビジネスプロセスモデルの設計や設計したモデルを実行することを主目的としたものであり、

監視や分析に関してはあまり考慮されていない。

本稿では、ビジネスプロセスモデルを活用した BAM システムを構築する方法について提案する。また、提案した方法のフィージビリティを検証するためのプロトタイプシステムを開発し、実システムへ適用するための課題について整理した結果について報告する。具体的には 2 章において、背景となる「BPM/BAM 技術」、「標準化の動向」について簡単に説明する。3 章では、これらの技術を組み合わせることにより、ビジネスプロセスモデルを活用して BAM システムを構築する方法に関して提案する。4 章では、提案した方法のフィージビリティを検証するためのプロトタイプシステムについて示し、実システムへ適用するための課題について整理し、最後に 5 章に今後の展開についてまとめる。

2. 背景

2.1. BPM/BAM 技術

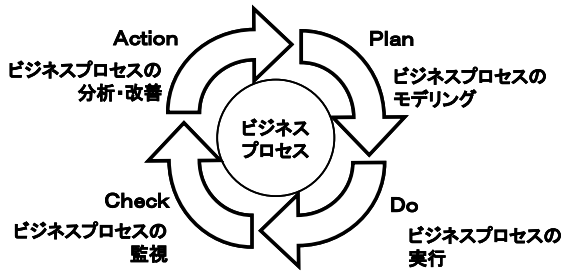


図 1 BPM ライフサイクル

BPM とは広義には「ビジネスプロセスを継続的に改善する活動」を指し、「業務の一連の流れ(ビジネスプロセス)を可視化し、モデリング、実行、監視、分析・改善のサイクル(以降 BPM ライフサイクル(図 1 参照))を回すことにより、継続的に業務改善を行っていく活動、あるいはそのための IT ソリューション」のことである。また、狭義の意味では、BPM ライフサイクルの Plan-Do のフェーズを実現するための技術やソフトウェアを指す場合もある。

BAM とは広義には「継続的なビジネスプロセス改善を遂行する上で、その実施状況や実績を監視する活動、あるいはそのための IT ソリューション」を指し、狭義には BPM ライフサイクルの Check-Action を実現するための技術やソフトウェアを指す場合もある。

BAM における監視対象は KPI(Key Performance Indicator: 重要業績評価指標) と呼ばれる値であり、業種や会社毎、利用者の視点毎に異なる。BAM の利用者は検討する業務改善の視点/範囲毎に大きくは経営者、管理者、担当者 の 3 つに分類される(図 2 参照)。

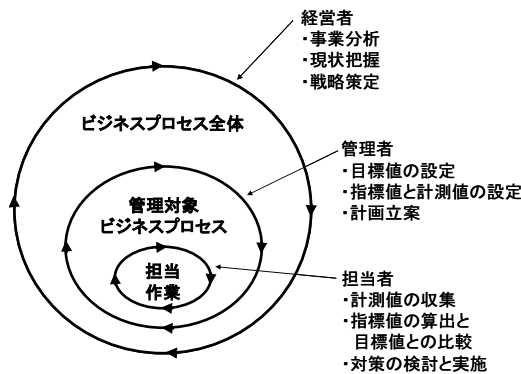


図 2 BAM 利用者の分類

経営者: 事業分析、現状把握、戦略策定などの作業を実施する。改善の範囲はビジネスプロセス全体にわたり、業務環境の大きな変化に対応するための改善を行う。

管理者: 目標値の設定、指標値と計測値の設定、計画の立案などの作業を実施する。改善の範囲は管理対象のビジネスプロセスであり、ビジネスプロセスの変更、リソースの再配置、目標値/指標値の補正などを随時実施する。

担当者: 計測値の収集、指標値と目標値の比較、対策の検討/実施などの作業を行う。改善の範囲は担当する作業であり、各作業の優先度付けや実施順序の変更などを随時実施する。

