



Oracleホワイト・ペーパー
2012年7月

Oracle Process Acceleratorsによる プロセス・エクセレンスの実現

ベスト・プラクティスを使用したBPM価値の迅速な実現

免責事項

以下の事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント（確約）するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクルの製品に関して記載されている機能の開発、リリース、および時期については、弊社の裁量により決定されます。

概要.....	3
BPMの進化：成功からプロセス・エクセレンスへ.....	3
BPMプロセス・エクセレンス実現への道.....	5
オラクルの答え：Oracle Process Accelerators.....	6
Oracle PAのベスト・プラクティス："内容"の検証.....	12
Oracle PAのベスト・プラクティス：プロセス、サービス.....	13
<i>BPMN</i> パターン：トップダウンのビジネス分解.....	13
<i>BPMN</i> パターン：事前構築のルーティング・パターンの利用.....	15
<i>BPMN</i> パターン：ビジネス・ルールによるビジネスの微調整.....	16
<i>BPMN</i> パターン：ビジネス・インジケータで開始、ダッシュボードの取得.....	18
サービス・パターン：ビジネス・サービス・コントラクトの定義.....	20
Oracle PAのベスト・プラクティス：	
ユーザー・インタフェース・コンポーネントのアーキテクチャ.....	21
表示パターン：すぐに使用できるOracle BPMのHuman Tasks とHuman Workflowへの接続.....	22
表示パターン：ページ・フラグメントの再利用.....	23
コントローラ・パターン：ページ・フローの再利用.....	24
モデル・パターン：アプリケーション・レイヤーへのビジネス・ロジックの配置.....	25
すべてを統合：卓越したOracle PA再利用パターンでのTRM UI.....	27
パターン：アプリケーション・コンポーネントのモジュール化.....	29
結論.....	30

概要

Oracle Process Accelerators (Oracle PA) は組織がプロセス・エクセレンスをより迅速に実現できるように支援します。Oracle Process Acceleratorsは、Oracle Business Process Management (Oracle BPM) Suite 11gで開発されたビジネス・プロセス・ソリューションです。Oracle Process Acceleratorsは、そのまま展開することも、顧客固有の要件に合わせて拡張することもできます。Oracle Process Acceleratorsは、BPM展開の価値実現を促進するだけでなく、Oracle BPM Suiteを使用してプロセス主導型ソリューションを策定している組織のベスト・プラクティスを具体化し、ブループリントとなります。Oracle Process Acceleratorsを採用する組織では、展開するOracle Process Acceleratorsの対象となるビジネス・プロセスを改善できるだけでなく、Oracle Process Acceleratorsのベスト・プラクティスとパターンを適用することで、プロセス管理のイニシアチブを低リスクでより速く成熟化させる、またとない機会が得られます。

BPMの進化：成功からプロセス・エクセレンスへ

BPMが一般の企業に受け入れられるものとなり、採用が急速に増加していることから分かるように、組織では、ビジネス・プロセス管理 (BPM) ソフトウェアを使用する価値について、ますます認識が高まっています。組織のBPMへの投資目的は、効率性の向上、ビジネスの可視性、および俊敏性の獲得です。最近の傾向では、組織が下位のインフラストラクチャ・レイヤーから上位の抽象化レイヤーに焦点を移しており、BPMの成長が加速しています。この焦点の変化は、パッケージ・アプリケーションの境界内で構成するだけでサポートされるコモディティ化されたプロセスを超えて、プロセスをエンド・ツー・エンドで実行するというニーズによるものです。BPMは事実上、強力ながらも柔軟なアプリケーションを含んでおり、特定のビジネス・プロセスで実行するように最適化されながらも十分に拡張可能で、他のアプリケーションにプラグ・アンド・プレイできるという、将来への橋渡しとなるものです。

運用効率を高めるためにBPMモデルに投資することを決定したら、組織では、1つのイニシアチブのROIでのみ成功を測定するのではなく、長期にわたって一貫して成果をあげるという新しいパラダイムを取り入れる必要があります。結局のところ、BPMの目的は、1つのプロセスを解決するだけでなく、複数のプロセス管理イニシアチブで効率的に実行され、増加するビジネス・プロセス・ポートフォリオに対しても適切に提供される、"プロセス・エクセレンス"を実現することでもあります。

プロセス・エクセレンスを実現するための決定要素は何でしょうか。変更管理、リーダーシップの支援、およびBPMモデル開始へのビジネス関与の重要性には、大きな関心が払われています。

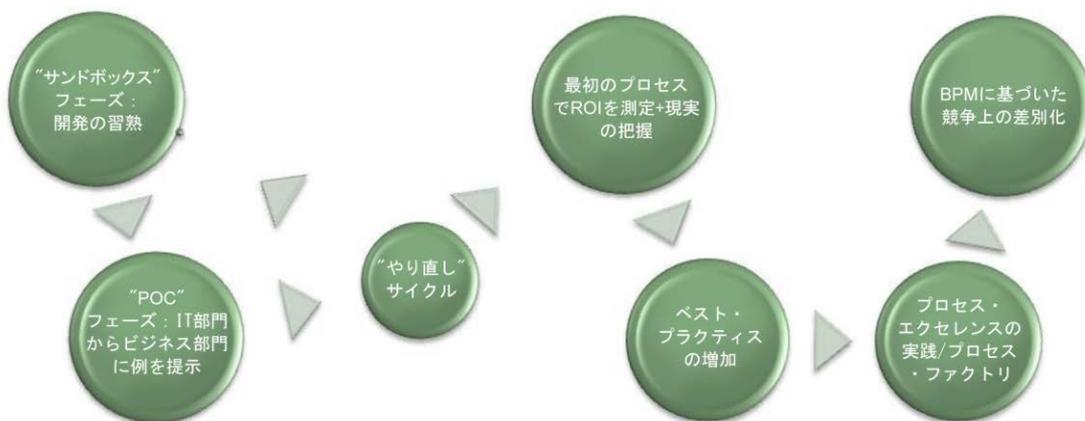
ここでは、BPMを導入後、高スループットのビジネス/IT共同提供モデルの実現に向けてどのような方法で組織を支援するかに焦点を当てて説明します。以下に、目標とする最終結果を示します。これらすべてが、プロセス・エクセレンスの成功要因となります。

- 企業が価値創出への要件であるBPM生成モデルを習得している。"白紙"の状態から開始するよりも、アプリケーションを確認してアプリケーションの構成方法を提示するほうが簡単です。BPMは通常、パッケージ・アプリケーションでは標準化されていないプロセスを対象としています（プロセス内の一部のステップが標準化されている場合を含む）、企業は、事前構築されたソリューションの効率性と反復的なBPM提供モデルの柔軟性を組み合わせたモデルを採用することによって、視覚化からソリューションに効率的に移行できます。
- ITスタッフがBPMのスキルを習得している。BPMを検討している企業では、長期的な社内能力開発に本格的に取り組む必要があることを戦略的に理解しています。ほとんどのIT組織では、Javaや.NETのスキルを持つ既存のリソースを活用し、高い生産性と協調性を備えたBPM専門家となるよう、トレーニングを提供することを考えています。パターンを確立すると、手続き型モデルやオブジェクト指向モデルからビジネス・プロセス指向モデルへの移行を促進できます。これらのパターンは、IT部門が開発期間を短縮するのに役立つだけでなく、ビジネス・プロセスの設計にフローチャートではなくBPMNを使用するなど、ビジネスとの連携にも役立ちます。
- 外部と内部のビジネス・ステップを1つのエンド・ツー・エンドのプロセスに統合する。BPMは孤立したものではありません。ビジネスを変えていくことを目的としたソリューションでスタンドアロンのものはほとんどありません。統合は、BPMプロジェクトの遂行においては非常にコストの高い部分であり、特に多くの可変部分を伴う反復的な改善において高いコストがかかる可能性があります。BPMとSOA戦略を完全に統合していることが必須であり、アプリケーションを含めた、開始点となる事前統合のシナリオを用意しておくことも必須です。

上記の実現要素に目を向けると、ベスト・プラクティスの提供モデルに基づいて事前構築のソリューションを（すべてのレベルのBPM成熟度の）組織に提供するBPMベンダーが必要であることが明らかです。組織では、これらの事前構築のソリューションと、共通のビジネス/ITモデルを確立するために提供されるBPMソリューションを反復的に改善する手法を利用できます。

BPMプロセス・エクセレンス実現への道

どのようにすれば組織は、プロセス・エクセレンスを短期間で実現し、"BPMのファクトリ"（BPMを効率的に活用する組織）となり、成功を繰り返し提供できるでしょうか。この質問に答えるため、オラクルでは顧客、パートナー、およびアナリストから情報を収集するとともに、次世代ソリューションの提供やOracleアプリケーションの構築にBPMを使用しているオラクル社内チームからも情報を収集しました。以下の図に示す成熟の段階を通過している組織で、さまざまなパターンが見られました。



通常、プロセス・エクセレンス実現への道は一直線ではありません。組織では、BPMの利用を徐々に習得し、多くの場合、上記の"やり直し"のサイクルを通過します。この"やり直し"のサイクルで、複雑さを低減したりエンタープライズ・クラスのソリューションや一貫した成功を提供したりするには、より優れたモデルが必要であることに気付きます。ほとんどの組織では、最終的に、BPMを使用して成功を実現し、一貫したROIを創出し、"プロセスのファクトリ"となるよう続行し、競争上のさらなる差別化を実現します。

ビジネスとITが常に連携し、ビジネスが要件を反復的に提起してITがアーキテクチャ・パターンを注入し、本番環境の設計を行って、既存の資産を再利用する場合に、反復的なプロセスがBPMに非常に適しています。BPMベンダーは、このような反復的かつアーキテクチャ的な提供モデルの実現に向けてビジネスとITが進行を加速するのをどのように支援できるのでしょうか。

