

システム構築現場における SOA実践アプローチ

株式会社 日立製作所 ソフトウェア事業部
AP基盤ソフトウェア本部 AP基盤技術支援センタ

桐越 信一

Contents

- 1. SOAの適用形態**
(適用形態の考え方)
- 2. SOAシステムの事例**
(実際の構築事例における基準・設計例)

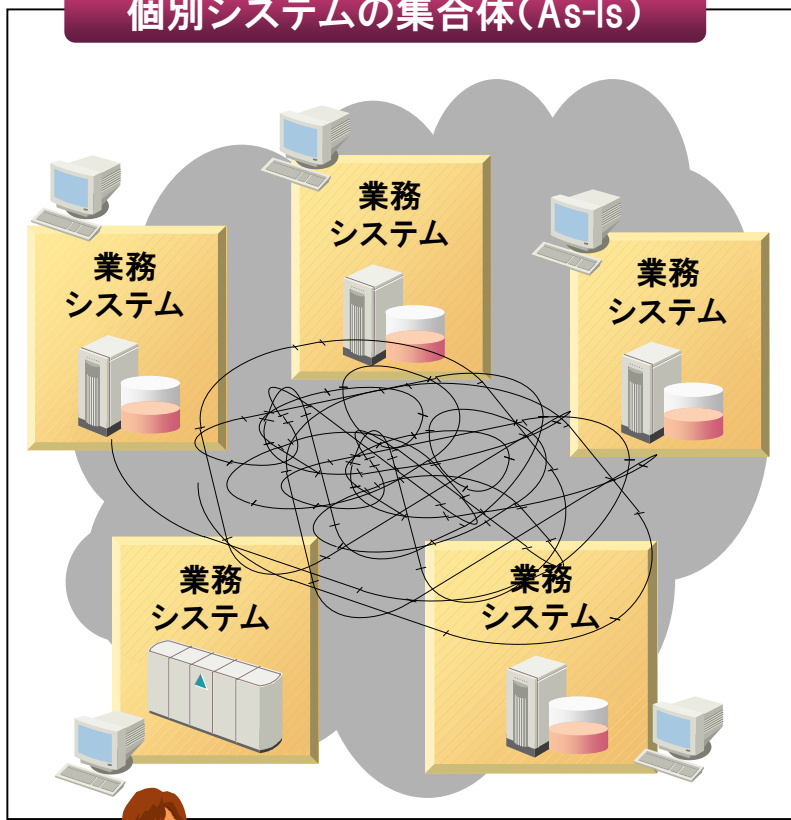
1

SOAの適用形態 (適用形態の考え方)

1-1. SOAに対する適用レイヤとは

SOA化と言うと・・・

個別システムの集合体(As-Is)



As-Is → To-Be



全体最適なシステム形態(To-Be)

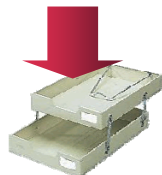


- プロセスと言うのはどこのレイヤのプロセスなのか？
→ ビジネスプロセスを繋いで統合していくと言っても、それはどこの部分なのか？
- そもそもシステムの統合はプロセスだけなのか？
→ 業務の流れを統合していくだけではなく、情報も統合すべき要素なのでは？

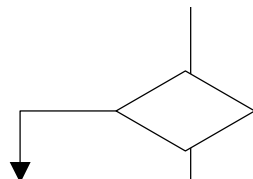
1-2. 業務と作業の構造

業務プロセス

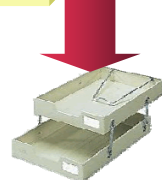
注文伝票



受注受付



受注伝票
与信依頼票



与信受付

輸出
審査担当

業務 (受注)

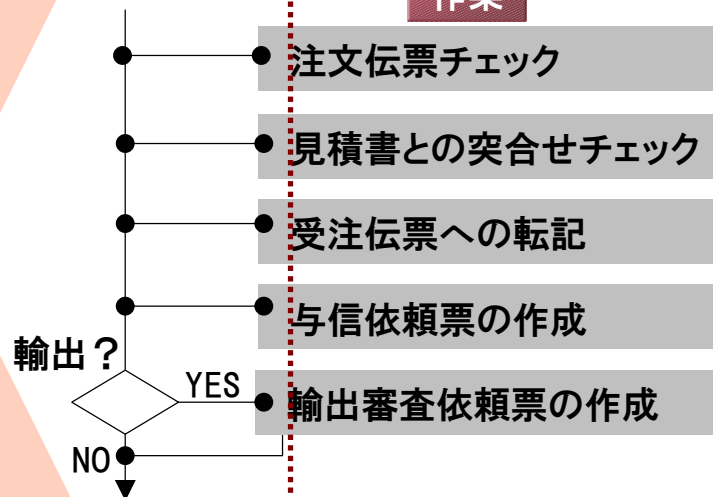
受注担当者



作業プロセス

作業

作業内容(含む規則)



- 記入内容チェックと修正
- 金額突合せと金額の差異判定
- 注文伝票からの転記と商品名→製造品名
- 与信に必要な情報の記入
- 輸出審査に必要な情報の記入

業務 (与信)

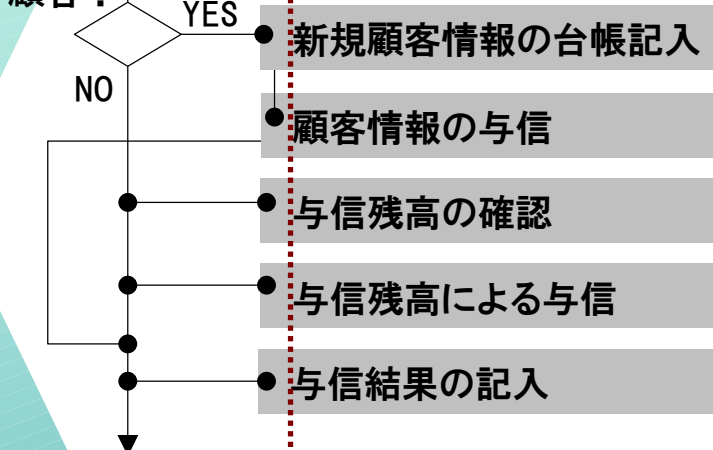
与信担当者



新規顧客?

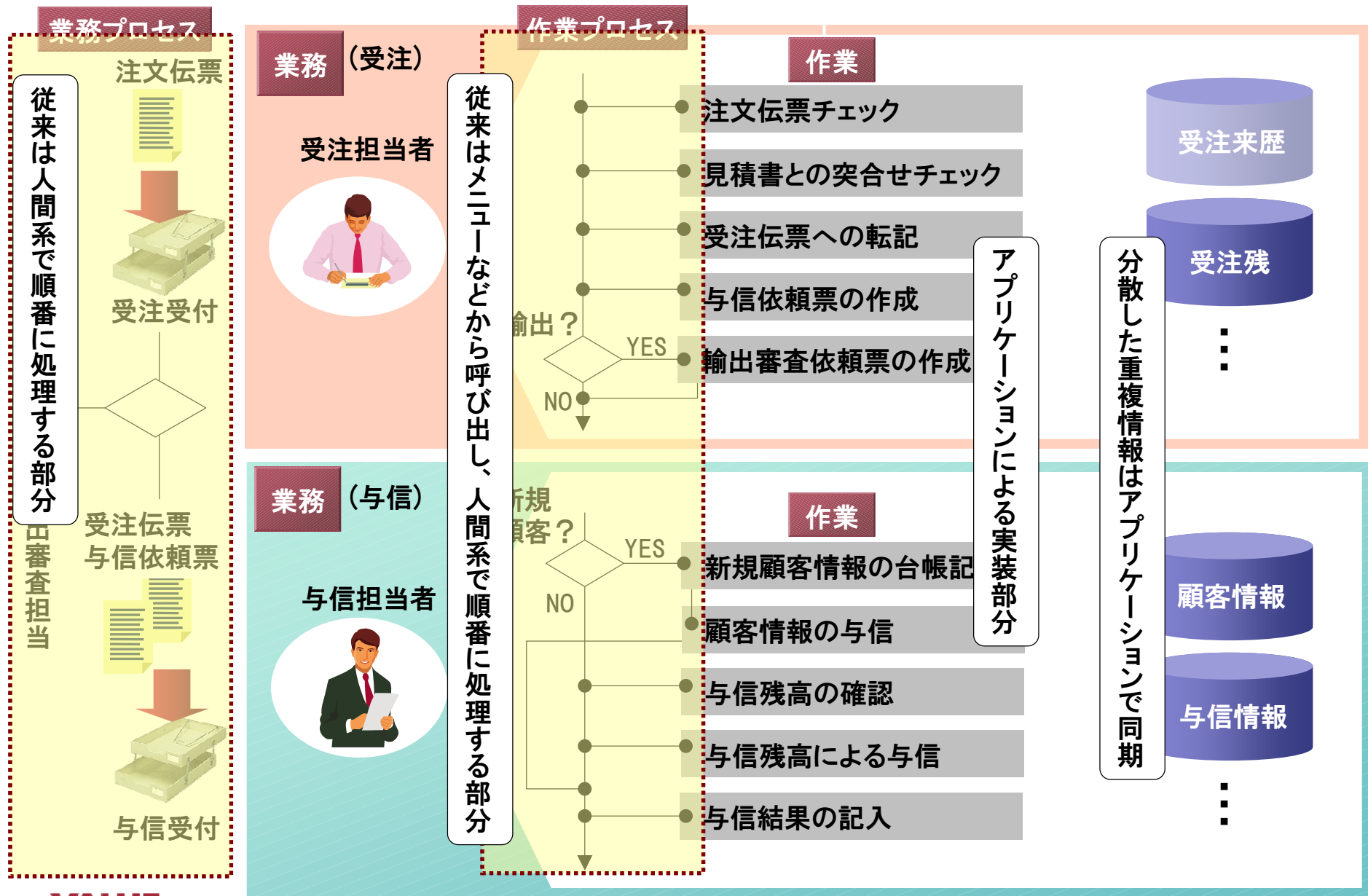
作業

作業内容(含む規則)