2.2. 標準化の動向

本節では BPM 技術に関する標準化の動向として、ビジネスプロセスの表記、および定義に関する仕様について説明する。

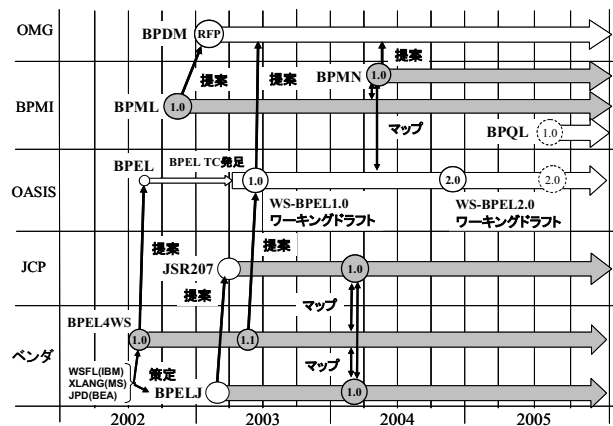


図 3 標準化の動向

図 3 に各標準化団体が策定しているビジネスプロセスの表記、および定義に関する仕様、および仕様間の関連を示す(参考文献[5]-[10])。図 3 より各団体/仕様間で提案やマッピングが活発に行われていることがわかる。ここでは主要な仕様である BPDM, BPMN, WS-BPEL に関して簡単に紹介する。

BPDM (Business Process Definition Metamodel)

OMG(Object Management Group)が策定しているビジネスプロセスのメタモデルである。2003 年 1 月 6 日に RFP(Request For Proposal: 提案依頼書)が発行され、各標準化団体が提案を行っている。2006 年 2 月現在では、以下のものが定義されているが、まだ完成された仕様とはなっていない。

- コンセプトレベルのメタモデルのクラス図
- BPDM で定義しているクラスと UML のクラスとの対応 (以下、BPDM セマンティクス)
- BPDM から BPEL へのマッピング
- BPMN から BPDM へのマッピング

BPMN (Business Process Modeling Notation)

BPMI(Business Process Management Initiative)で策定されたビジネス・プロセスを図を用いて表現

するときの表記に関する仕様である。また、表記要素と BPML や WS-BPEL とのマップについて定義されている。BPMN 1.0 が 2004 年 5 月 3 日に正式リリースされている。

WS-BPEL (Web Services - Business Process Execution Language)

OASIS(Organization for the Advancement of Structured Information Standards)で策定されている Web サービスを基本としたビジネスプロセスの挙動を定義するための書式に関する仕様である。IBM の WSFL, Microsoft の XLANG, BEA の JPD を基に開発された BPEL4WS が 3 社共同で OASIS に提案され、その提案を基に標準化が行われている。また BPEL4WS で定義したビジネスプロセスを Java 実行環境で実行するための BPELJ を JCP(Java Community Process)に提案しており、JSR207 として公開されている。

上記の各仕様を用いて、BPMN で記載されたビジネスプロセスモデルを WS-BPEL, JSR207 へマッピングすることにより、Java 環境にて実行可能となるように標準化が進められている。また、他の表記法や定義についても、BPMN へとマッピングすることにより、相互変換が可能となるように検討が進められている。しかし、これらの仕様は基本的には記述したモデルを実行可能とすることを主目的としており、ビジネスプロセスを実行した結果を監視、分析することについてはあまり検討されていない。OMG では、BSBR(Business Semantics of Business Rules, PRR(Production Rule Representation), OSM(Organization Structure Metamodel) 等の RFP を発行しているが、現時点では、まとまった資料は公開されていない。本稿では監視、分析の観点でビジネスプロセスモデルを活用することにより、BAM システムを構築する方法について提案する。

3. BAM システム構築の提案

3.1. BAM システムの要件

本節では BAM システム構築の要件について記載する。BAM の利用者は 2.1. で記載したとおり、利用者の視点毎に監視すべき範囲、改善 (設定変更) のタイミングが異なる。また、BAM システムの利用者として BAM システムの設定変更や運用を行うシステム管理者やビジネスプロセスを設計、修正する業務管理者が存在し、各利用者が保持しているスキルが大きく異なっている。各利用者が行う設定間では整合性が取れている必要があり、ビジネスプロセスが変更された場合にも追従できなければならない。別の要件としては、BAM システムはビジネスプロ