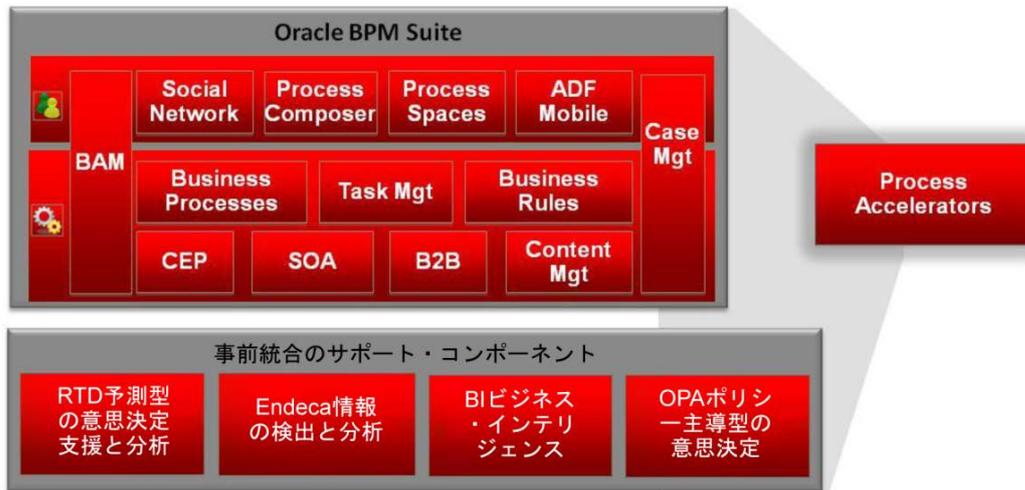
オラクルの答え：Oracle Process Accelerators

プロセス・エクセレンス実現への道を分析した結果、オラクルでは、強力なBPM製品機能をIT (Oracle BPM Studio) とビジネス・ユーザー (Oracle BPM Composer) の両方に追加しました。また、もつとも成功している組織では、BPMの習得レベルが高く、業界の重要なビジネス・プロセスを果敢に対象としたり、業界で持続可能な差別化を推進したりしていることが分かりました。

オラクルにとっての課題は、プロセス・エクセレンスを実現するまでの顧客の道のりを短くし、すべての顧客がOracle BPM Suiteを最大限活用できるようにすること、つまりすべての顧客に"最悪な状況を招かずに最先端のソリューション"を提供することでした。これに対する答えは、BPM用のOracle Process Acceleratorsに重点的に投資することでした。



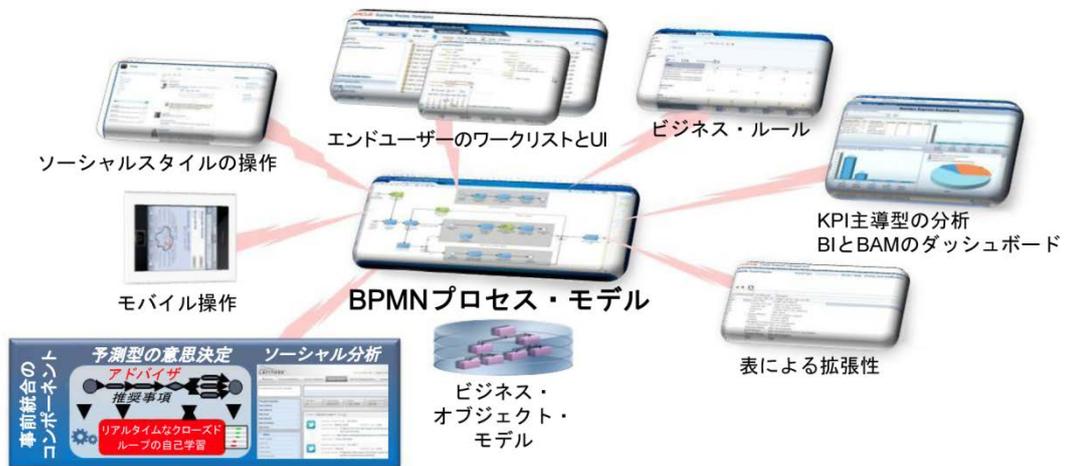
Oracle Process Accelerators は、オラクルが開発し、テストを行ったビジネス・プロセス・ソリューションであり、Oracle BPM Suite の上に構築されています。Oracle BPM には、モデリング・ツールとシミュレーション・ツール (Oracle BPM Studio、Oracle BPM Composer、Process Simulation)、Oracle Business Rules、Oracle Human Tasks と Oracle Human Workflow、Oracle Business Activity Monitoring (Oracle BAM)、コラボレーション用の Process Spaces (Oracle WebCenter の上に構築) が含まれています。Oracle BPM は Oracle Service-Oriented Architecture (Oracle SOA) の上に配置され、Oracle Application Development Framework (Oracle ADF)、Oracle Fusion CRM などの数多くの Oracle アプリケーションでも使用されているマルチチャネルの JSF ベースの UI フレームワーク、および統合コンテンツ管理ソリューションを利用します。



一連の非常に強力なツールが密接に統合されています。また、近い将来にビジネス・プロセスで必要な場合は、Oracle PAは、Oracle Real-Time Decisions (Oracle RTD)、Oracle Endeca、Oracle Policy Automationなどの事前統合のコンポーネントと、Siebel CRMなどの他のOracleアプリケーションで提供されるサービスの両方を利用します。完全なソリューションを迅速に提供するためには、これらすべてが不可欠です。

概して、Oracle BPMは不可欠な機能がすべて含まれた、効果的で非常に強力かつ完全なセットを提供します。ただし、強力な機能の利用には責任が伴います。その強力な機能を利用するには制御が必要であり、真に柔軟性に優れ、再利用可能で、迅速なソリューションを組み合わせるには、各ツールをいつ使用すべきか熟知している必要があります。このような場合に役立つのが、ビジネスとITが協力して目標を達成するために使用できるモデルであるOracle PAのベスト・プラクティスとパターンです。このことは、IT部門が新しいBPMツールとパラダイムに適切なアーキテクチャと実用的なモデルを採用する場合にも役立ちます。

Oracle PAは、顧客要件にエンド・ツー・エンドで対処するために、Oracle BPMソリューションのさまざまな機能を優れた方法でいかに統合するのかを実証しています。以下の図は、Oracle PAフレームワークに含まれる主要なコンポーネントをいくつか示しています。



Oracle PAの基本方針は、すぐに使用できる上に、拡張性も考慮して設計されていることです。Oracle PAはそれ自体で価値あるソリューションですが、BPM実装に広く使用できるベスト・プラクティス・パターンの参照フレームワークとしても機能します。

Oracle Process Acceleratorsは、業界特有の重要なプロセスと、あらゆる業界に共通するビジネス・プロセスの両方をターゲットにしています。"BPMプロセス・エクセレンス実現への道"を探究しているときに、組織がビジネス・モデルの中核であるプロセスを対象を絞った場合に、BPMでいかにしてROIを実現しているのかを見てきました。ただし、ITとビジネスの両方をそれほど複雑でない事業と連携させる前に、このようなプロセスに即座に対象を絞ることはリスクを伴うことでもあります。業界に焦点を合わせたOracle PAでは、ベスト・プラクティスを組み込むことで、プロセス・エクセレンスを実現する過程で困難に直面する可能性が少なくなるため、基幹産業のプロセスに迅速に対処できます。

Oracle PAによる顧客価値の提案

- **持続的価値の実現にかかる時間**：プロセス・エクセレンスを迅速かつ低リスクで実現
- **最適に適合**：事前構築のソリューションを再利用し、一意の要件に合わせて拡張
- **プロセスの最適化**：KPIを監視して改良
- **ベスト・プラクティス**：組込みの設計パターンを利用して効率的な変更管理を実現
- **コラボレーションの実現**：共通の提供モデルを中心にビジネスとITを連携
- **BPMの価値の拡大**：ビジネスの複数の要素を自動化 – 業界固有の要素とあらゆる業界の要素

オラクルでは、もっとも実用的で役立つ将来のOracle Process Acceleratorsの候補を継続的に評価しています。これらのソリューションは、業界の重要なプロセスのビジネス要件に対応し、オラクル、設計パートナー、および顧客の専門家の経験を組み込んでいます。実施している組織では、固有の要件に応じて業種別のOracle PAを拡張および変更し、独自の"秘術"を追加したりベスト・プラクティスを差別化したりできます。結局のところ、Oracle BPMの利点は柔軟性であり、Oracle PAは進化するように設計されています。

「オラクルの顧客の場合、Oracle Process Acceleratorsを展開するというオプションにより、Oracle BPMのメリットをさらに多くの従業員に拡大し、開発期間を短縮できるため、11gの展開で得られる潜在価値が高まります」

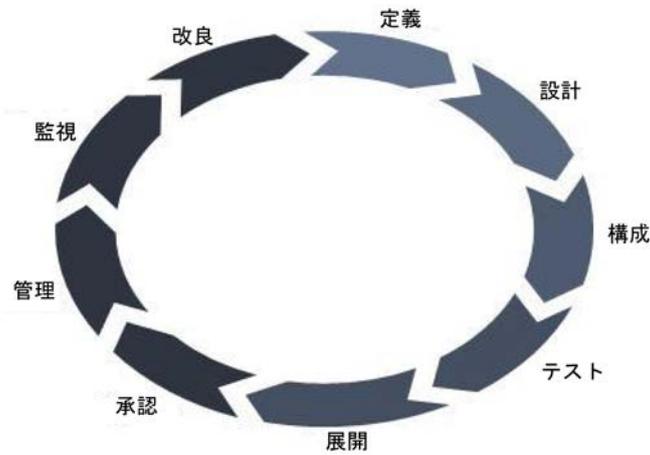
Ovum、Somak Roy (O100125-024)