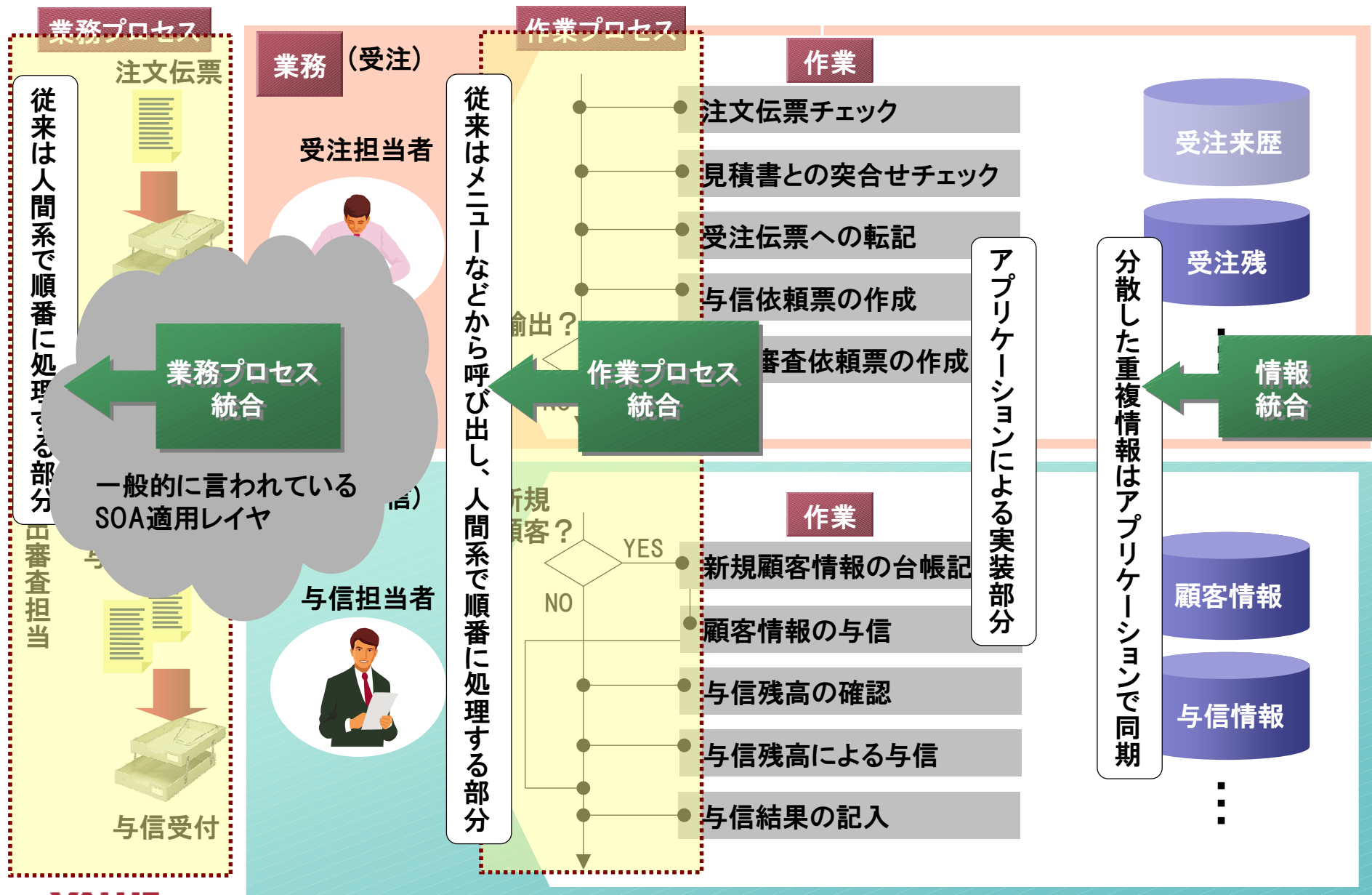


- 顧客情報の記入
- 信用情報を元にした与信
- 当該顧客与信残高確認
- ルールに基づく与信判定
- 判定結果の記入

1-3. 業務と作業の構造



1-4. 業務と作業の構造



1-5. SOAによるシステム統合形態

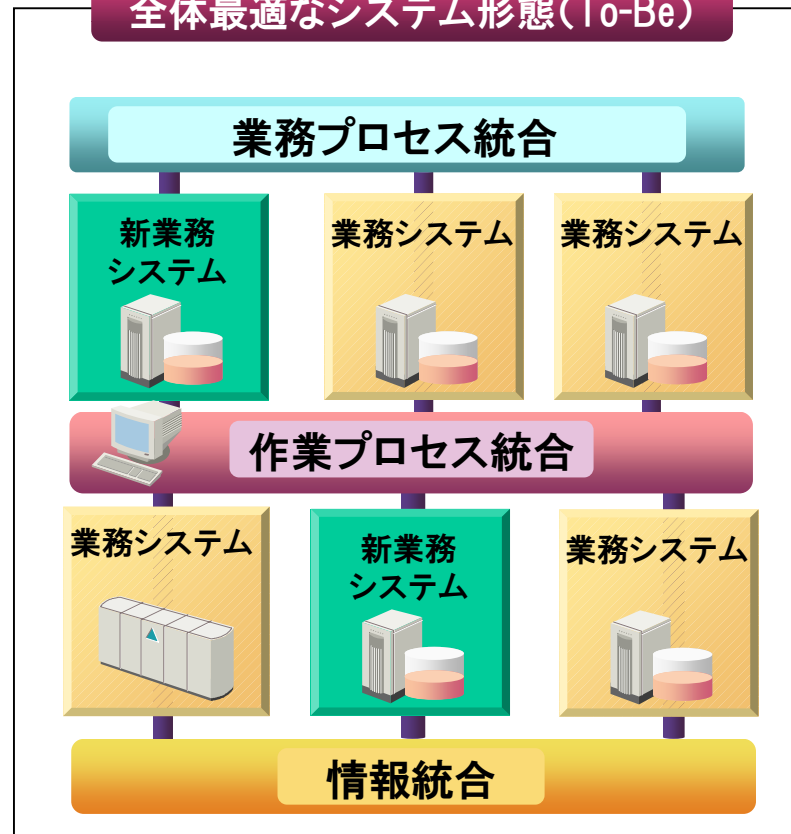
ポイント

- 一般に現状の情報システムの更改形態は、個別のシステム構築が多い
＜課題＞ その結果、情報の重複、情報連携方式の複雑化などの無駄が発生している
- 各種統合レイヤで全体最適なシステムを実現
＜効果①＞ 業務・作業プロセスを再編し局所化・共通化を図ることで、変化に強いシステムを実現可能
＜効果②＞ 組合せ型開発により、迅速に新業務システムの構築が可能
＜効果③＞ 情報統合での重複データの排除と情報の連携方式統一による「見える化」の実現が可能

個別システムの集合体(As-Is)



全体最適なシステム形態(To-Be)



1-6. SOAによるシステム統合形態の詳細

SOAによるシステム統合

業務プロセス統合

個人組織間などの業務連携部分

自動業務フロー

定型的な業務のフローを制御

対話業務フロー

非定型的な業務のフローを制御

作業プロセス統合

個人の作業間の連携部分

作業フロー

作業のフローを制御

作業ポータル

作業の画面を統合

情報統合

情報の連携部分

フェデレーション

仮想表による情報アクセス

レプリケーション

複製物による情報同期

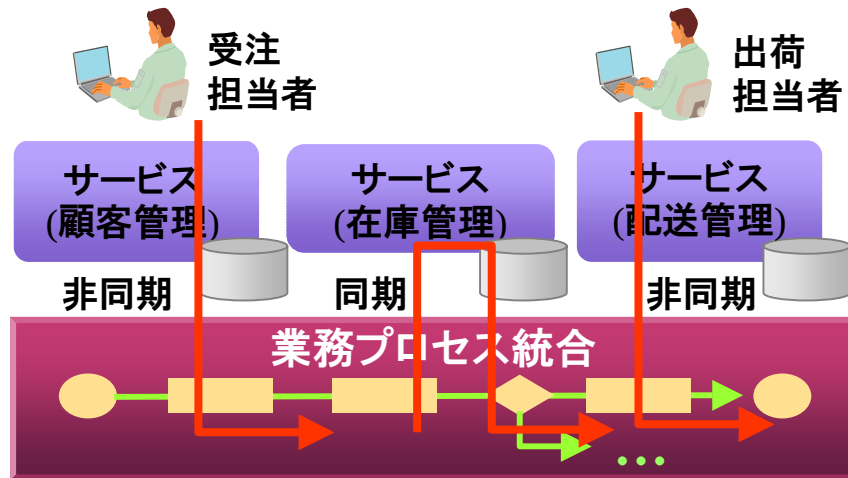
ETL

データを抽出(Extract)、加工(Transform)、書出し(Load)による情報同期
バッチ的なデータウェアハウス機能

1-7. 業務プロセス統合(自動業務フローと対話業務フロー)

自動業務フロー

- サービス間を同期・非同期で連携

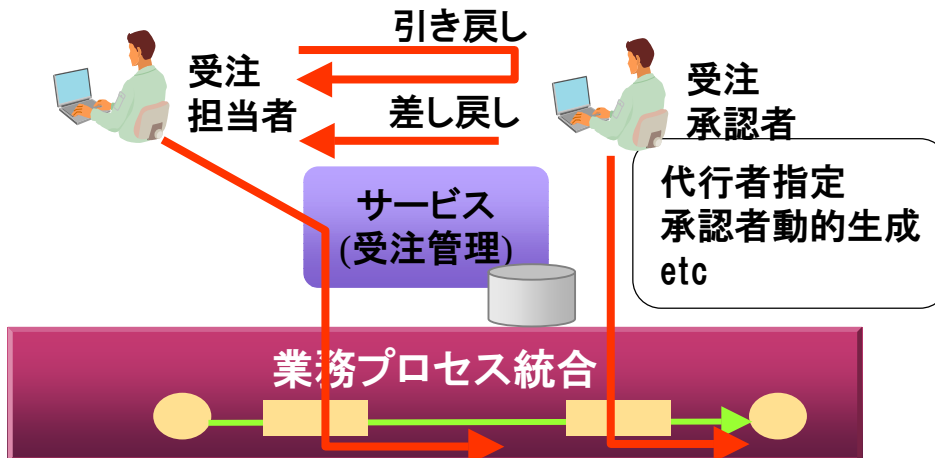


ポイント

- 定型、非定型業務プロセスに沿ったサービスの実行
- 各種サービス接続アダプタによるサービス接続部分の作り込み防止
- データ、フォーマット変換機能内蔵によるデータ整合性の確保

対話業務フロー

- サービスを共有し、非定型プロセスで業務実行



Cosminexus の場合
uCosminexus Service Platform

業務プロセス統合基盤

自動業務フロー
制御基盤

対話業務フロー
制御基盤

サービス
接続基盤

作業プロセス統合基盤

情報統合基盤

1-8. SOAによるシステム統合形態の詳細

SOAによるシステム統合

業務プロセス統合

個人組織間などの業務連携部分

自動業務フロー

定型的な業務のフローを制御

対話業務フロー

非定型的な業務のフローを制御

作業プロセス統合

個人の作業間の連携部分

作業フロー

作業のフローを制御

作業ポータル

作業の画面を統合

情報統合

情報の連携部分

フェデレーション

仮想表による情報アクセス

レプリケーション

複製物による情報同期

ETL

データを抽出(Extract)、加工(Transform)、書出し(Load)による情報同期
バッチ的なデータウェアハウス機能

1-9. 作業プロセス統合(作業フロー)

ポイント

- 定型的な作業フローに沿った作業の実現
 - ・作業順番をシナリオ化し、それに沿って作業を進めることでの作業効率の向上
 - ・蓄積されたノウハウに基づく作業実行により、属人的な作業を標準化
- 作業シナリオに基づく作業フロー定義と業務画面の簡易開発を実現

作業者

The screenshot displays the uCosminexus Navigation Platform interface within a Microsoft Internet Explorer browser window. The interface is divided into two main sections: a workflow diagram on the left and a detailed task screen on the right.

Workflow Diagram (Left): A flowchart titled 'ネットワーク障害対応' (Network Troubleshooting). It starts with '障害番号入力' (Input trouble number), followed by '障害内容確認' (Confirm trouble content). A decision diamond leads to three paths: '原因調査' (Cause investigation), '修理依頼' (Request repair), and '復旧通知' (Notification of restoration). The '原因調査' path leads to another decision diamond, which branches into 'エスカレーション' (Escalation) and '修理依頼' (Request repair).

Task Screen (Right): Titled '原因調査' (Cause investigation). It contains instructions for performing communication tests. Below the instructions, there is a list of '障害状況' (Trouble status) with radio buttons for selection: 'お客様の端末' (Customer's terminal), 'ルータ' (Router), 'プロキシ' (Proxy), 'センタ' (Center), and 'サーバ' (Server). The screen also includes a '前へ' (Previous) button and buttons for 'エスカレーション' (Escalation) and '修理依頼' (Request repair).