セスの監視、分析を行うものであるため、BAM システムの動作が実業務へ与える影響は最小限に抑える必要がある。

要件をまとめると以下の通りとなる。

要件 1 : 利用者の視点/スキルに応じた設定を可能とし、各設定間の整合性を保つ必要がある。

要件 2 : ビジネスプロセスが変更された場合にも、各設定は変更を追従できる必要がある。

要件 3 : BAM システムの監視負荷は実業務への影響を最小限に抑える必要がある。

3.2. BAM システムの構成

本節では 3.1. に記載した要件を満たすための BAM システムの構成に関して記載する。

図 4 が提案する BAM システムの構成である。

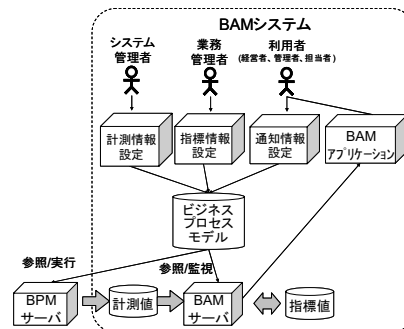


図 4 BAM システム構成図

図中の BPM サーバは、ビジネスプロセスを実行するモジュールであり、BAM サーバはビジネスプロセスの実行を監視するモジュールである。BAM アプリケーションは BAM サーバと通信し、利用者に各種情報を提供するモジュールであり、計測情報設定、指標情報設定、通知情報設定は、それぞれ後述する各情報を設定するモジュールである。

3.1. 節の要件 1 を満たすために以下のように各利用者毎の設定情報を分離し、ビジネスプロセスモデルと関係付けることにより、各設定の整合性を保つようにする。

システム管理者はビジネスプロセスモデルに対して、計測情報設定モジュールを用いて計測情報を付加する。計測情報とは、ビジネスプロセス実行時にビジネスプロセスインスタンスに対応した属性を表す計測値を取得するために必要な情報である。BPM サーバは計測情報を元にビジネスプロセスの実行時に計測情報に設定されている方法を用いて値を取得し、ビジネスプロセスモデルに対応付けた計測値として保存する。

業務管理者はビジネスプロセスモデルに対して指標情報設定モジュールを用いて指標情報を付加する。指標情報とは、業務を評価するために設定される指

指標値を、計測値とビジネスプロセスモデルを元として算出するために必要な情報である。BAM サーバは指標情報を元に BPM サーバが保存した計測値から指標値を算出する。

利用者はビジネスプロセスモデルに対して通知情報設定モジュールを用いて通知情報を付加する。通知情報とは、業務に影響を与える状況、もしくは兆候を伝えるために必要となる、計測値、指標値、およびビジネスプロセスモデルを元として状況を判断するための情報である。

本構成により、システム管理者、業務管理者、利用者は各自の関心事、およびその元となる情報のみについて検討を行えばよく、また設定のために新しいスキルを習得する必要がなくなる。

3.3. ビジネスプロセスメタモデル

3.1.節の要件 2 を満たすために、各設定情報はビジネスプロセスモデルを元として関係付け、ビジネスプロセスや各設定が変更された場合にも設定が追従可能なようにビジネスプロセスのメタモデルを設計する。ビジネスプロセスのメタモデルは 2.2.節にて記載した OMG で現在策定中の BPDM を参照して、設定情報を付加する形式にて設計する。参考として、BPDM のコンセプトメタモデルを図 5 に示す。

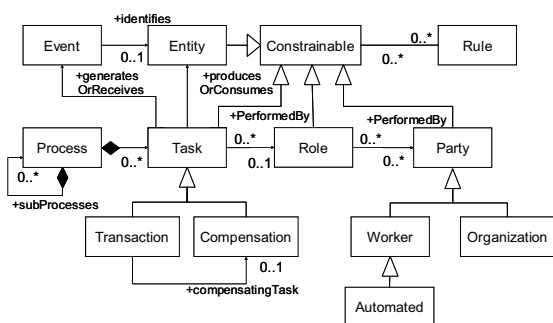


図 5 BPDM のコンセプトメタモデル

ビジネスプロセスモデルの変更時にも、上記メタモデルを用いて、ビジネスプロセスモデル間のマッピングを管理することにより、各種設定情報の追従が可能となる。

3.4. BPM との連携方式

3.1.節の要件 3 を満たすために、ビジネスプロセスを実行する BPM サーバと、監視、分析を行う BAM サーバを分離した構成とする。BPM サーバ、BAM サーバは共に同一のビジネスプロセスモデル、および関連付けられた各種設定情報を参照して動作することにより連携が可能となり、BAM サーバの監視による業務への影響を最小化することが出来る。また、図中では、計測値として外部記憶装置を用いて連携するように記載されているが、イベント等の非同期通信を用いたり、ESB(Enterprise Service