Oracle PAを大きく変更する場合でも、顧客の時間と労力が大幅に削減されます。一部の業界のOracle PAはスタンドアロン・ソリューションですが、標準のビジネス・オブジェクトを使用する汎用のインターフェースを介して、多くのOracle PAが既存のOracleアプリケーションと統合されています。

基幹産業のプロセスに加えて、多くの組織で一貫してリソース浪費の原因を改善するために投入するほど重要ではない、数多くの水平型のビジネス・プロセスが存在します。これらのプロセスは引き続き電子メールやスプレッドシートを使用してその場限りで実行されるため、組織でコストが増加し、生産性が低下し、可視性と俊敏性が制限されます。例を挙げると、出張申請管理、ドキュメントの転送と承認、内部サービスのリクエストと提供、新入社員研修などがあります。

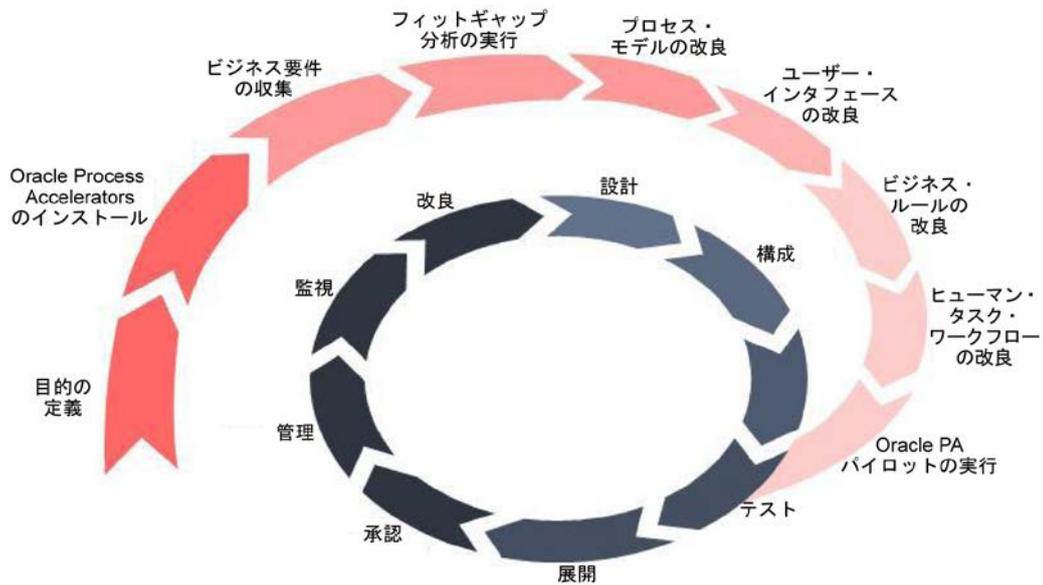
水平型のOracle PAはこれらのプロセスを対象としており、採用する組織では、複雑度や重要度の低い（したがってリスクの低い）プロセスを対象とすることで、Oracle BPMへの投資を活用すると同時にOracle BPMソリューションの提供経験を増やすことができます。水平型のOracle PAでもOracle PAベスト・プラクティスが組み込まれるため、バランスの取れた効率的なIT/ビジネス共同提供モデルを組織がセットアップするのに役立ちます。

以下に、このようなプロセス・ソリューションの提供サイクルを示します。



この提供サイクルは、対象となるビジネス・プロセスの特定と定義で始まり、BPM の構成（必要なすべてのサービスと UI を含む）、テスト、および展開が含まれます。特定の段階に関係なく、このサイクルの重要な概念はサイクルであることです。プロセス・エクセレンスを必要とする組織では、ビジネス・プロセスの KPI とメトリックを測定し、ビジネスのパフォーマンスを監視し、プロセスを改良し、再定義、再設計、および再展開することが求められます。1 回のサイクルでは終わりません。業界内で競争力を発揮するのに不可欠なコア・ビジネスのプロセスに対処できるよう、提供モデルは迅速かつ強固である必要があります。

Oracle PA は、顧客が効率的な IT/ビジネス提供モデルを迅速にセットアップするための第一歩でもあります。以下の図は、組織が Oracle PA を使用することで提供モデルをいかに促進できるのかという概念を示しています。



上記のモデルでは、顧客がプロセスの範囲に適切な Oracle PA を選択すると、顧客のビジネス要件を Oracle PA の要件範囲と即座に比較できます。ビジネス・ユーザーが Oracle PA の機能に"触れる"ことができるよう、1つまたは複数の Oracle PA をインストールします。優先順位に基づいて KPI、ビジネス・ルール、ヒューマン・ワークフローを改良し、展開前に本番環境のパイロットを実行できます。顧客で再利用されるすべての要素がベスト・プラクティスに基づいて構成され、拡張性を考慮して設計されます。その結果、組織では、効率的な BPM モデルに不可欠なベスト・プラクティスが自然に採用されます。

Oracle PAのベスト・プラクティス：“内容”の検証

BPMですぐに生産性を向上するには、Oracle PAのアーチファクトを利用することが重要です。確かなアーチファクトとともに、組込みのベスト・プラクティスとステップ・バイ・ステップの手法を利用すると、“最悪な状況”となるリスクが大幅に軽減されて、BPMの“初心者”から最先端に迅速に移行できます。

ここまで、実践ではなく理論について説明してきました。以降では、Oracle PAの内容を詳しく説明し、具体的なベスト・プラクティスを示します。水平型のOracle PAであるOracle Travel Request Management (Oracle TRM) を、このホワイト・ペーパーのプラクティス例の事例としておもに使用しています。

以下に示すベスト・プラクティスのいくつかは、要約すると、“ジョブに最適なツールの使用”であると言えます。結局のところ、ベスト・プラクティスは、経験のあるアーキテクトだけがビジネス要件の背後で見分けるような、非常に複雑なパターンの取得に関するものだけではありません。Oracle BPM Suiteなどの一式のツールセットを使用する場合は特に、最小限の労力で最大限の成果を得るための戦略を簡単に特定することが、ベスト・プラクティスと見なされます。これらのベスト・プラクティスは、ビジネスとITの両方に役立つ製品を使用した結果の達成に経験を組み込むものです。

ここに示されているベスト・プラクティスは、BPMとSOA関連のパターンのプロセスおよびサービスと、ユーザー・インタフェース関連のパターンのプロセスおよびサービスの2つのグループに分類されます。Oracle PAの提供時にはさらに多くのパターンが使用されますが、これらの2つのグループから、BPM実装のすべての要素を制御することがいかに重要であるかが分かります。BPMソリューションは、数あるコンポーネントの中でも特にプロセス、サービス、およびユーザー・インタフェースで構成され、完全なBPMスイートではすべてのコンポーネントを容易に提供する必要があります。

Oracle PAのベスト・プラクティス：プロセス、サービス

2つの領域の最初のパターンについて説明します。

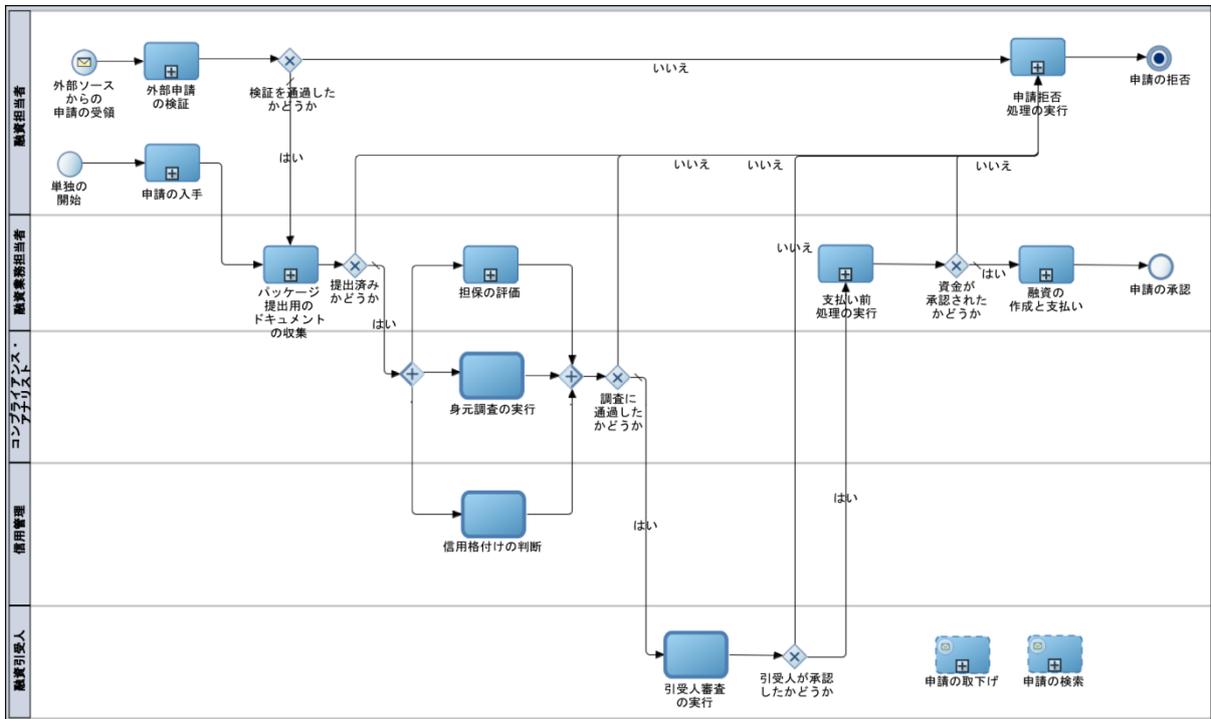
- *BPMN*パターン：トップダウンのビジネス分解
- サービス・パターン：ビジネス・サービス・コントラクトを定義、*BPMN*で使用