業務プロセス統合基盤

Cosminexus の場合
uCosminexus Navigation Platform

作業プロセス統合基盤

作業フロー
制御基盤

ポータル
制御基盤

情報統合基盤

1-10. 作業プロセス統合(作業ポータル)

ポイント

- 複数サービスの画面インタフェースを統合し、作業のポータルを実現
 - ・利用者の担当作業に応じて、複数作業の画面インタフェースを統合
 - ・既存業務システムを活用した統合を実現。段階的な統合も可能
- 画面間での連携・データ引継ぎの簡易開発を実現



業務プロセス統合基盤

Cosminexus の場合
uCosminexus Portal Framework

作業プロセス統合基盤

作業フロー
制御基盤

ポータル
制御基盤

情報統合基盤

1-11. SOAによるシステム統合形態の詳細

SOAによるシステム統合

業務プロセス統合

個人組織間などの業務連携部分

自動業務フロー

定型的な業務のフローを制御

対話業務フロー

非定型的な業務のフローを制御

作業プロセス統合

個人の作業間の連携部分

作業フロー

作業のフローを制御

作業ポータル

作業の画面を統合

情報統合

情報の連携部分

フェデレーション

仮想表による情報アクセス

レプリケーション

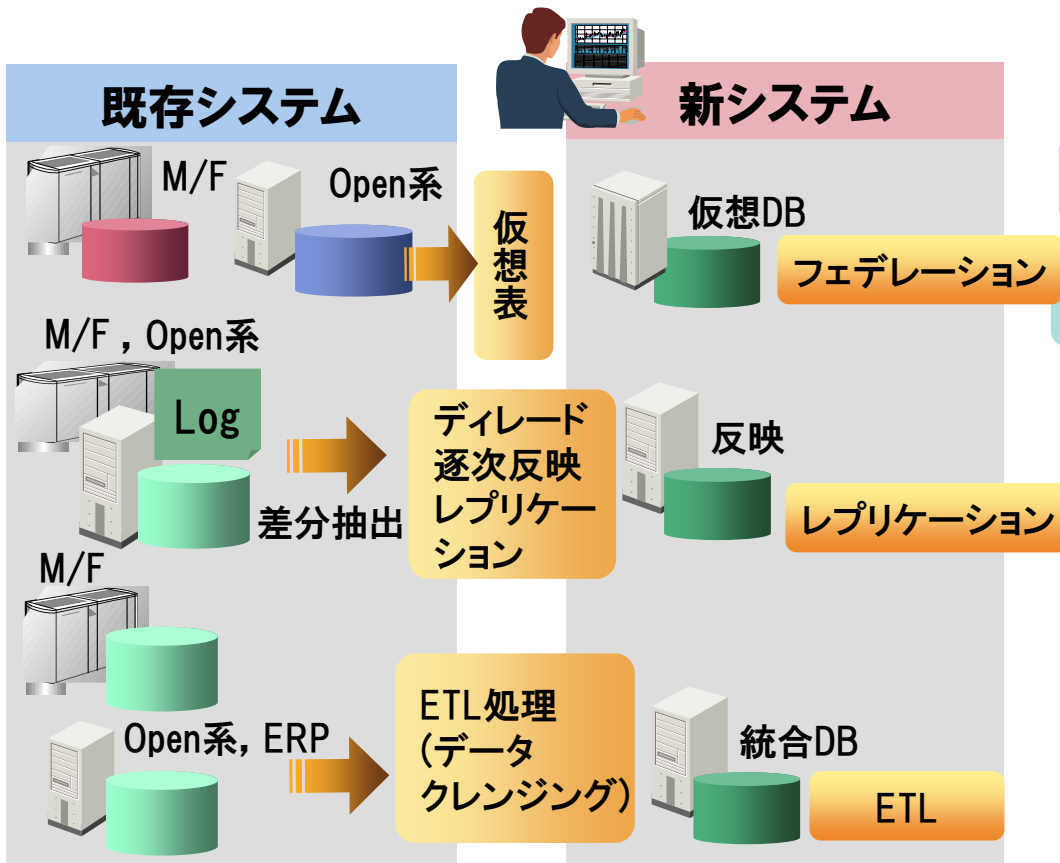
複製物による情報同期

ETL

データを抽出(Extract)、加工(Transform)、書出し(Load)による情報同期
バッチ的なデータウェアハウス機能

ポイント

- 複数システムに散在する情報を仮想表として統合し、その仮想表にアクセス
- 複数システムに散在する情報をある程度リアルタイムに複製
- 企業内に散在する大量データを、目的に合わせて統合



※ただし、DB製品やDBのバージョンによって
実現できるものと出来ないものがある

業務プロセス統合基盤

作業プロセス統合基盤

Cosminexus の場合
uCosminexus Information Federator
uCosminexus Information Replicator
DataStage/QualityStage

情報統合基盤

フェデレーション
制御基盤

レプリケーション
制御基盤

ETL
制御基盤

1-13. ビジネスプロセス可視化でのBAMによる業務改善

BAMとは

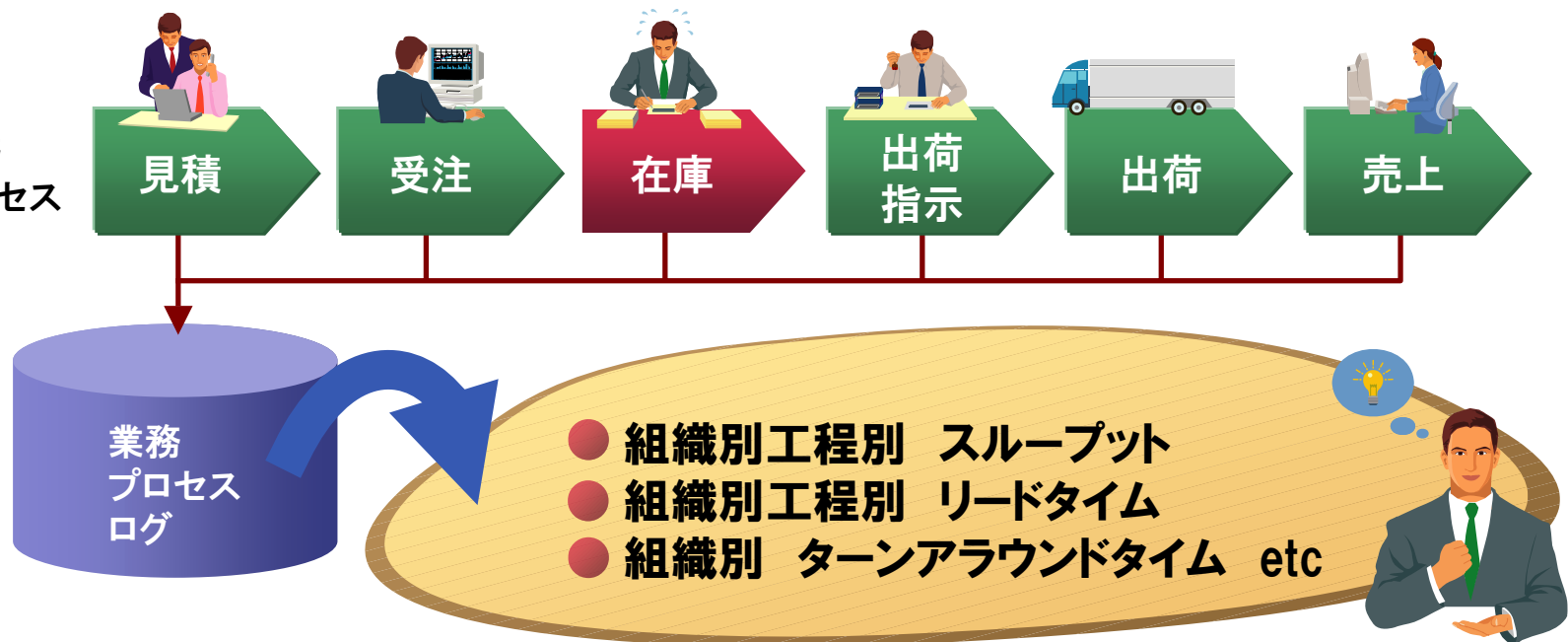
Business Activity Monitoringの略であり、あらかじめ定めた業務の流れ(プロセス)に沿って、それぞれの作業状況をリアルタイムに監視すること
2001年 米ガートナーが提唱

BAMの狙い

- **業務改善**
BAMで顕在化した問題点を業務プロセスに反映
- **意思決定スピードの向上**
業務主管部門あるいは経営層での迅速な問題把握

例

業務
プロセス



作業プロセスからの改善への反映例



Point

オペレーションログ
から改善点発見

ログ
収集



新たな知識・ノウハウの創出



1-15. SOAの利点

SOAの特徴	イベント	利点
ビジネスプロセスとAP(サービス)の分離	ビジネスプロセスの変化	● IT面での改修工数の低減 ビジネスプロセスが外出しのため、AP(サービス)に影響を与えず、変更に対応可能
	AP(サービス)の修正・入れ替え	● IT面での改修工数の低減 ビジネスプロセスとAP(サービス)が分離していることで、プロセスに影響を与えずAP(サービス)のみ変更・入れ替えが可能
AP(サービス)のコンポーネント化	AP(サービス)の作成・修正	● IT面での開発効率向上と改修工数の低減 ・共有できるコンポーネントにより、開発効率の向上やテスト品質の向上が期待できる ・修正時、修正範囲を局所化できる
各種アダプタ、変換基盤によるシステム連携	システム連携時	● IT面での開発効率向上と改修工数の低減 ・アダプタの利用により、連携部分の作りこみが不要になることでの開発効率向上が期待できる。 ・既存との連携部分の改修不要による工数の低減
ビジネスプロセスの可視化	業務監視や業務改善	● 業務・経営面での改善への寄与 ・ビジネスプロセスが可視化できるため、業務監視が行いやすい ・業務の処理量やボトルネックなどが把握できるため業務改善に繋げやすい

2

SOAシステムの事例紹介

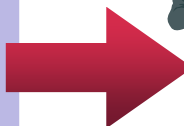
(実際の構築事例における基準・設計例)