Bus)の購読/公開のメカニズムを用いて連携する方式も考えられる。このように構成することにより、複数の BPM サーバ上で実行されるビジネスプロセスを一台の BAM サーバにより統合的に監視したり、トランザクションが多いために分析に時間がかかるような場合に、複数の BAM サーバを設置するなど、スケーラビリティの高い柔軟な構成とすることが可能となる。

4. 実システム適用への課題と取り組み

4.1. プロトタイプシステム

本節では 3 章で示したアーキテクチャのコア部分であるビジネスプロセスモデルを用いた各モジュール間の連携に関してフィージビリティを確認するため、および実環境への適用時の課題を洗い出すために構築したBAMシステムのプロトタイプに関して記述する。監視・分析対象業務は、図 6に示すB2B販売管理システムの見積もり/受注業務のビジネスプロセスである。

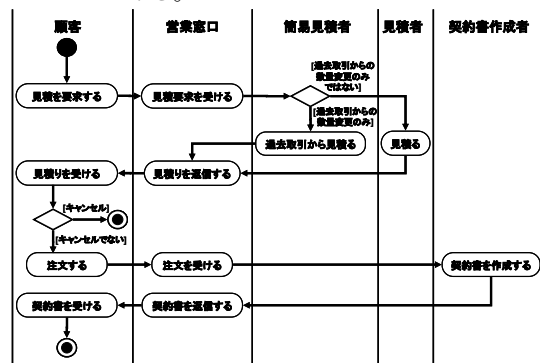


図 6 監視対象プロセス

プロトタイプシステムは図 7のような構成であり、監視対象システム、BAMサーバ計算機、クライアントから構成される。

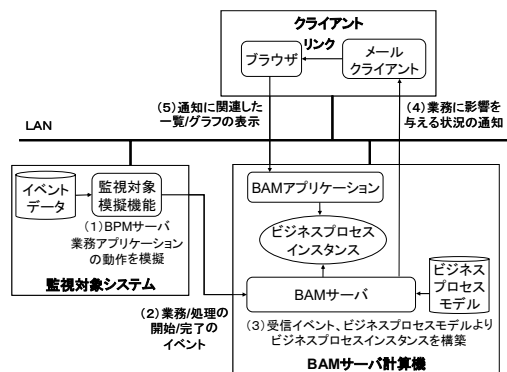


図 7 プロトタイプシステムの構成

各構成要素は以下の(1)～(5)のような順序で処理を実行する。

(1) 監視対象模擬機能では BPM サーバや各アプリケーションの動作を模擬し、図 6で示した監視対象プロセスを動作させる。

(2) 模擬した監視対象プロセスに対応して、業務/処理の開始/終了や計測情報に従って収集した計測値をイベントとして BAM サーバに通知する。

(3) BAM サーバは、受信したイベントからビジネスプロセスモデルに従い、ビジネスプロセスインスタンスを構築し、計測値や指標値をビジネスプロセスインスタンスに関連付けて管理する。

(4) ビジネスプロセスモデルに対応付けて設定された通知情報に従って監視を行い、業務に影響を与える状況(兆候)になった場合に利用者に対して、状況を伝える通知を行う。

(5) 利用者は通知された内容に従い、各値の一覧やグラフを参照して、対策を検討する。

今回のプロトタイプシステムで使用したメタモデルは、BPDM のコンセプトメタモデルに、BPDM セマンティクスで定義されているクラスを追加し、UML のクラス間の整合性を考慮して作成した。また、各種設定情報がプロセス、ノードに対して付加できるように拡張を行った。図 8 に作成したメタモデルの主要な要素の関係を示す。