*BPMN*パターン：トップダウンのビジネス分解

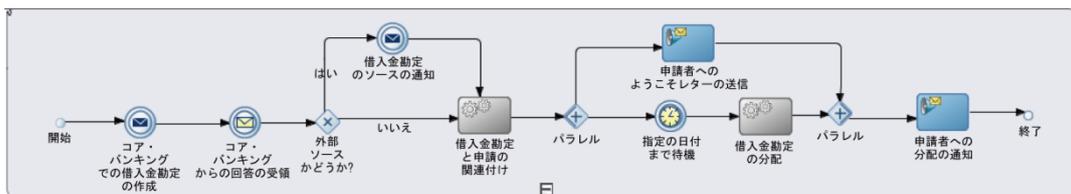
オラクルでは、Oracle Process Acceleratorsでビジネス・プロセスをモデリングするための具体的な標準を定義しました。*BPMN 2.0*には、プロセス・モデリングの3つの"適合クラス"である記述適合クラス、分析適合クラス、および共通実行可能クラスがすでに含まれています。これらのクラスのどの要素と属性が*BPMN 2.0*準拠のツールでサポートされている必要があるのかについては仕様に定義されています。適合クラスは、最上位レベルの記述適合クラスから最下位レベルの実行可能クラスまで、さまざまな抽象化レベルとして示されます。具体的には、記述適合クラスと分析適合クラスでは、図に示されている要素と属性に焦点を置いており、実行におもに使用される非表示の要素は除外されます。共通実行可能クラスは、プロセスを実行するために必要となる要素と属性を追加します。

オラクルでは、Oracle PAの実装時に*BPMN 2.0*で開始し、高度なモデリング表現の使用で生じる自由度をさらに制限するルールを追加しました。これらのルールには、ネーミング規則、記号の使用、共通パターン、レイアウト、例外処理、および他の重要なトピックが含まれます。ここでは、具体的なルールではなく原則に焦点を絞り、特に、要件から本番環境までのビジネスとITの同期に役立つ、ビジネス・レベルと記述モデルの原則について説明します。

- **最上位の*BPMN*モデルを完全にビジネス関連で維持する。**ビジネス・プロセスのトップレベルの記述にビジネス・ロジック以外のロジックを取り入れる理由はありません。*BPMN*フローがいかにも複雑でも、まずビジネス用語で記述されている必要があります。これは、単純な原則です。ただし、一般に、*BPMN*でITの自由度が高くなる場合は、最上位のモデルに機能面を含める傾向があり、実行面を含めることもあります。コンテンツが失われることを心配する必要はありません。詳細は後で対処でき、必要なレベルで文書化できます。トップレベルのビジネス・フローのテストでは、基盤となるビジネス・プロセスのオーナーとなる役員にこのフローを安心して提示できるかどうかを考えます。過度な場合は、サブプロセスに詳細を隠すなどして抽象化する必要があります。以下に、ビジネス・ユーザーが非常に理解しやすい、融資組成のトップレベルの*BPMN*フローを示します。



Oracle BPM 11gは、ビジネス・モデリング・ツールOracle Business Process ComposerとIT指向のツールOracle BPM Studioが含まれているため、トップダウンのビジネス主導型のプロセス・モデリングに非常に役立ちます。Oracle BPM 11gはサブプロセスに関してBPMN 2.0標準をサポートしているため、ビジネス・レベルから実行可能なITレベルに容易に分解できます。分解の例として、上記のBPMNプロセスの融資の作成と実行タスクの詳細を以下に示します。



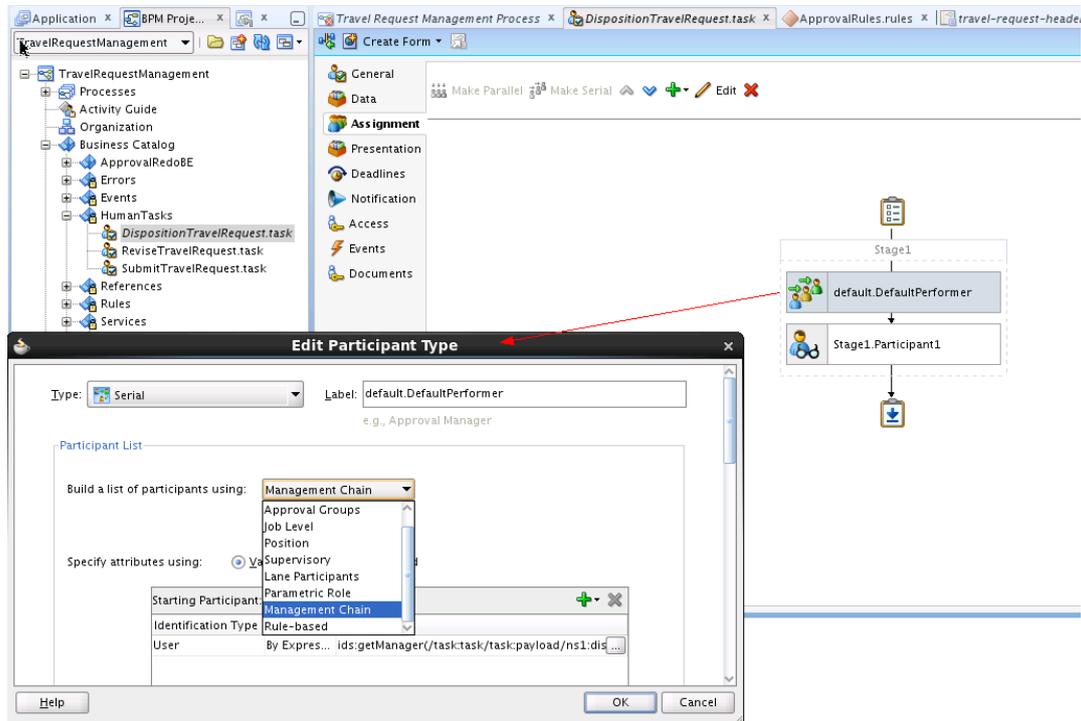
コア・バンキング・システムとの統合などの技術的な詳細は、ここで指定します。

- ゲートウェイはビジネス・ルーティング用である。この原則は、上記の原則から派生したものです。BPMNのゲートウェイの目的は間違って解釈されることが多く、フローチャート内で決定事項のように見えます。そのため、変数値が配置されている場所がチェックされ、これに基づいてif-then-elseの決定が行われています。しかし、BPMNモデルはフローチャートではありません。BPMNの上位レベルのゲートウェイでは、ビジネス条件がチェックされます。基盤となる変数は後で定義されます。ゲートウェイがトップレベルに属するかどうかを適切にテストするには、さまざまなビジネス・ユーザーやロールで実行されるさまざまなタスクにそのゲートウェイがつながっているかどうかをテストします。
- *最初に、ビジネス・オブジェクトを把握する。*オラクルの経験では、高レベルのビジネス・プロセスを設計してこれらのプロセスを伝える最適な方法は、何よりもまず、ビジネス・フローの中心となるビジネス・オブジェクトを把握して記述することです。結局のところ、BPMNプロセスは、Incident Report、Banking Loan、Travel Requestなどのビジネス・オブジェクトに基づいて機能します。ビジネス・オブジェクトとは何かを理解していると思っている場合でも、プロセスを細部まで設計してビジネス・オブジェクトの変換を進める前に、一般的な定義を使用し、キー属性をサンプル値とともにリストアップすると非常に役立ちます。このことは、ビジネス・オブジェクトによってビジネス・プロセスが決まるケース管理シナリオでさらに重要になります。

BPMNパターン：事前構築のルーティング・パターンの利用

BPMNモデルには、自動化されたシステム・タスクとヒューマン・タスクの両方が含まれます。ヒューマン・タスクには、承認、拒否、詳細情報の提供などのアクションが含まれます。複雑なヒューマン・タスク・ワークフローの設計が複雑になる要因の1つとして、そのタスクで作業する次のユーザーは誰なのかを判断することが挙げられます。組織構造の平均的な複雑さは増しており、たとえば、特にマトリックスのレポート様式を採用している場合に、複数の部門と各部門内の管理階層のために複雑さが増しています。これらの構造に基づいて割当てルールを動的に決定することが不可欠になっています。

Oracle BPM Suite では、Human Task スイート・コンポーネントで定義されているパターンに基づいて参加者のリストを動的に作成できるため、このタスクが容易になります。以下のスクリーンショットは、Human Task の Assignment タブ (TRM の DispositionTravelRequest) を示しています。

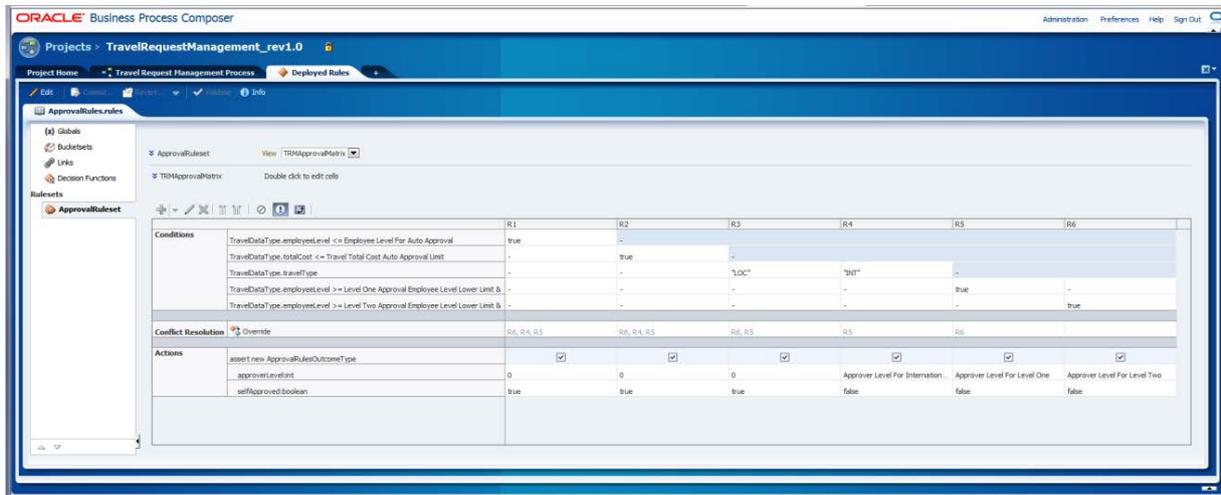


上記の例では、参加者のリストは"Management Chain"に基づいて作成されています。また、参加者タイプが"Serial"になっており、承認はパラレルや1つ上のレベルだけ ("Single") でなく管理チェーン順に上方向に進みます。また、タスクの実行時にプロセス情報に基づいて、ビジネス・ルールを使用して参加者のタイプのリストを動的に選択する必要がある場合は、"Rule-based"を選択できます。