2-1. SOA事例(業務プロセス統合) : 製造業



課題

- ・サービス毎に主管の組織／システムが異なり、全体取り纏めが困難
- ・ERP、M/Fなど連携対象はさまざま
- ・処理に時間・コストを要していた

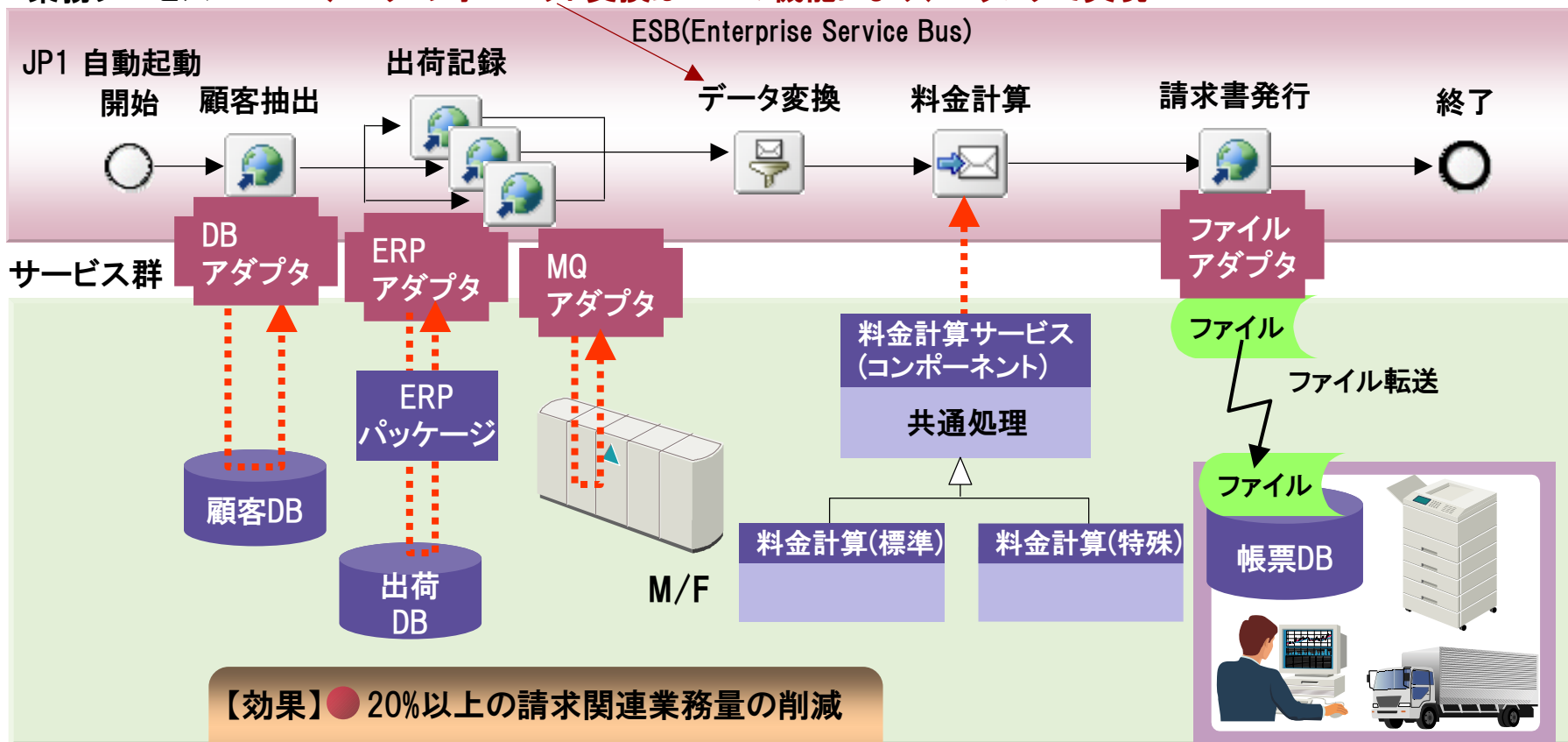


解決策

- 新規開発部分を除き、ESBが持っているアダプタで連携を実現
- 既存I/Fは変更せず
- 請求書発行まで一括処理

業務プロセス

データ・フォーマット変換はESBの機能により、パラメタで実現



【効果】● 20%以上の請求関連業務量の削減

● 処理リードタイムの短縮

2-2. SOA適用の方針と基準

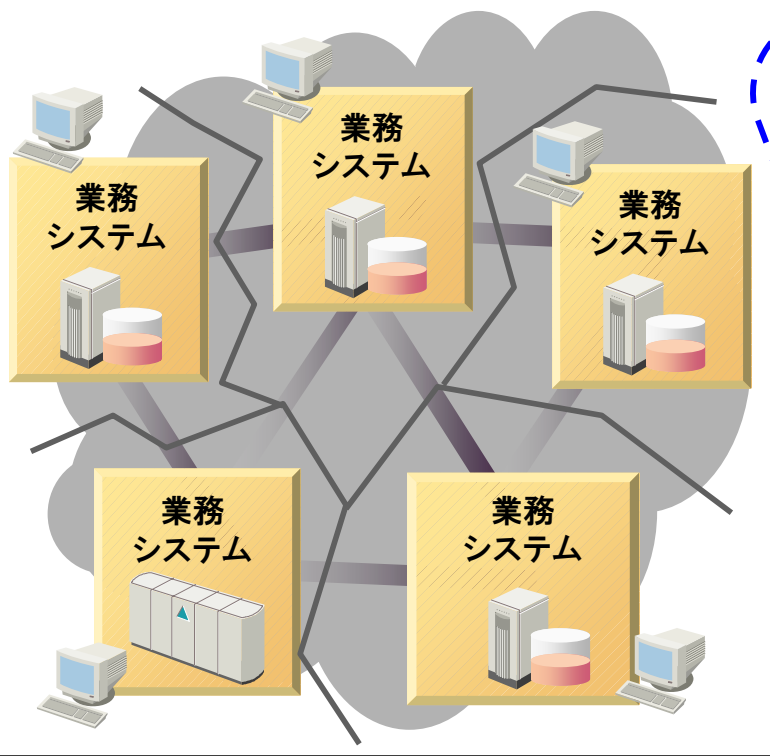
	項目	説明
基本方針	ITガバナンスモデルSOA適用範囲	SOAによる連携とSOA以外の技術による連携の棲み分け方針
	システム統合モデル	システム統合方式の基本方針を標準アーキテクチャとして決定
移行方針	システム統合モデル移行方針	業務プロセス最適化に向けた段階的移行方針の策定
	SOA適用対象システム	移行計画の各フェーズにおけるSOA適用対象システムの明確化
SOA適用基準	SOA連携パターン	SOA連携の各種連携パターンとその適用基準
	マスタ間情報連携	マスタ間の情報統合のために利用する統合基盤の適用方針
	ビジネスプロセス・サービスの粒度	ビジネスプロセス・サービスの粒度を決定するための基準
	サービス連携における情報受渡し	サービス連携における情報受渡しパターンの適用基準
	接続アダプタ	各種アダプタの適用基準
ビジネスプロセス制御基準	分離判断①	案件が途中から動的に1:nの形態となる業務処理の制御方式
	分離判断②	業務プロセスの一部がパッケージ等で隠蔽される場合の制御方式
	取消業務処理方式	「取消」業務に対応するためのビジネスプロセスの処理方式
	ビジネスプロセス起動方式	ビジネスプロセスの起動方式と入力情報の取得方式
	サービス呼出方式	サービス／ビジネスプロセスの呼出方式と呼出プロトコル
運用基準	ログ出力方式	システム連携におけるログ採取ポイントとログ出力方式
	障害検知・通知	システム連携における障害の検知方法と障害検知時の動作基準

2-3. 全体最適なシステムへの移行アプローチ

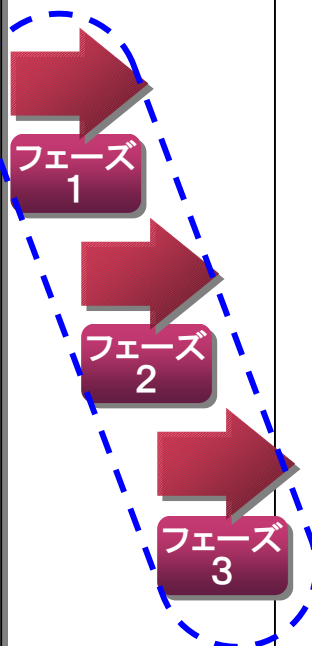
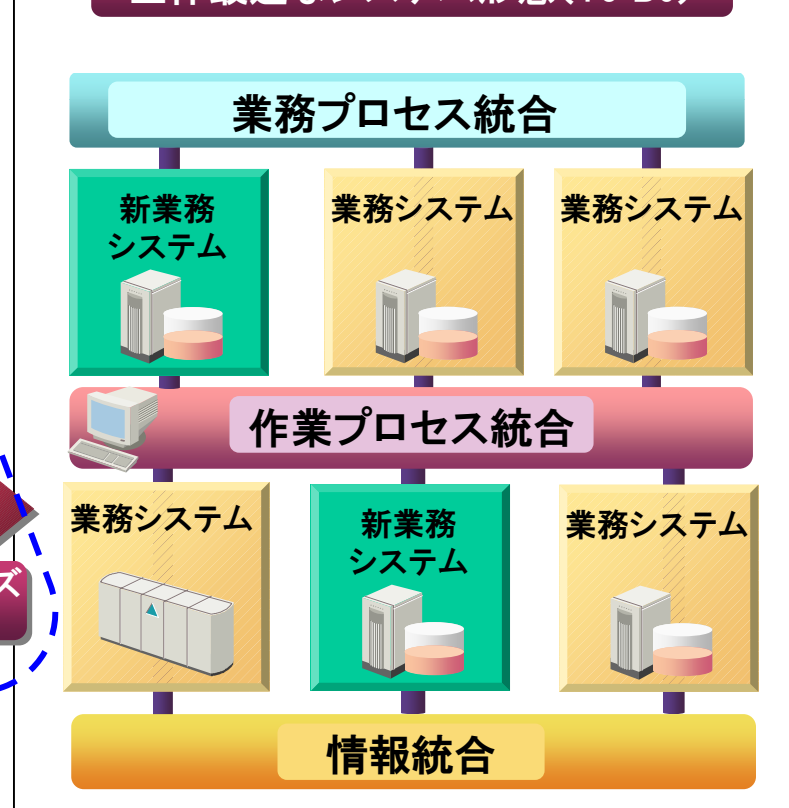
ポイント

- 全体最適なシステムへの全面更改は
 - ＜課題①＞ 情報連携などの部分での設計難易度も高く、全般にリスク(費用、業務移行)も大きい
 - ＜課題②＞ 開発期間が長期化し、効果も最後にならないと出てこない
- 段階的なシステム最適化により、効果早期創出とリスクを回避する
 - ＜効果＞ 現行システムを活用しながら、業務・作業プロセスのシステム化、新規システムの追加、現行システムの変更・刷新などを段階的に実施(効果の早期創出、コスト・時間を最適化)

個別システムの集合体(As-Is)



全体最適なシステム形態(To-Be)



2-4. 段階的移行に対する方針

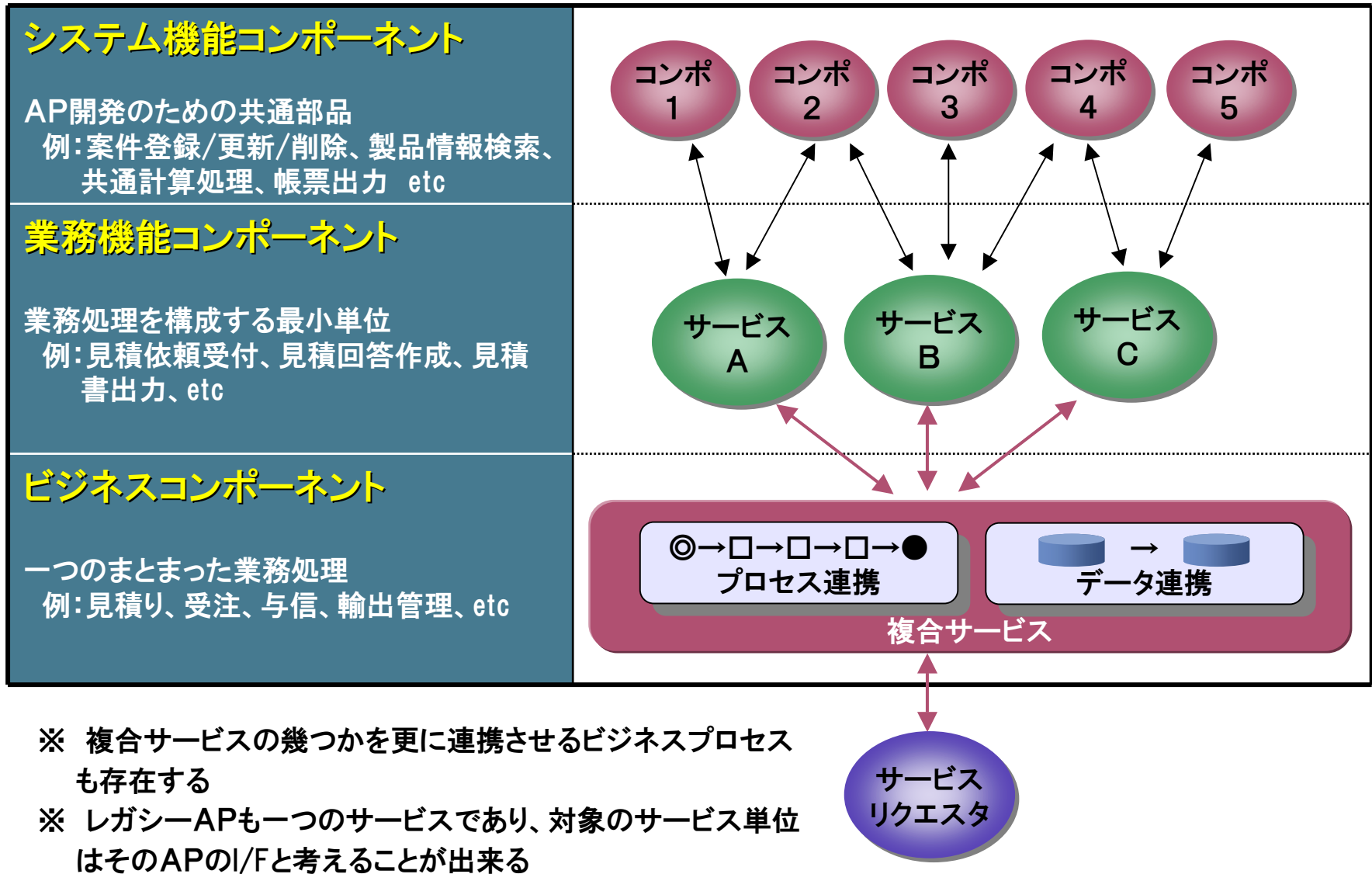
統合基盤	メリット	デメリット
業務プロセス統合 (プロセス統合)	比較的短期間で構築可能	バックグラウンドでの統合基盤のため効果が見えにくい
作業プロセス統合 (フロント統合)	直接利用者に関わる部分であり、見た目の効果がすぐに把握できる	新規開発部分であり、開発工数が掛かってくる
情報統合	AP開発者や運用者にとっての工数低減が可能	・新規のマスタについては、再設計が必要となる ・統合基盤は利用DBIによって機能制限がある