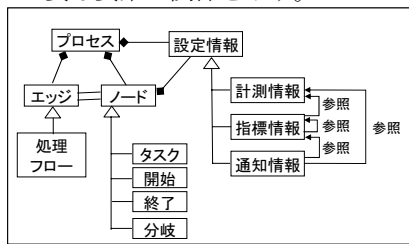


図 8 メタモデル概要

通知情報、指標情報は表 1, 2 のような形式で設定し、設定された各種設定情報は図 9 のようにビジネスプロセスモデルに関連付けられて管理されるようにした。

表 1 通知情報

| 通知名 | 条件式 | アクション |
|-------------|------------------------------------|--------------|
| 注文獲得率警告 | 注文獲得率 < 50 | 経営者にメール |
| 見積回答時間警告 | 見積回答猶予時間 != NULL && 見積回答猶予時間 < 300 | タスクの担当者にメール |
| 見積回答時間超過 | 見積回答猶予時間 != NULL && 見積回答猶予時間 < 0 | タスクの担当者にメール |
| 普通見積時間警告 | 普通見積猶予時間 != NULL && 普通見積猶予時間 < 300 | タスクの担当者にメール |
| 普通見積時間超過 | 普通見積猶予時間 != NULL && 普通見積猶予時間 < 0 | タスクの担当者にメール |
| 簡易見積時間警告 | 簡易見積猶予時間 != NULL && 簡易見積猶予時間 < 300 | タスクの担当者にメール |
| 簡易見積時間超過 | 簡易見積猶予時間 != NULL && 簡易見積猶予時間 < 0 | タスクの担当者にメール |
| 未処理普通見積件数超過 | 未処理普通見積件数 >= 10 | プロセスの管理者にメール |
| 未処理普通見積件数復帰 | 未処理普通見積件数 <= 5 | プロセスの管理者にメール |
| 未処理簡易見積件数超過 | 未処理簡易見積件数 >= 10 | プロセスの管理者にメール |
| 未処理簡易見積件数復帰 | 未処理簡易見積件数 <= 5 | プロセスの管理者にメール |

表 2 指標情報

| 指標値 | 算出式 |
|-----------|---|
| 注文獲得率 | 件数(見積を受ける[完了日時=NULL]) / 件数(注文する) * 100 |
| 見積回答時間 | 見積りを返信する. 完了日時 - 見積り要求を受ける. 開始日時 |
| 普通見積時間 | 普通見積. 完了日時 - 普通見積. 開始日時 |
| 簡易見積時間 | 簡易見積. 完了日時 - 簡易見積. 開始日時 |
| 見積回答猶予時間 | (見積りを返信する. 完了日時 == NULL)? 見積りを返信する. 完了期限 - 現在時刻 : NULL |
| 普通見積猶予時間 | (普通見積. 完了日時 == NULL)? 普通見積. 完了期限 - 現在時刻 : NULL |
| 簡易見積猶予時間 | (簡易見積. 完了日時 == NULL)? 簡易見積. 完了期限 - 現在時刻 : NULL |
| 未処理簡易見積件数 | 件数(簡易見積[完了日時 = NULL]) |
| 未処理普通見積件数 | 件数(普通見積[完了日時 = NULL]) |

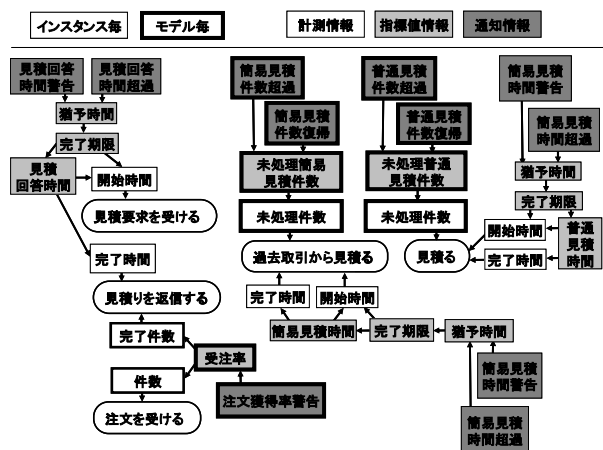


図 9 各種設定情報の例

各利用者(経営者、管理者、担当者)は、BAM サーバからの通知を受け、通知の内容から利用者毎に異なる関心事についての一覧やグラフを参照することが可能である。実際にプロトタイプシステムを実行した画面例を図 10 に示す。画面例は管理者に「未処理簡易見積超過」の通知が電子メールにより送られ、電子メールの内容に記述されたリンクにより、「見積回答時間毎の未処理件数のグラフ」や「未処理のタスクの一覧」を確認したものである。