このOracle BPMパターンは不可欠です。これを使用しない場合は、BPMフロー内で複数の組織構造にまたがってロジックを作成する必要があるため、ビジネス・プロセスというよりもフローチャートとなり、もっとも重要なこととして柔軟性に欠けます。

BPMNパターン：ビジネス・ルールによるビジネスの微調整

ビジネス・ルールがITの独占領域にならないようにします。変化するビジネス条件に基づいてプロセスの動作を動的に微調整するためには、ビジネス・ルールはビジネスの状況に委ねる必要があります。Oracle BPM Suite ではこの原則が採用されており、ビジネス・ルールはビジネス・アナリストが使用するために設計された Oracle BPM Composer に表示されます。以下のスクリーンショットはビジネス・ルールの例を示しており、具体的には、必要な承認レベルを決定するために TRM で使用される、意思決定表のビジネス・ルールを示しています。



複雑なビジネス・ロジックは、最上位のマトリックスの条件によって最下位のマトリックスで実行されるアクションが決まる、このような意思決定表に定義するのが適しています。たとえば、列"R4"では、出張のタイプが海外である場合、承認者のレベルは"Approver Level for International Travel"となり、この組織ではシニア・バイス・プレジデントとして定義されています。これが、TRMの最初のオラクル社内展開で構成されたポリシーとなっています。この社内展開で、オラクル自身がTRMを採用する最初の顧客となっており、Oracle PAの"有言実行"に対する明確なコミットメントとなっています。

良いパターンは、ルール内で決定を行い、その決定に基づいてプロセス内の異なるパスを選択するように、ルールの直後にゲートウェイを使用します。ビジネス用語で表した分析とルールを使用し、システム動作に基づいてルールを微調整することで、ビジネス・プロセスの改善が可能です。TRMでは、組織グループの予測出張コストによる現在までの経費など、分析メジャーに基づいて経費管理と承認のオーバーヘッドのバランスを取るために、"Employee Level for Auto Approval"、"Level One Approval Employee Level Lower Limit"、およびその他のレベルを調整できます。

	Name	Value
(x)	Employee Level For Auto Approval	2
(x)	Travel Total Cost Auto Approval Limit	200.00
(x)	Approver Level For International Travel	3
(x)	Level One Approval Employee Level Lower Limit	3
(x)	Level One Approval Employee Level Upper Limit	5
(x)	Level Two Approval Employee Level Lower Limit	6
(x)	Level Two Approval Employee Level Upper Limit	7
(x)	Approver Level For Level One	2
(x)	Approver Level For Level Two	3

プロセスの実行中にルールをWebブラウザで変更でき、結果をOracle Business Activity Monitoringにおいてリアルタイムで監視できます。ITの関与やプロセスの再展開は不要です。

BPMNパターン：ビジネス・インジケータで開始、ダッシュボードの取得

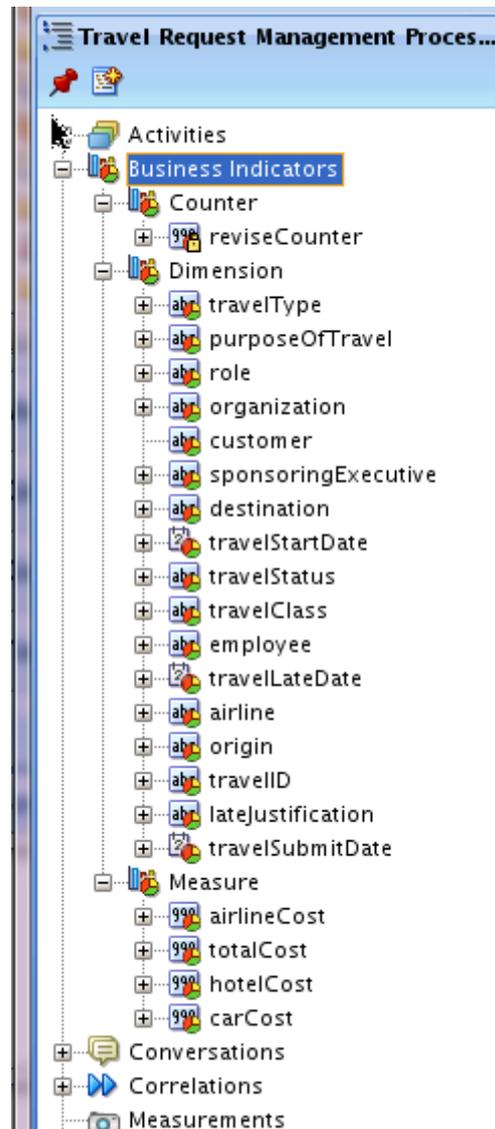
ビジネス・プロセスの動的な微調整は、ダッシュボードを監視するビジネス意思決定者が行います。ほとんどの場合、ダッシュボードはタスク・フロー・ロジックで展開するのではなく、後の反復時に対処します。ダッシュボードは導入後に自然に展開され、ユーザーは最初のバージョンを確認したときに、プロセス制御を改善する方法や、新しい要件を定義する方法が分かります。

"卵が先かにかわとりが先か"という問題がある一方で、新しい興味深いメジャーが見つかったら、ビジネスでは新しいメジャーの過去の履歴データを確認できます（その時点までのデータが取得されていない可能性もあります）。

厳しいダッシュボード要件を超えるOracle BPMの一連のビジネス・インジケータを定義し、Oracle BAMを利用してプロセスで生成されたデータを収集することで、Oracle Process Acceleratorsはこの問題に対処します。ビジネス・インジケータが定義されると、Oracle BAMによってデータ収集が開始されます。このデータは、ダッシュボードが具体化されるとすぐに表示されます。そのため、Oracle BAMのポイントアンドクリック方式のOracle BAMダッシュボード定義機能を使用して、これらのビジネス・インジケータを使用するようにOracle PAのダッシュボードを容易に拡張できます。

ビジネス・インジケータにはメジャー、ディメンション、およびカウンタがあります。

- メジャーは集計に使用されます。たとえば、部門、コスト・センター、マネージャなどによって集計された飛行機、ホテル、車、および総コストを管理するために、TRM PAで使用されます。
- ディメンションは、出張、日付、出発地、目的地、従業員、航空会社などの目的別に区分するために使用されます。
- カウンタはイベントの発生回数を追跡します。TRMでは、出張リクエストで行われた改定数を追跡するカウンタが定義されています。



Oracle BPM Suiteのドキュメントには、プロセス内のプロセス情報を追跡する他の強力な機能として、ビジネス・インジケータとメジャーの詳細情報が記載されています。

Oracle BPM Suiteの利点の1つに、Oracle BAMを備えていることが挙げられます。Oracle BPMプロセスからOracle BAMにフィードが実行されます。Oracle BAMには事前定義のプロセス・ダッシュボードが付属しており、このプロセス・ダッシュボードを使用して、ビジネス・アナリストは、Oracle BPMからOracle BAMにフィードされた一連のビジネス・インジケータに基づいて新しいダッシュボードをすばやく定義できます。付属している一連のビジネス・インジケータはすぐに使用でき、他のインジケータも容易に追加できます。すべてのインジケータが自動的にOracle BAMにフィードされます（必要な操作は"ゲートを開く"ことだけです）。

各Oracle PAでは、以下の原則に基づいて、範囲内のビジネス・プロセス用に設計された一連のインジケータとダッシュボードが追加されます。

- 必要なビジネス・インジケータを正しく特定して取得するために、ライフ・サイクルの早い段階でKPIとダッシュボード要件を特定する
- モデリング段階で、主要なイベントとマイルストーンを特定する
- Oracle BAMを使用して外部システムを利用する
- Oracle BAMのアラートを使用して障害を監視する
- 分析ツールを、運用監視に最適なOracle BAMと、履歴と傾向の分析に最適なBIと組み合わせて使用する
- 同じ一連のダッシュボードで異なるユーザーに対応せずに、ロール固有のダッシュボードを定義する。複数のタブに同じウィジェットが表示されても構わない。各ビジネス・ロールに適したフィルタを適用する

サービス・パターン：ビジネス・サービス・コントラクトの定義

Oracle BPMでは、各ノードとタスクで実行する処理について技術的な指示を組み込む必要があります。外部サービス呼び出すときに、技術情報のために、サービスで実際に何が実行されているかを把握するのが難しくなる場合があります。BPMNのタスクが既存の技術サービスに示されるだけになる可能性があります。これは、これらのサービスがBPMNのビジネス・タスクと同じ抽象化レベルでない場合にも当てはまります。ベスト・プラクティスは、新しいビジネス・サービスを既存の技術サービスの上に明示的に定義することです。これにより、BPMNには全体的な技術情報やデータ・モデルは示されずに、BPMNのタスクによって正しい情報がフィルタリングされます。"ビジネス・サービス・コントラクト"となる明示的に定義されたサービスを通じて、プロセスで必要になるものだけを指定する必要があります。このサービス抽象化レイヤーを導入する利点は、明確さだけでなく柔軟性もあります。基盤となるモデルが変更になった場合（属性の変更、サービスを提供するアプリケーション全体の置換えなど）、コントラクトによってビジネス・プロセスが安定します。サービス・コントラクトに下位互換性があるのが理想的です。下位互換性を容易に確保できない場合は、サービスのさまざまな"バージョン"を作成して古いサービス・コントラクトを配置したままにすることができます。