- 見た目の効果は フロント統合＞情報統合＞プロセス統合 であるが、構築工数もそれに準拠する。
- よって、比較的統合しやすい部分からの導入とし、基本的な導入順序は
①プロセス統合 ②情報統合 ③フロント統合 の順とするのが実施しやすい。
- ただし、新規システムにおいては、同時構築の形態を採用してもよい。
- また、フロント統合において、現状操作性が問題になっていて、統合効果をすぐに望めるような作業の画面は優先的に構築する。

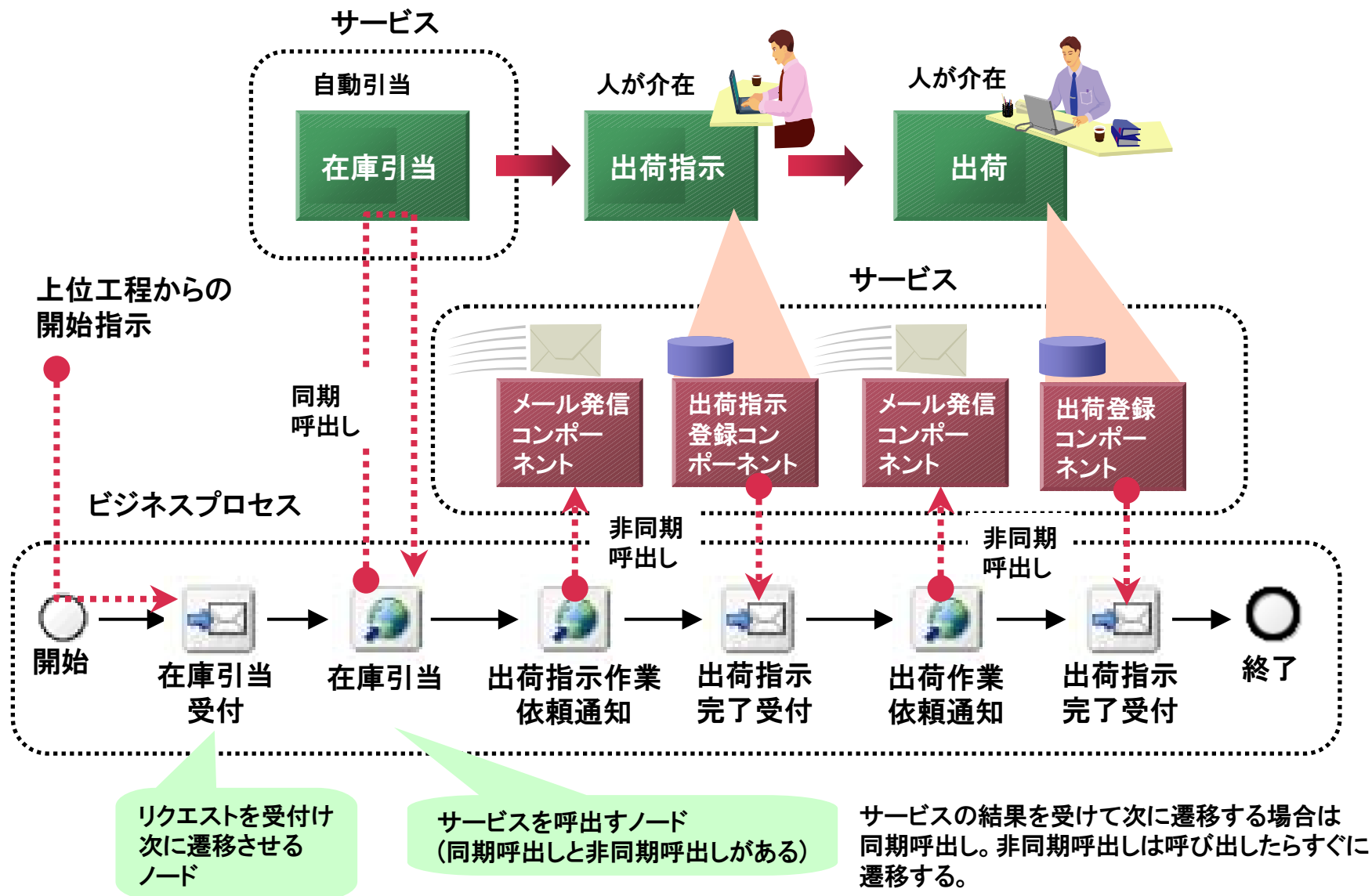
2-5. SOA適用の方針と基準

	項目	説明
基本方針	ITガバナンスモデルSOA適用範囲	SOAによる連携とSOA以外の技術による連携の棲み分け方針
	システム統合モデル	システム統合方式の基本方針を標準アーキテクチャとして決定
移行方針	システム統合モデル移行方針	業務プロセス最適化に向けた段階的移行方針の策定
	SOA適用対象システム	移行計画の各フェーズにおけるSOA適用対象システムの明確化
SOA適用基準	SOA連携パターン	SOA連携の各種連携パターンとその適用基準
	マスタ間情報連携	マスタ間の情報統合のために利用する統合基盤の適用方針
	ビジネスプロセス・サービスの粒度	ビジネスプロセス・サービスの粒度を決定するための基準
	サービス連携における情報受渡し	サービス連携における情報受渡しパターンの適用基準
	接続アダプタ	各種アダプタの適用基準
ビジネスプロセス制御基準	分離判断①	案件が途中から動的に1:nの形態となる業務処理の制御方式
	分離判断②	業務プロセスの一部がパッケージ等で隠蔽される場合の制御方式
	取消業務処理方式	「取消」業務に対応するためのビジネスプロセスの処理方式
	ビジネスプロセス起動方式	ビジネスプロセスの起動方式と入力情報の取得方式
	サービス呼出方式	サービス／ビジネスプロセスの呼出方式と呼出プロトコル
運用基準	ログ出力方式	システム連携におけるログ採取ポイントとログ出力方式
	障害検知・通知	システム連携における障害の検知方法と障害検知時の動作基準

2-6. SOAにおけるサービスとコンポーネントの粒度



2-7. ビジネスプロセスとサービスの関係

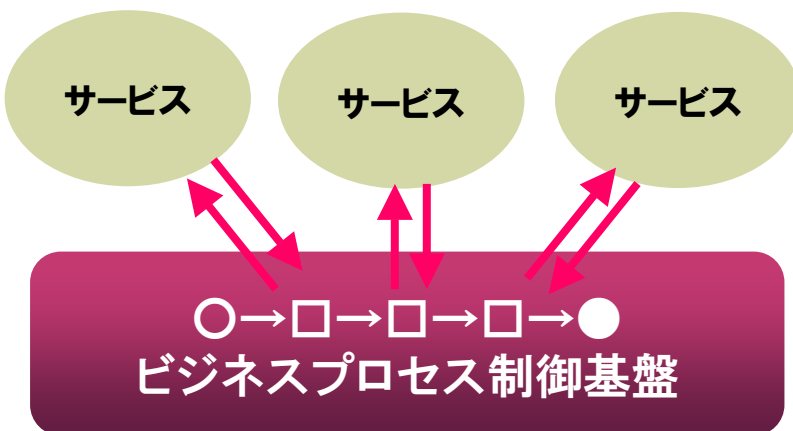


2-8. SOA適用の方針と基準

	項目	説明
基本方針	ITガバナンスモデルSOA適用範囲	SOAによる連携とSOA以外の技術による連携の棲み分け方針
	システム統合モデル	システム統合方式の基本方針を標準アーキテクチャとして決定
移行方針	システム統合モデル移行方針	業務プロセス最適化に向けた段階的移行方針の策定
	SOA適用対象システム	移行計画の各フェーズにおけるSOA適用対象システムの明確化
SOA適用基準	SOA連携パターン	SOA連携の各種連携パターンとその適用基準
	マスタ間情報連携	マスタ間の情報統合のために利用する統合基盤の適用方針
	ビジネスプロセス・サービスの粒度	ビジネスプロセス・サービスの粒度を決定するための基準
	サービス連携における情報受渡し	サービス連携における情報受渡しパターンの適用基準
	接続アダプタ	各種アダプタの適用基準
ビジネスプロセス制御基準	分離判断①	案件が途中から動的に1:nの形態となる業務処理の制御方式
	分離判断②	業務プロセスの一部がパッケージ等で隠蔽される場合の制御方式
	取消業務処理方式	「取消」業務に対応するためのビジネスプロセスの処理方式
	ビジネスプロセス起動方式	ビジネスプロセスの起動方式と入力情報の取得方式
	サービス呼出方式	サービス／ビジネスプロセスの呼出方式と呼出プロトコル
運用基準	ログ出力方式	システム連携におけるログ採取ポイントとログ出力方式
	障害検知・通知	システム連携における障害の検知方法と障害検知時の動作基準