図 10 画面例

4.2. 実環境適用の課題

プロトタイプシステムを用いて検証した結果、提案したBAMシステムのアーキテクチャのコア部分であるビジネスプロセスモデルを用いた各モジュール間の連携が可能であることが確認できた。

また、実システムへの適用においては、以下のよう
な課題があることが分かった。

- **ビジネスプロセスインスタンスのライフサイクル管理**

プロトタイプシステムではイベントから生成するビジネスプロセスインスタンスをメモリ上のみで管理している。実システムへの適用を考えた場合には管理するプロセスやタスクの数が膨大となるため、永続化やメモリ開放などのライフサイクル管理を行う必要があると考えられる。ここで、ビジネスプロセスインスタンスに関連付けて管理されている計測値や指標値はそれを参照する、指標値、通知条件やグラフ一覧などによりライフサイクルが大きく異なっていることが分かった。効率的なライフサイクル管理を行うための管理単位や方式についての検討が必要である。

- **イベントの整合性確認と補完処理**

実業務への影響を最小限とするために、BPM サーバと BAM サーバは疎結合で連携しており、また BAM サーバは複数の BPM サーバからのイベントを扱う必要がある。このとき、イベントの到着順序は保障されておらず、また欠落する可能性がある。これらが原因となって分析結果が大きく異なる場合、BAM システムとしては致命的な問題となる可能性がある。対策として、ビジネスプロセスモデルを元として、受信したイベントの整合性を確認する方式や、欠落したイベントを補完してビジネスプロセスインスタンスを構築する方式について検討する必要がある。

- **既存システムとの接続**

現在稼動しているシステムの大半は BPM のような機能を用いて動作しているわけではない。本稿で提案した BAM システムの適用範囲を拡大するためには、既存のシステムに対して、同様なシステムを構築する方式を検討する必要がある。このとき、既存システムからビジネスプロセスモデルを定義(抽出)する方式や、既存システムから各種計測値を取得する方式が大きな課題となると考えられる。

5. おわりに

本稿では、ビジネスプロセスモデルを活用することにより、各利用者の視点/スキル毎に設定内容を変えられ、設定間の整合性を保つ BAM システム構築の方法についての提案を行った。また、プロトタイプシステムを開発/検証することにより、提案したアーキテクチャのコア部分であるビジネスプロセスモデルを用いた各モジュール間の連携が実現可能であることを確認し、実システムへ適用する上での課題について洗い出した。

今後、洗い出された課題を解決するための方式について検討すると共に、今回のプロトタイプシステムで確認していない各モジュールについてのフィージビリティの確認を実施する予定である。

また、標準化の動向を踏まえて、適用範囲の広い BAM システムの開発を継続して行っていく予定である。

参考文献

- 1) Gartner : Business Activity Monitoring: BAM Architecture, Gartner SYMPOSIUM ITXPO(2003)
- 2) Gartner : Magic Quadrant for Business Rule Engines, 2005(2005)
- 3) Derek Miers, Paul Harmon : The 2005 BPM Suites Report, BUSINESS PROCESS TRENDS A BPT Report(2005)
- 4) Curtis Hall, Paul Harmon : Enterprise Architecture, Process Modeling & Simulation Tools Report, BUSINESS PROCESS TRENDS A BPT Report(2005)
- 5) OMG : BPD (Business Process Definition Metamodel) RFP, <http://www.omg.org/cgi-bin/doc?bei/2003-01-06>
- 6) BPMI : BPMN (Business Process Modeling Notation) , <http://www.bpmi.org/specifications.esp>
- 7) OASIS : Web Services Business Process Execution Language(WSBPEL) TC, http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel
- 8) BEA : BPEL4WS (Business Process Execution Language for Web Service) , <http://dev2dev.bea.com/technologies/webservices/BPEL4WS.jsp>
- 9) BPMI : BPML (Business Process Markup Language) , <http://www.bpmi.org/specifications.esp>
- 10) JSR 207 エキスパートグループ - PD4J (Process Definition for Java) , <http://jcp.org/en/jsr/detail?id=207>