Oracle TRM Process Acceleratorでは、TRMModelと呼ばれる独立したビジネス・サービス・レイヤーが作成されています。一連のビジネス・サービスを作成すると、抽象化レイヤーが追加されて、本格的なSOAツールを使用して管理しない限り、制御が困難になる可能性があります。オラクルの戦略では、Oracle BPMはサービス指向アーキテクチャの上に構築されます。これにより、ビジネス・サービスの作成と使用が簡素化されます。BPMNタスク実装のポイントアンドクリックが、サービスとタスクを関連付ける、基盤となるバインディングに変換されます。

Oracle PAのベスト・プラクティス：

ユーザー・インタフェース・コンポーネントのアーキテクチャ

Travel Request Management (TRM) により、組織では出張リクエストの承認と追跡を管理できます。中心となるビジネス・オブジェクトは出張リクエストであり、承認を得るために従業員が申請します。承認フローは、出張のタイプ、従業員のレベル、出張の目的などのさまざまな要因に基づいて即座に決定されます。以下のスクリーンショットは、ビジネス・プロセスを開始するために従業員が入力する"Submit Travel Request"画面を示しています。

Submit Travel Request

Request ID: TR000046
 Status: New
 Employee ID: jcooper
 * Travel Purpose: Customer Visit
 * Travel Type: Domestic
 * Origin: SFO
 * Destination: EWR
 * Travel Justification: Strategy meeting
 * Trip Start Date: 7/31/2012
 * Trip End Date: 7/2/2012
 Submit Date: 7/16/2012
 Late Justification:

Sponsoring Exec: William Faulkner
 Customer: Best Co
 Reference ID:
 Submitted By: jcooper
 Traveler: James Cooper
 Title: Loan Agent 1
 Email: jcooper@emailExample.com
 Cost Center:
 Organization: BPM
 Org VP:

Cost Estimates (USD)

Airline	594
Hotel	280
Car	106
Total	980

Error
 Trip End Date before today's date
 Example: 10/11/2012

Submit one trip segment for a single round trip. Submit multiple segments for multiple destinations.

Segment	Origin	Destination	Start Date	End Date	Total Cost Estimate
1	SFO	EWR	7/31/2012	8/2/2012	980

図1：Submit Travel Request画面

上記の画面には、ヘッダー（出張リクエストの全般情報が表示されている領域）、出張セグメント（画面下部の表）など、出張リクエストのさまざまなコンポーネントが表示されています。また、上記の画面には、実行されているフィールド検証も表示されています。

Oracle Travel Requestは、複雑度が中程度のビジネス・オブジェクトです。ただし、Oracle BPMの柔軟性をUIレイヤーに拡張するベスト・プラクティスに基づいて、TRMアプリケーションのUIを実装する仕組みを示すために、上記のような外見上はシンプルなサポート・ユーザー・インタフェース (UI) を使用できます。これは、機能、柔軟性、およびTCOのレベルに影響します。

次に、この外見上はシンプルなUIの陰に隠れたベスト・プラクティスとパターンをいくつか説明します。これらをModel-View-Controller (M-V-C) モデルに基づいて調べ、もっとも可視的なビジネスの側面からコアのバックエンドの側面まで、V-C-Mの順で詳しく確認します。

表示パターン：すぐに使用できるOracle BPMのHuman TasksとHuman Workflowへの接続

ごく基本的ながらも不可欠なベスト・プラクティスとして、すぐに使用できるOracle BPM製品で提供されている機能を使用することが推奨されます。その場合、Oracle Business Process Workspaceを、3つのOracle PAであるOracle TRM、Oracle Document Routing and Approval (Oracle DRA)、およびOracle Internal Service Request (Oracle ISR) へのアクセス・ポイントとして使用します。Workspaceでロール別のすべてのタスクにアクセスできるため、再構築が不要です。

The screenshot displays the Oracle Business Process Workspace interface. The top navigation bar includes 'Tasks', 'Activity Guides', 'Process Tracking', and 'Standard Dashboards'. The main content area is divided into several sections:

- Applications:** Lists 'DocumentRoutingandApprovalProcess' and 'InternalServiceRequestProcess'.
- External Applications:** Includes 'Manage Travel Requests', 'Maintain Document Categories', and 'Manage Service Types'.
- Worklist Views:** Shows 'Inbox', 'My Work Queues', and 'Standard Views' (Due Soon, High Priority, Past Day, Past Week, Past Month).
- Reports:** Displays 'Task Status' with a bar chart showing counts for Assigned (9), Suspended, Completed (12), Expired, Withdrawn (2), Alerted, Errored, Deleted, and Info Requested. A 'Total' of 57 is shown.

The central 'My Tasks' table lists tasks with columns for Title, Number, Priority, Assignees, State, and Created. One task is highlighted: 'LATE Travel request for James Cooper Jun 15, 2012 Los Angeles USD590 Training' (ID: 202050, Priority: 1).

The detailed view for this task, titled 'Approve Travel Request For: James Cooper', shows the following information:

- Request ID:** TR000035
- Status:** Pending final approval
- Employee ID:** jcooper
- Travel Purpose:** Training
- Travel Type:** Domestic
- Origin:** San Jose
- Destination:** Los Angeles
- Travel Justification:** Take software class
- Trip Start Date:** 6/15/2012
- Trip End Date:** 6/15/2012
- Submit Date:** 6/11/2012
- Late Justification:** Need to take this one day course to utilize the info learn on critical project.

Cost Estimates (USD):

Airline	545
Hotel	0
Car	45
Total	590

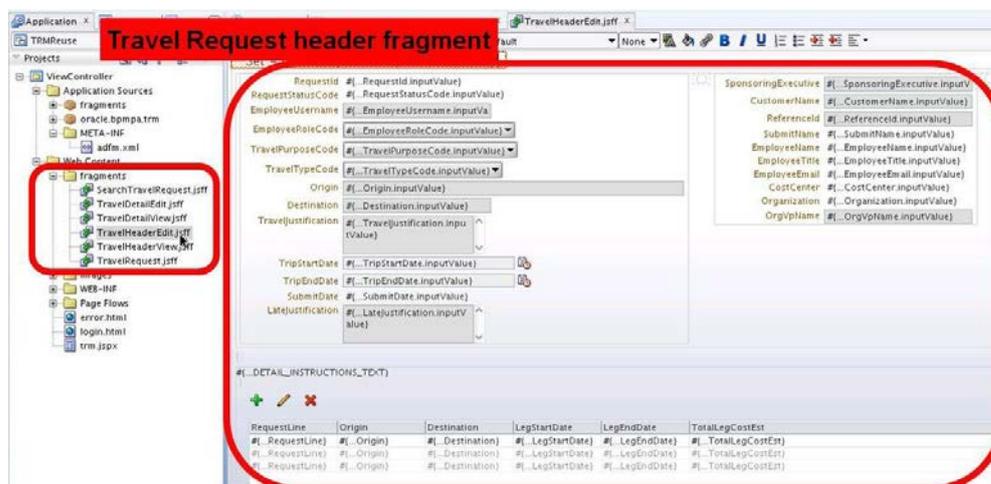
A table at the bottom shows the trip segments:

Segment	Origin	Destination	Start Date	End Date	Total Cost Estimate
1	San Jose	Los Angeles	6/15/2012	6/15/2012	590

ユーザーに公開される他のアプリケーションは、アプリケーション領域に追加されます。Oracle PAの新しいBPMNプロセス・インスタンスの場合、このアプリケーション領域でユーザーが新しいリクエストを開始すると、BPMシステムでビジネス・プロセス・インスタンスが開始されます。

表示パターン：ページ・フラグメントの再利用

ページ・フラグメントは、一度定義すると他のページで必要に応じて再利用される表示コンポーネントです。フラグメントを定義および使用してUIを再利用すると、UIの柔軟性と保守性が向上するとともに、実装コストが低減します。21ページの「図1：Submit Travel Request画面」では、出張リクエストを表示するすべてのUI画面にヘッダー・フィールドが表示されるため、出張リクエストの"ヘッダー"をページ・フラグメントにする候補としてすぐに特定できます。以下の図は、Oracle Application Development Frameworkのページ・フラグメントを示しています。

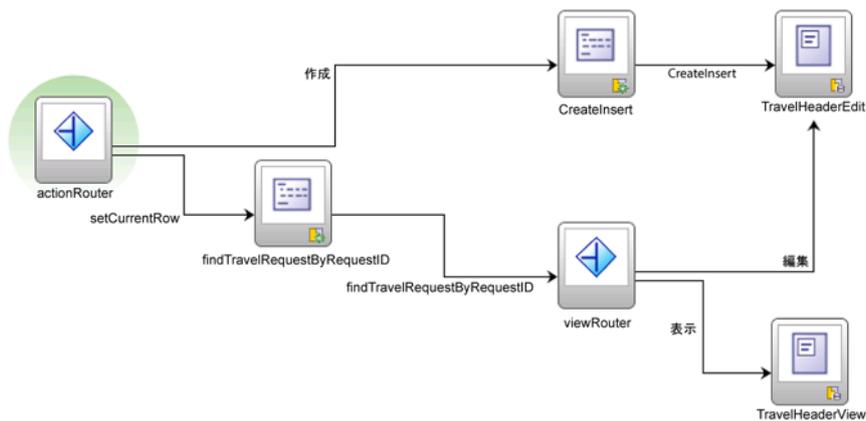


ページ・フラグメントをすばやく作成してドラッグアンドドロップで再利用可能な、Oracle ADFのようなUIフレームワークが必要なのか、またはOracle BPMの機能でよいのか、思い悩むかもしれません。オラクルでは、UI設計はプロセスの参加者が表示して作業する目立つ側面であるため、BPMの成功に重要であると考えています。ほぼ間違いなく、BPMソリューションほど、強力なUIフレームワークを適切に使用するソリューションはありません。特に、同じビジネス・オブジェクトが作成、エスカレーション、承認、拒否、展開、完了、再利用されたり、さまざまなビジネス・ユーザーがさまざまな時点で異なる許可レベルとアクセス権を持っていたりする、人間中心のソリューションでこれが当てはまります。数多くの画面が存在しますが、それらの画面に数多くの共通点があります。