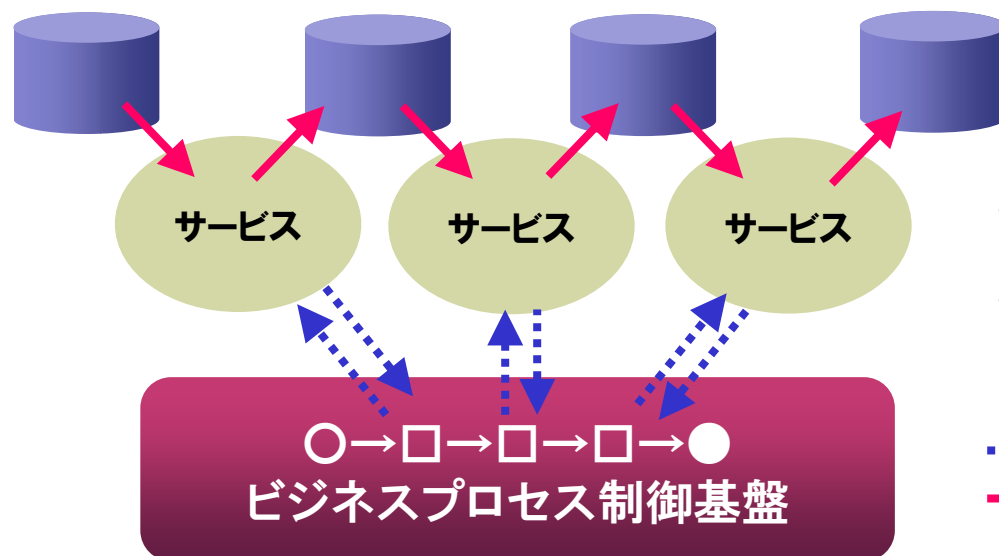
2-9. SOAにおける情報連携パターン

① 業務データ受渡しパターン



- ・ 制御情報と業務データを共に連携基盤が処理
- ・ 業務データが大量になるとリソース／オーバーヘッド共に増大

② 制御情報受渡しパターン



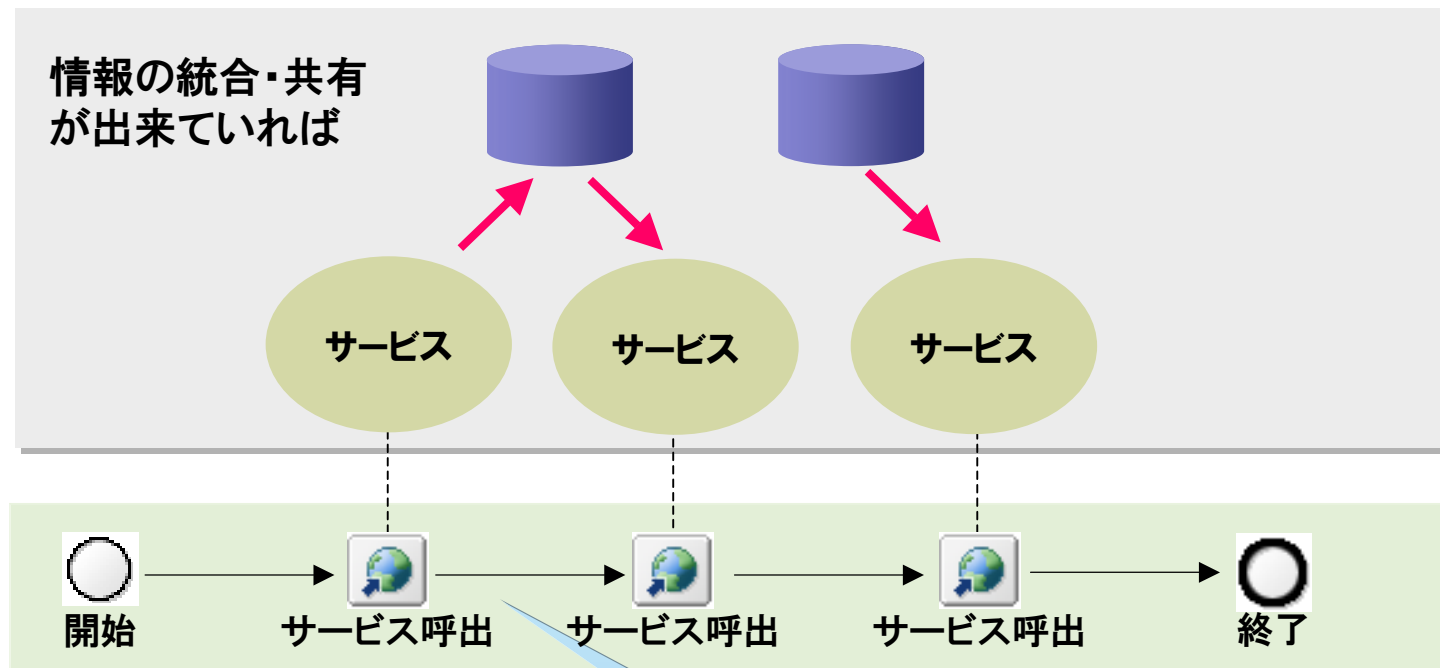
- ・ 制御情報だけを連携基盤で処理し、業務データは別ルートで受け渡し
- ・ 連携基盤は業務データを取り込まないので、リソース／オーバーヘッドは少ない



制御情報

業務データ

2-10. 理想的な情報連携パターン



関係情報は？

- <パターン①> 業務処理が終わったことを通知するための指示情報
- <パターン②> 次の業務が開始できるための最低限の情報(例:キー情報)
- <パターン③> ビジネスプロセスの中で分岐などの判断となる情報

業務データを
そのまま渡す
設計になりがち

2-11. ビジネスプロセスにおける情報受渡し基準

N o	システム間で渡される情報のパターン	システム形態	システム形態の理由・背景
1	業務情報を全て渡す必要がある場合	同一情報を持つDBが分散されている場合 既存システムの連携I/Fが変更不可の場合 (データ変換・フォーマット変換を含む)	- MF・オープン系等でアーキテクチャが異なるため - N/W負荷等で性能を維持するために分散 - 会社間取引のデータ - 上記以外に何らかの理由によりDBが重複している場合(例外)
2	業務の指示情報(終了情報、キー情報 等)、およびビジネスプロセス内で条件判断となる情報のみを渡す場合	受け取り側の業務システムがその情報をもとに業務遂行可能なケース全てが対象	情報システムのあるべき姿 (重複情報の分散DBは存在せず 業務DBが共有可能なことが前提)

- 今後の情報システムは上記#2のパターンで構成する
(重複データがシステム間を渡るのは基本的に会社間取引の形態のみとする)

2-12. SOA適用の方針と基準

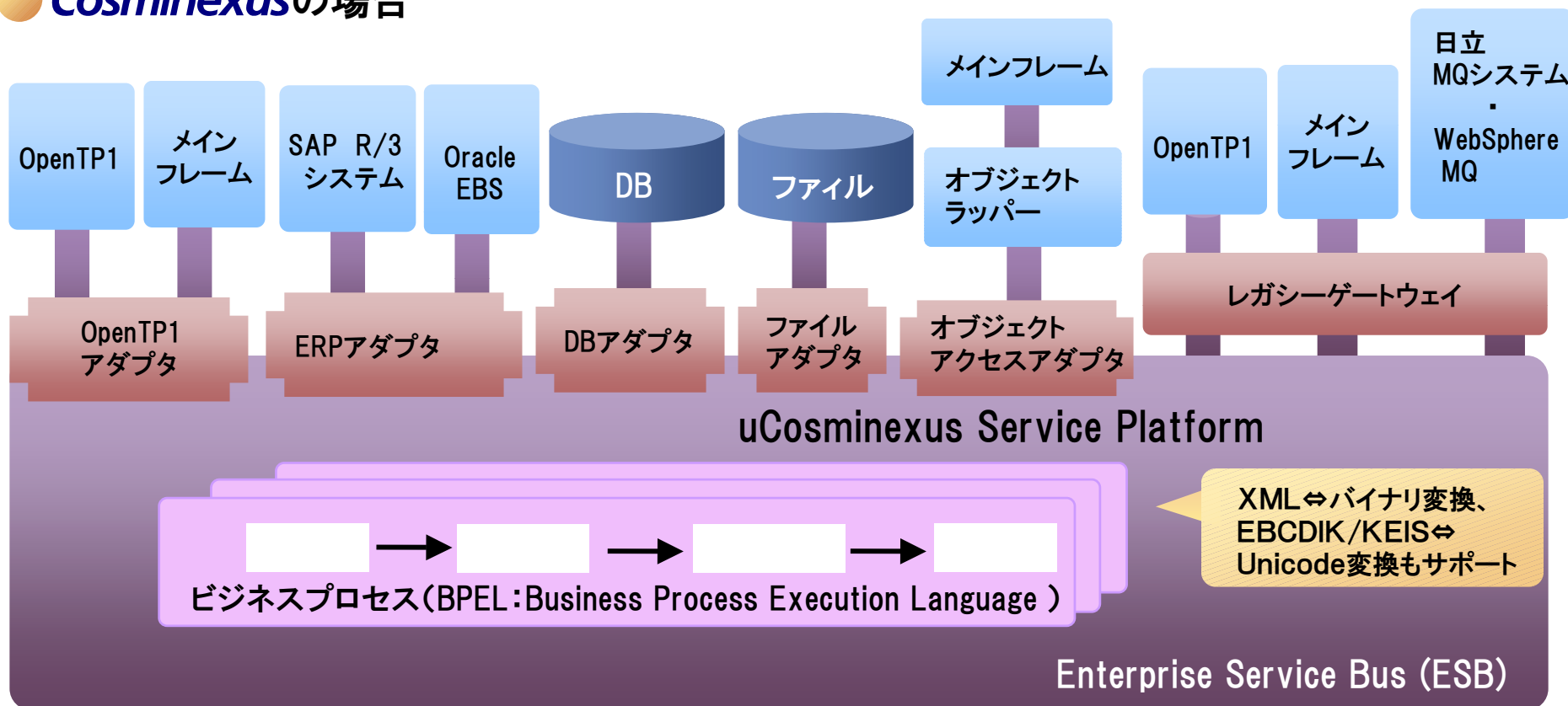
	項目	説明
基本方針	ITガバナンスモデルSOA適用範囲	SOAによる連携とSOA以外の技術による連携の棲み分け方針
	システム統合モデル	システム統合方式の基本方針を標準アーキテクチャとして決定
移行方針	システム統合モデル移行方針	業務プロセス最適化に向けた段階的移行方針の策定
	SOA適用対象システム	移行計画の各フェーズにおけるSOA適用対象システムの明確化
SOA適用基準	SOA連携パターン	SOA連携の各種連携パターンとその適用基準
	マスタ間情報連携	マスタ間の情報統合のために利用する統合基盤の適用方針
	ビジネスプロセス・サービスの粒度	ビジネスプロセス・サービスの粒度を決定するための基準
	サービス連携における情報受渡し	サービス連携における情報受渡しパターンの適用基準
	接続アダプタ	各種アダプタの適用基準
ビジネスプロセス制御基準	分離判断①	案件が途中から動的に1:nの形態となる業務処理の制御方式
	分離判断②	業務プロセスの一部がパッケージ等で隠蔽される場合の制御方式
	取消業務処理方式	「取消」業務に対応するためのビジネスプロセスの処理方式
	ビジネスプロセス起動方式	ビジネスプロセスの起動方式と入力情報の取得方式
	サービス呼出方式	サービス／ビジネスプロセスの呼出方式と呼出プロトコル
運用基準	ログ出力方式	システム連携におけるログ採取ポイントとログ出力方式
	障害検知・通知	システム連携における障害の検知方法と障害検知時の動作基準

2-13. SOAにおける連携基盤(ESB)でのアダプタ

ポイント

- ビジネスプロセス(BPEL)を制御する基盤を持つ
- メインフレームシステム、オンラインシステム、ERPなどの既存資産をシームレスに連携できるアダプタが用意されている。(WSは標準実装)
- データフォーマットの変換機能や文字コード変換機能が提供されている

Cosminexusの場合



2-14. 接続アダプタ適用基準

方針

- 既存システムとの接続においては、既存のI/Fを変更しないことを最優先とする
- 新規開発のSOAサービスは、すべてWSとする

接続アダプタ	適用
DBアダプタ	RDBからの参照、あるいは直接、RDBに更新可能な処理(排他制御などを運用でカバーできるような更新時)において適用
MQアダプタ	既存システムとの連携がMQの場合に適用
ファイルアダプタ	他のシステムとのI/Fがファイル渡しの場合に適用。または、後続する受渡しがftpの場合に適用する
ERPアダプタ	OracleEBSの外部接続用に適用。なお、ERPの外部接続方針については「ERPアドオン設計方針」参照のこと

2-15. SOA適用の方針と基準

	項目	説明
基本方針	ITガバナンスモデルSOA適用範囲	SOAによる連携とSOA以外の技術による連携の棲み分け方針
	システム統合モデル	システム統合方式の基本方針を標準アーキテクチャとして決定
移行方針	システム統合モデル移行方針	業務プロセス最適化に向けた段階的移行方針の策定
	SOA適用対象システム	移行計画の各フェーズにおけるSOA適用対象システムの明確化
SOA適用基準	SOA連携パターン	SOA連携の各種連携パターンとその適用基準
	マスタ間情報連携	マスタ間の情報統合のために利用する統合基盤の適用方針
	ビジネスプロセス・サービスの粒度	ビジネスプロセス・サービスの粒度を決定するための基準
	サービス連携における情報受渡し	サービス連携における情報受渡しパターンの適用基準
	接続アダプタ	各種アダプタの適用基準
ビジネスプロセス制御基準	分離判断①	案件が途中から動的に1:nの形態となる業務処理の制御方式
	分離判断②	業務プロセスの一部がパッケージ等で隠蔽される場合の制御方式
	取消業務処理方式	「取消」業務に対応するためのビジネスプロセスの処理方式
	ビジネスプロセス起動方式	ビジネスプロセスの起動方式と入力情報の取得方式
	サービス呼出方式	サービス／ビジネスプロセスの呼出方式と呼出プロトコル
運用基準	ログ出力方式	システム連携におけるログ採取ポイントとログ出力方式
	障害検知・通知	システム連携における障害の検知方法と障害検知時の動作基準

2-16. ビジネスプロセス設計方針(通常処理)

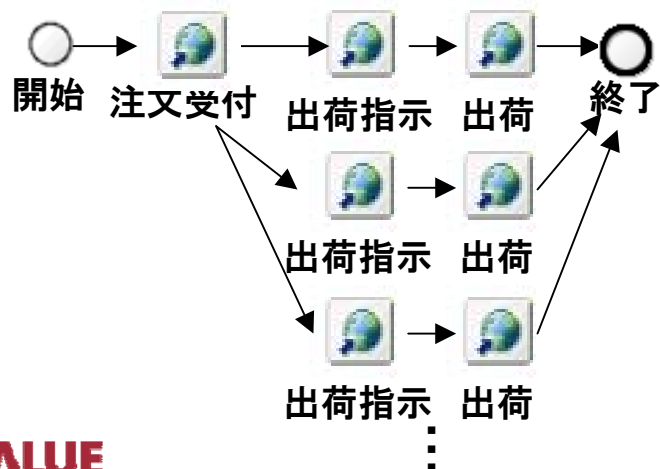
課題

- パッケージなどの利用によりビジネスプロセスが隠れる部分がある
 - ＜課題①＞ パッケージ内のDB連携や人手による業務処理でビジネスプロセスが遂行されることで、全体のビジネスプロセスの一部がパッケージ内に隠れてしまう。
 - ＜課題②＞ これらは外側のSOA基盤では連携できず、SOAとしてのビジネスプロセスは部分的に途切れた形態となってしまふ。

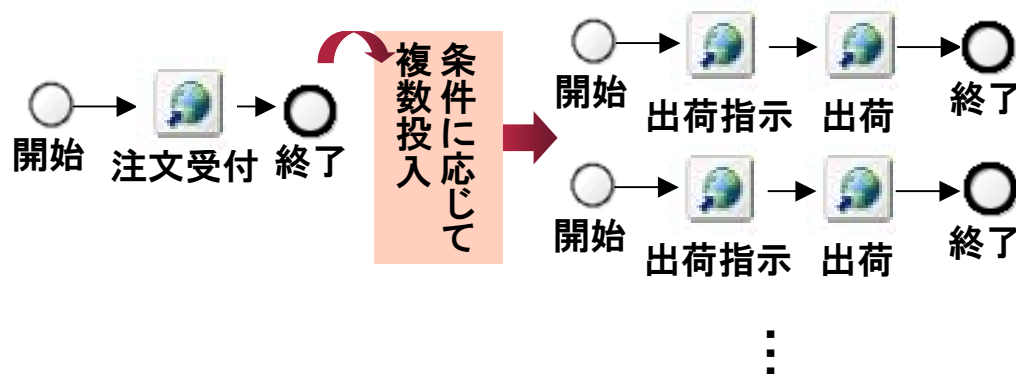
方針

- 業務内容によっては、部分的に分離するビジネスプロセスで構成するのが現実的
 - ＜ポイント①＞ 作業案件はビジネスプロセス上、必ずしも1案件が1ビジネスプロセスとは限らない。(例:1受注に対して、分納によりn出荷。1受注に対して、n製造指示など)
 - ＜ポイント②＞ 上記のような業務処理を想定したビジネスプロセスを、事前に定義することは困難であり、動的処理となってしまふため、部分的に分離したビジネスプロセスを定義する方が自由度が高く現実的である。

- 受注案件に対する分納の場合
(同一ビジネスプロセスでは対処できない)

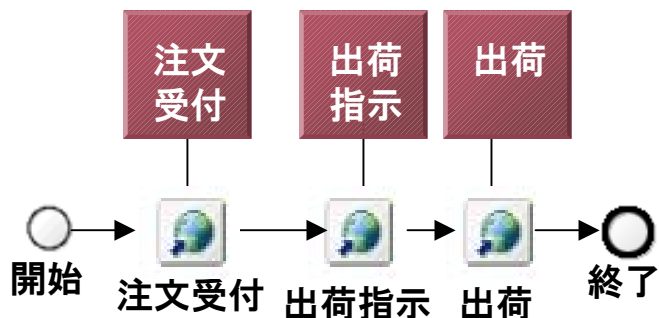


- 部分的に分離したビジネスプロセスで構成する



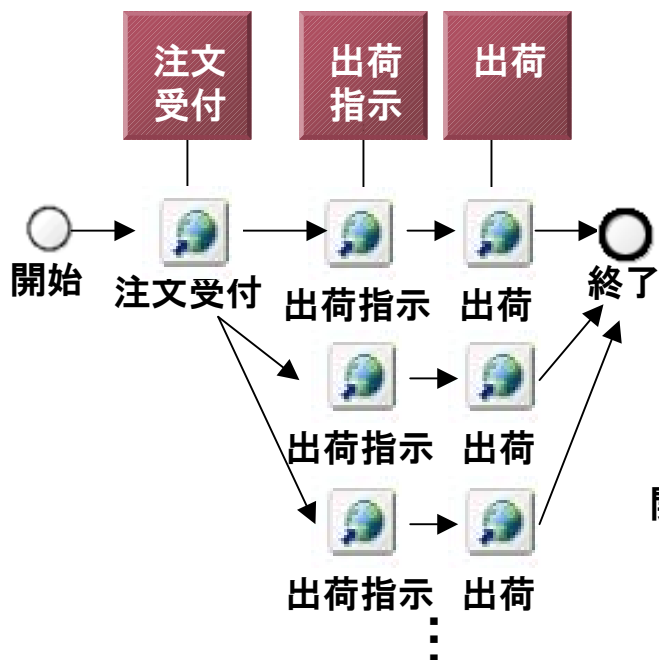
2-17. ビジネスプロセス設計方針(分離判断基準①)

● 一つのビジネスプロセスで定義

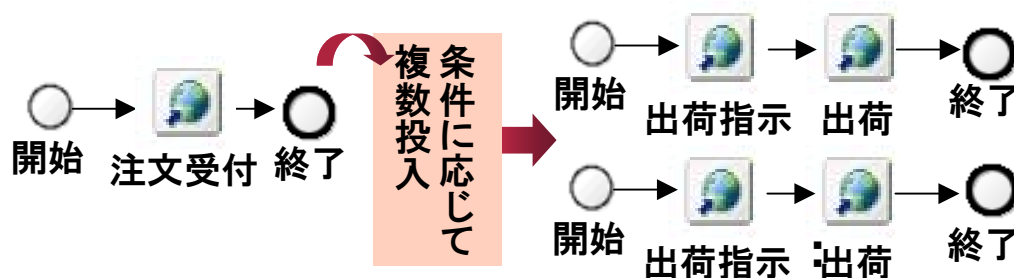


案件が最後まで1:1で推移し、かつ、サービスコンポーネントの処理がビジネスプロセスに沿って自動的に流れていくような業務処理形態

● 分離したビジネスプロセスとして定義

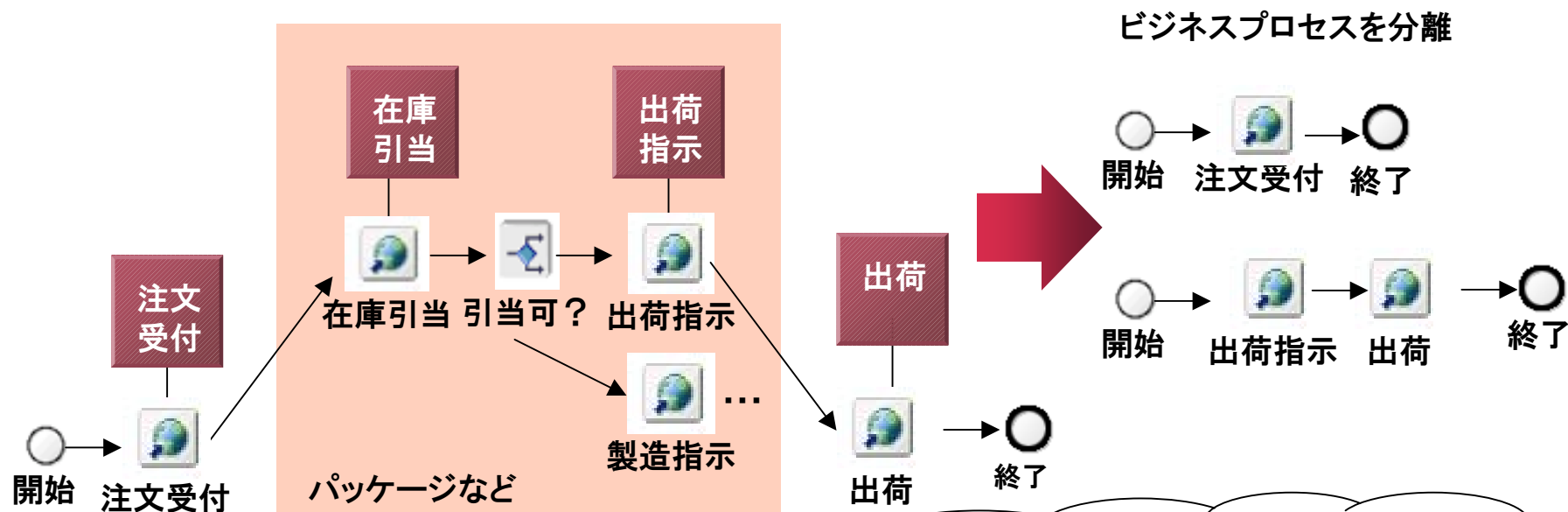


案件が途中から動的に1:nの形になってしまう業務処理形態



2-18. ビジネスプロセス設計方針(分離判断基準②)