コントローラ・パターン：ページ・フローの再利用

フラグメントなどのUIコンポーネントを再利用することは重要ですが、十分ではありません。Webアプリケーションには通常、複数のページがあり、次のページへのナビゲーションは即時に判断されます。たとえば、出張リクエストのページで「Submit」ボタンを押して新しい出張リクエストを作成したり、既存の出張リクエストをクリックしたりして詳細が含まれた表示専用画面にアクセスできます。

このすべてのナビゲーション・ロジックをコード化できる上、単にコード化するのが簡単に思えますが、これはベスト・プラクティスではありません。ITアーキテクトは、M-V-CモデルのControllerの部分は、おもにナビゲーション・ロジックの増加を防ぐために作成されたことを理解しています。BPMソリューションの開発時に、IT部門で"適切な処理"を簡単にするツールが必要です。以下のスクリーンショットは、この問題に対するオラクルの答えである"タスク・フロー"を示しています。



Oracle ADFでは、タスク・フローを使用してUIフローをグラフィカルに設計します。タスク・フローにページをドラッグアンドドロップしてから、矢印でページを接続できます（制御フロー）。必要なすべてのコードが基盤となるOracle ADFで生成されるため、コードを記述する必要がありません。

タスク・フローを定義したら、定義したタスク・フローをUIページにドロップできます。以下の図は、出張リクエスト・ヘッダーのタスク・フローをSubmit Travel Requestページにドロップして、出張リクエスト・ヘッダーのすべてのナビゲーションを継承する方法を示しています。Oracle ADFによって、必要なすべてのバインディングが即座に生成されます。

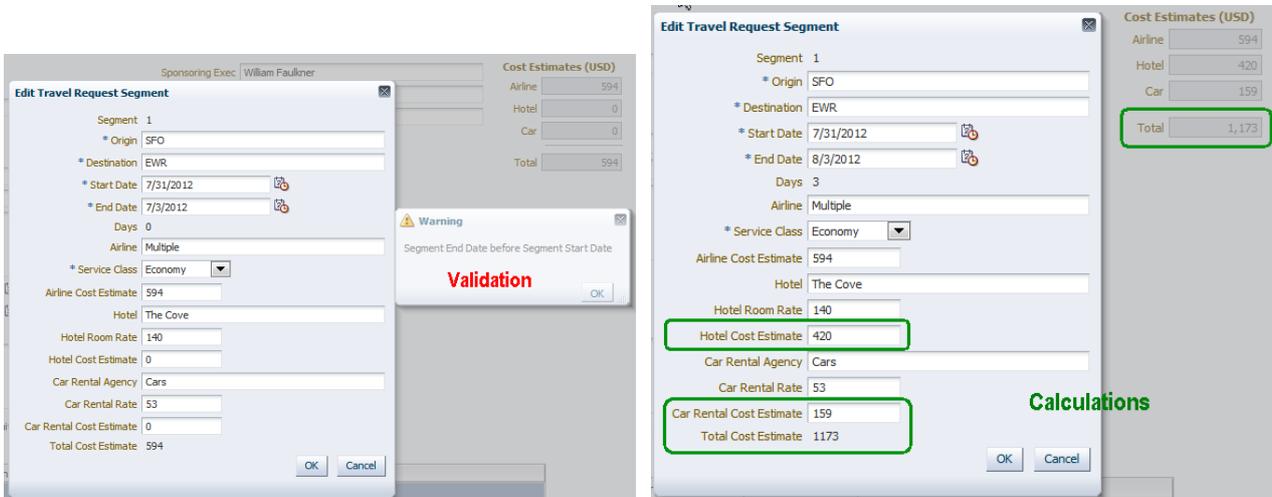


このスクリーンショットの右側に、ドラッグアンドドロップでページに追加できる、ボタンやパネルなどの従来のUIコンポーネントも示しています。TRMには作業用のUIがすべて含まれていますが、Oracle ADFを使用してTRMや他のPAを容易に拡張できます。ビジネスとITでは、追加のビジネス要件を満たすように機能を反復的に追加できます。また、Oracle BPM Composerを使用して、多くの構成をビジネス・ユーザーにそのまま提供することもできます。

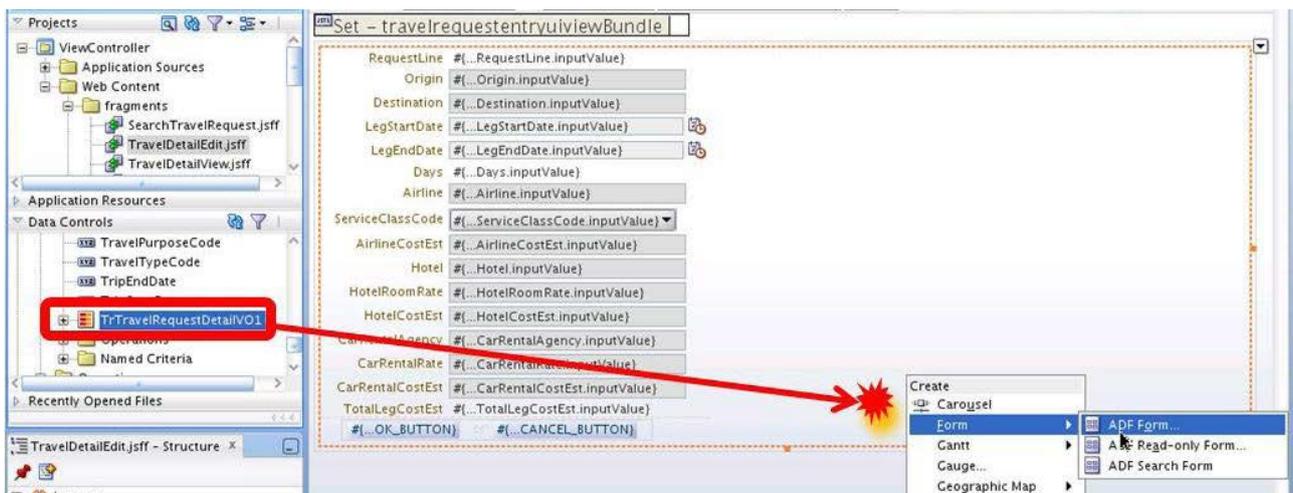
モデル・パターン：アプリケーション・レイヤーへのビジネス・ロジックの配置

以下の2つのスクリーンショットは、TRMで出張区間情報を収集するために使用されるポップアップ "New Travel Request Detail" で実行される、計算と検証を示しています。ユーザーがUIにデータを入力すると、検証と計算が実行されます。たとえば、終了日が開始日の前でないかチェックされ、日数と1泊料金に基づいてホテルのコスト見積りが計算されます。簡単ですが、適切に構成しないと十分に再利用されず、高いテスト・コストが発生し、柔軟性に欠けたものになります。この場合、コアのビジネス・オブジェクトの検証を実行すると、たとえば、Submit画面のCost Estimatesとポップアップの計算との間などで、ロジックの重複が生じます。

このシナリオでは、ビジネス・オブジェクトのロジックをビジネスの検証と統合するのが非常に適切です。ビジネス・オブジェクトが変更になった場合、検証を変更するのは1カ所だけで済むため、上位レイヤーのすべての場所で個別に変更を行う必要がありません。



Oracle BPM Suiteで用意されているツールOracle ADF-Business Component (Oracle ADF-BC) を使用すると、ビジネス・ロジックの追加と検証をDBスキーマ・レベルなどの"モデル"レベルで実行できます。Oracle ADF-BCでは、モデルやビジネス・オブジェクトの定義をポイントするだけで、データ・コントロールを作成できます。その後、ドラッグアンドドロップを使用して、これらのコントロールをUIページに追加できます。このような簡単な操作で、ビジネス・オブジェクトの属性に対応するUIのすべてのフィールドを作成できるだけでなく、これらの属性に関連するすべての検証も実行できます。以下のスクリーンショットは、出張リクエスト詳細を示すデータ・コントロールを、"New Travel Request Detail"ポップアップで再利用するページ・フラグメントのUIテンプレートにドロップする方法を示しています。結果（新しいADF Formの作成）がすでに示されています（フィールド、およびOKボタンとCancelボタンを含む）。

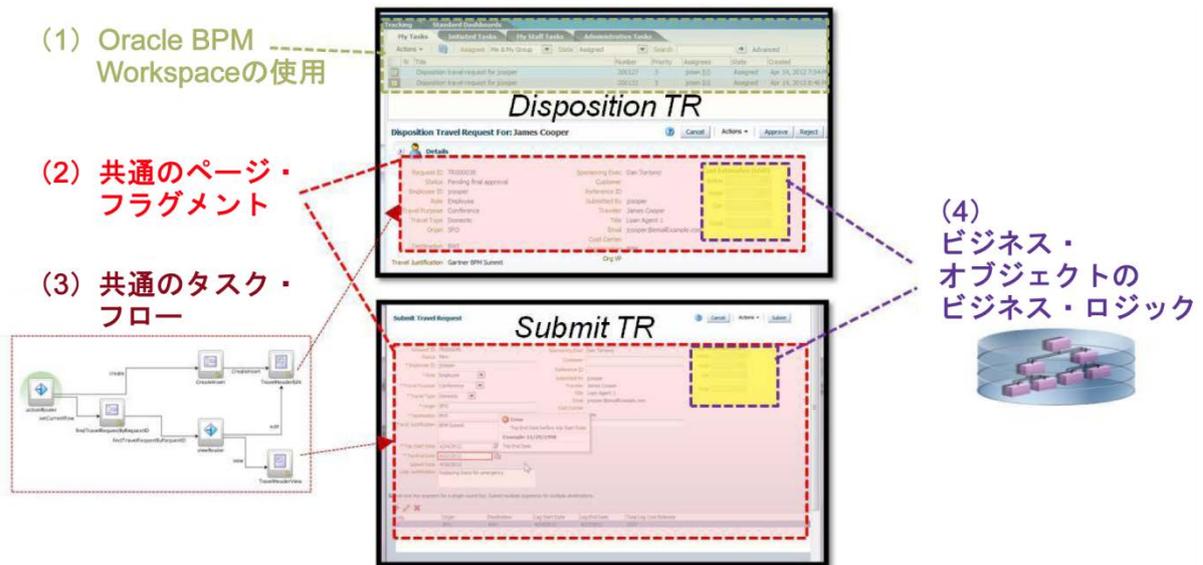


ページ・フラグメントによってすべての検証と計算が実行されます。また、ページ・フラグメントを、出張リクエスト詳細を処理するすべての画面に組み込むことができます。x個の画面でこの機能を使用している場合、UI実装の複雑さがx分の1に低減され、今後は新規作成、改良、および更新されるビジネス要件に基づいて、ビジネス・オブジェクトの定義を変更する回数の倍数分低減されます。UIとビジネス・ロジック間の基盤となるバインディングを設定し、本格的な開発者が空いた時間に参照できるようにXMLファイルに保存しました。

ビジネス検証をどこに配置するかは、再利用、近接性、パフォーマンスなどの数多くの要因によって異なります。Oracle ADF-BCなどの製品機能を利用すると、適切な選択を容易に行えます。基盤となるオブジェクト（データベース・テーブルなど）をポイントして出張リクエストをOracle ADF-BCでビジネス・オブジェクトとして定義したら、各属性にビジネス・ルールを定義できます。このシナリオの利点は、関連付けられたすべてのUIでビジネス・オブジェクトの検証が継承されることです。後で検証の追加が必要になった場合は、ビジネス・オブジェクトに追加するだけです。Oracle PAでは、固有のビジネス要件に基づいてビジネス・オブジェクトを拡張する方法について、具体的な例を数多く提供しています。

すべてを統合：卓越したOracle PA再利用パターンでのTRM UI

TRMのUIに戻って、具体的な再利用のベスト・プラクティスとパターンについて概要を説明します。以下の図は、TRMの2つの画面"Submit Travel Request"と"Disposition Travel Request"を示しています。"Submit Travel Request"は従業員が今後の出張を報告して承認を要求するために使用し、"Disposition Travel Request"はマネージャが承認、拒否したり、詳細情報を要求したりするために使用します。Disposition Travel Request画面には、Submit Travel Request画面のほとんどのフィールドが読取り専用で表示されます。Disposition Travel Request画面を完了するには、Approve、Reject、またはRequest Changeのボタンをいくつか追加するだけです。この2つの画面は、Oracle BPMのごく一般的なパターンである、同じビジネス・オブジェクトOracle Travel Requestで機能する別々のステップであるため、非常に似ています。



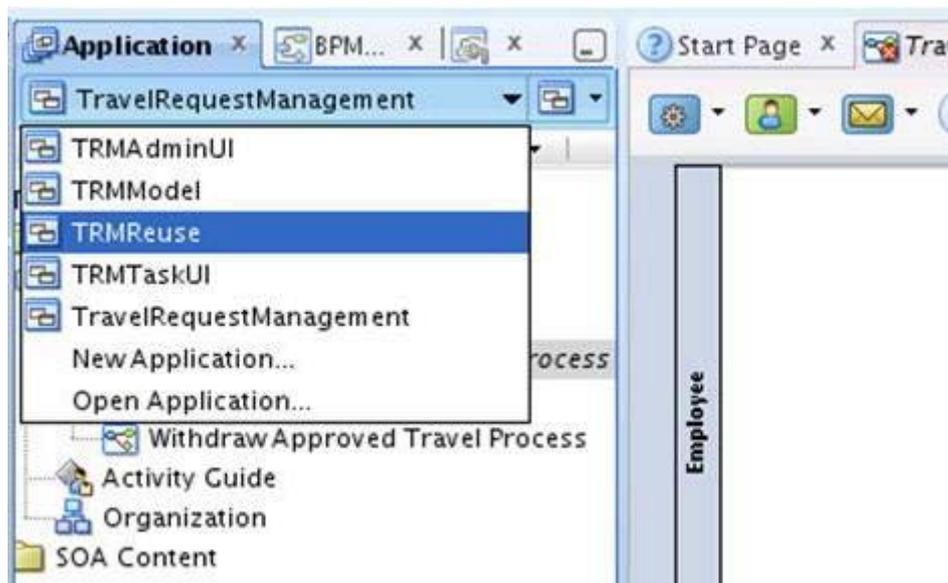
パターンなしでDisposition Travel Request画面を実装すると別のコンポーネントが作成されるため、切り取りと貼付けでコードを再利用するのが最適です。特にビジネス要件の変更時に、変更を多くの場所に適用する必要がある場合は、数多くの問題が発生します。代わりに、Oracle PAでは、中でも以下のパターンを利用するモデルが使用されます。

- 1) ビジネス・ロールに基づいて、Oracle BPM Workspaceで両方の画面にアクセスできる
- 2) 共通のページ・フラグメントを再利用する（赤で強調表示されている領域を参照）
- 3) 共通のタスク・フローにより、さまざまなページや領域でナビゲーション・ロジックを再利用できる（表示用のページと編集用のページで再利用するなど）
- 4) 検証と計算を基盤となるOracle ADF-BCオブジェクトで1度だけ定義する（黄色の強調表示を参照）

上記のようなパターンは、Oracle BPMコンポーネントの上に柔軟なソリューションを実現し、価値を迅速に提供するモデルについて、ITとビジネスを調整するのに役立ちます。

パターン：アプリケーション・コンポーネントのモジュール化

ベスト・プラクティスを適用すると、コンポーネントが増加し始めます。そのため、コンポーネントのライブラリとフレームワークを構築することが必要になります。その後、UI機能を再利用可能なコンポーネントに分割することが重要になります。そのため、オラクルでは、TRMで再利用可能なコンポーネントを、Oracle BPM Suite (TRMReuse) で容易に共有できる別のアプリケーションに分離しました。



上記のスクリーンショットでは、TRM でアプリケーションを分離した他の例として、管理者用 (TRMAdminUI)、タスクを実行するユーザー用 (TRMTaskUI) など、ビジネス・ロール・グループでグループ化されたビジネス向け UI も示しています。明確に定義された一貫して使用されるモジュールの機能を分離すると、再利用か再構築かを判断するときに正しい選択を行えます。SOA のサービス分離における他の多くのベスト・プラクティスは、大規模な BPM 開発環境に当てはまります。

「Oracle Process AcceleratorsはBPMの適切な出発点を示し、経験豊富な開発者に対し、プロセス自動化の進化するアーキテクチャをリセットおよび習得するのに最適な方法を示します。本書で説明しているOracle Process Acceleratorsのベスト・プラクティスは、プロセス改善チームが短期間で価値を生み出すのに役立ちます。また、Oracle Process Acceleratorsにより、チームでは、利便性、再利用性、拡張性を高めながらプロセスをより簡単に進化させる方法を習得できます」

IDC、Maureen Fleming

結論

低コストで適切なプロセス・ソリューションを一貫して提供している組織と、BPMで低いROIしか得られない組織の相違点は、IT-ビジネス提供モデルにあります。Oracle Process Acceleratorsは、ITチームとビジネス・チームの両方の習得期間を短縮し、ビジネスと拡張性の両方のベスト・プラクティスを考慮して開発される、重要なビジネス・プロセスを実行するための出発点として機能します。Oracle Process Acceleratorsは、ビジネス部門が価値を実現してIT部門が低コストで再利用可能なコンポーネントを提供するのを支援する、ベスト・プラクティスを組み込みます。その結果、Oracle Process Acceleratorsのベスト・プラクティスとフレームワークを採用した顧客は、プロセス・エクセレンスを短期間で実現できます。



Oracle Process Accelerators
2012年7月

著者：Cesare Rotundo

共著者：Chuck Jones、Vinay Pamadi、
Karthek Lenkala、Ralf Müller、
Bhagat Nainani、Manoj Das

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

海外からのお問い合わせ窓口：
電話：+1.650.506.7000
ファクシミリ：+1.650.506.7200

oracle.com



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2012, Oracle and/or its affiliates. All rights reserved. 本文書は情報提供のみを目的として提供されており、ここに記載される内容は予告なく変更されることがあります。本文書は一切間違いがないことを保証するものではなく、さらに、口述による明示または法律による黙示を問わず、特定の目的に対する商品性もしくは適合性についての黙示的な保証を含み、いかなる他の保証や条件も提供するものではありません。オラクル社は本文書に関するいかなる法的責任も明確に否認し、本文書によって直接的または間接的に確立される契約義務はないものとします。本文書はオラクル社の書面による許可を前もって得ることなく、いかなる目的のためにも、電子または印刷を含むいかなる形式や手段によっても再作成または送信することはできません。

OracleおよびJavaはOracleおよびその子会社、関連会社の登録商標です。その他の名称はそれぞれの会社の商標です。

IntelおよびIntel XeonはIntel Corporationの商標または登録商標です。すべてのSPARC商標はライセンスに基づいて使用されるSPARC International, Inc.の商標または登録商標です。AMD、Opteron、AMDロゴおよびAMD Opteronロゴは、Advanced Micro Devicesの商標または登録商標です。UNIXはX/Open Company, Ltd.によってライセンス提供された登録商標です。0112

Hardware and Software, Engineered to Work Together