● 分離したビジネスプロセスとして定義



なお、分離したビジネスプロセスを統括するビジネスプロセスを用意しておくか否かは検討しておく必要がある

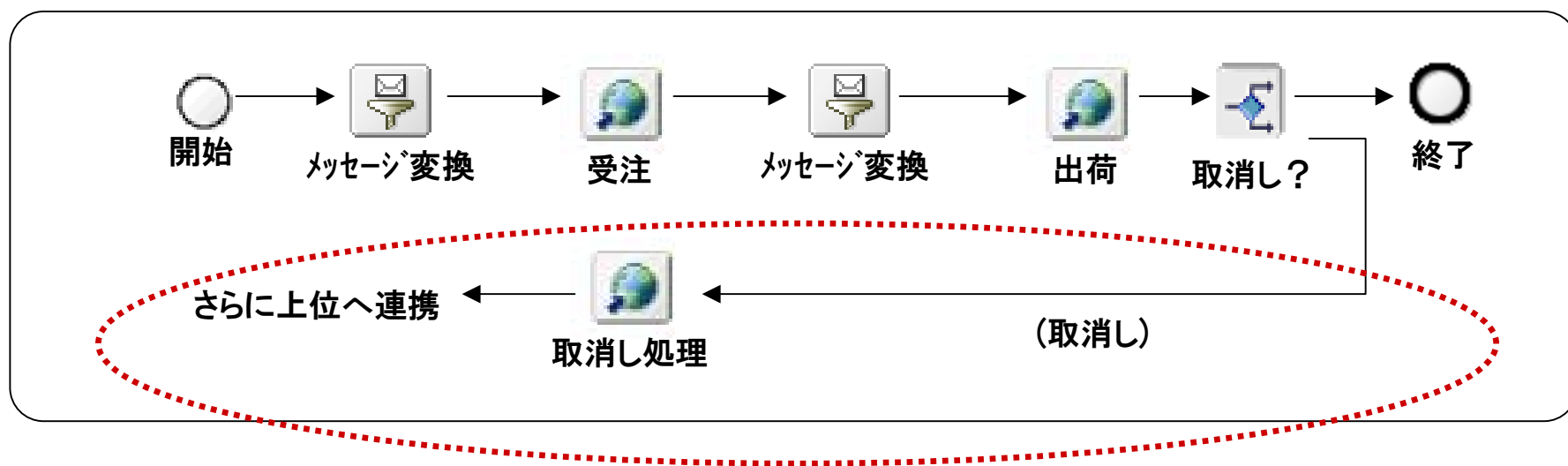
- ・ビジネスプロセスの一部がパッケージなどの中に盛り込まれ、外側からそのプロセスの制御が出来ないようなシステム形態。
- ・業務処理に必要な判断条件部分がパッケージ内に組み込まれてしまい、外側からその判断結果を得られないシステム形態。

2-19. ビジネスプロセス設計方針(取消し基準)

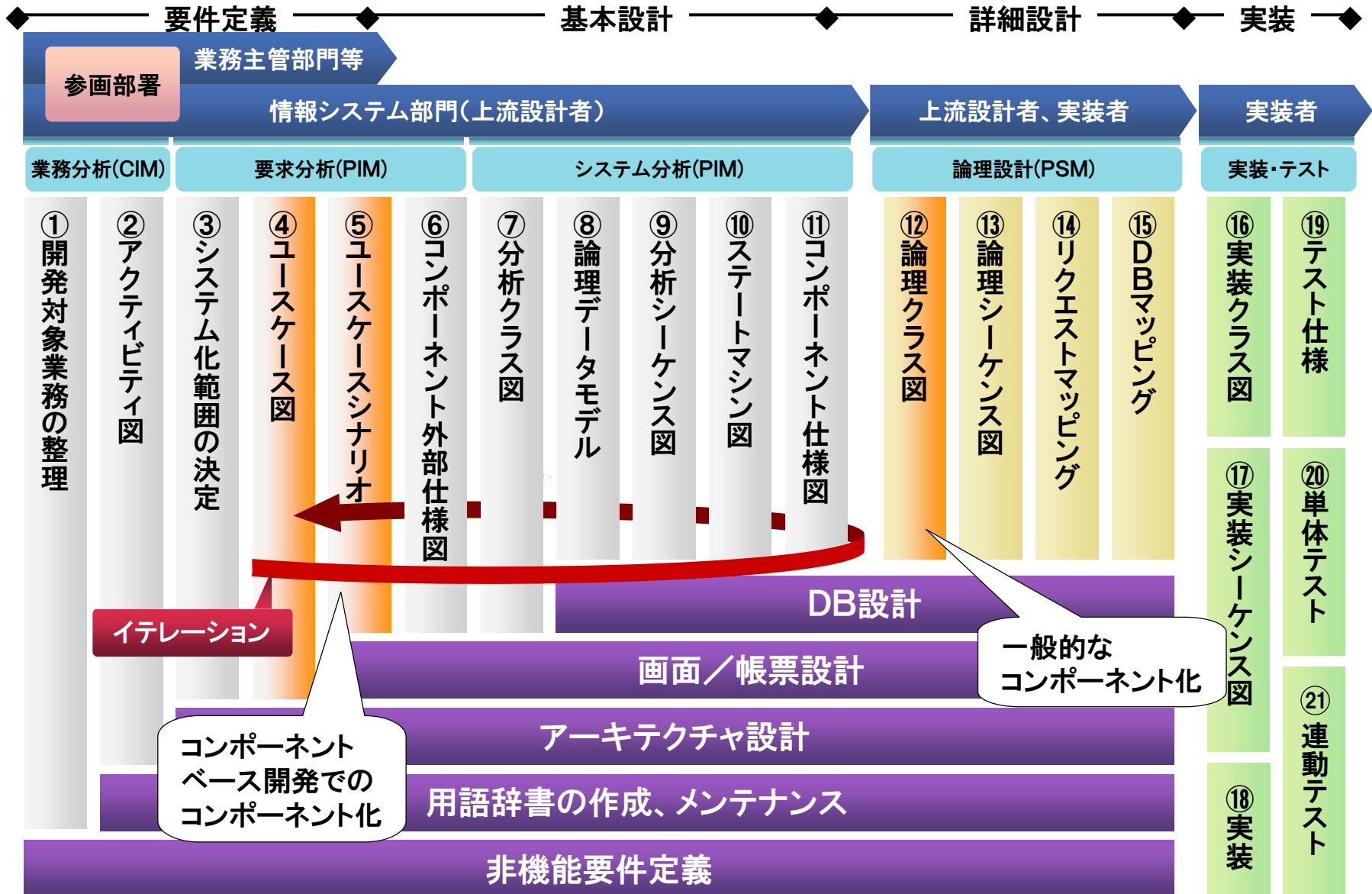
方針

- 受注などの通常の業務処理において、取り消しは「赤黒処理」で行われる。したがって、単純な「赤黒処理」ならば、ビジネスプロセスに沿って赤黒情報を流す処理方式になるはず。
- 逆進での取消しパターンがある場合には、ビジネスプロセスそのものに取り消しのプロセスを作っておき、その経路で処理させる。

下記のような手配系業務では、こういった取り消し処理は行わず、上流から「赤黒伝票」を流すことで対処するのが一般的。よって、下記のような逆進の取り消しニーズが発生した場合、ビジネスプロセスにそれらを作ることによって対処する。

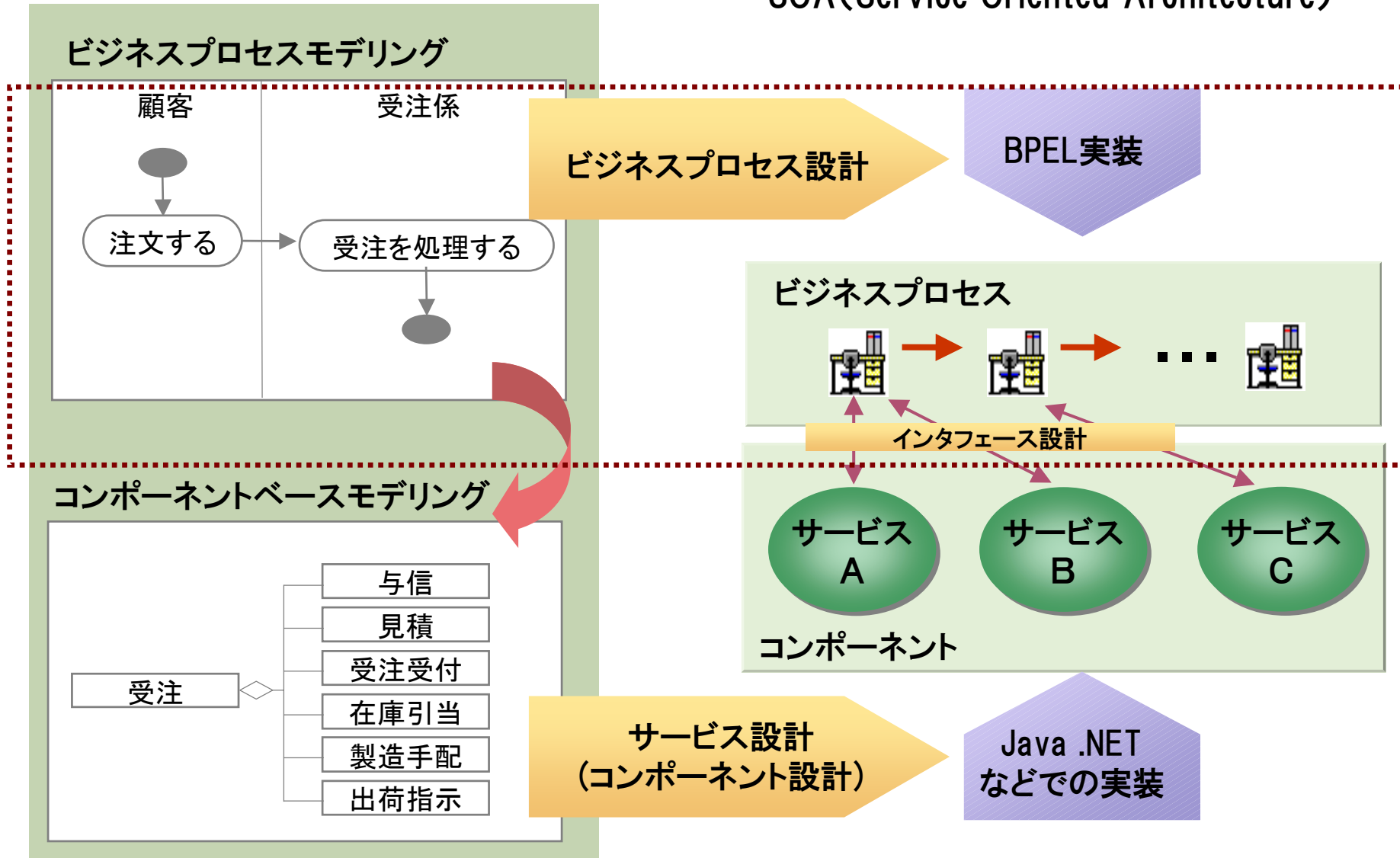


2-20. SOAでのサービス (コンポーネント) 開発プロセス



2-20. ビジネスプロセス設計とI/F設計を追加

SOA(Service Oriented Architecture)



2-21. 設計ガイドの策定(参考ページ)

● 設計ガイドの策定



詳細設計・実装における成果物の属人性を排除し、品質・開発効率を平準化

設計方針

- ・ 実装の共通化指針
- ・ 共通部品利用方針
- ・ 設計基準
- ・ 信頼性設計／性能設計

処理方式基準

- ・ ビジネスプロセス処理方式(フロー制御／イベント処理／…)
- ・ 他システム接続方式(MQ／FTP／DB／他社パッケージ／他システム外部I/F連携／…)
- ・ ジョブ制御(ビジネスプロセス起動方式)

運用設計

- ・ SOA基盤の開始・終了運用
- ・ ログ運用(ログ一覧／ログ設計／出力フォーマット)
- ・ 稼動統計取得運用(取得項目／取得方式)
- ・ メンテナンス運用(縮退運用／定義変更／デプロイ)
- ・ 障害監視・回復運用(プロセス・メッセージ監視／障害時運用／バックアップ・リストア)

成果物管理

- ・ 定義の移行(開発環境→検証環境／検証環境→本番環境)
- ・ 成果物のバージョン管理
- ・ 設計ドキュメント管理

- SOAの導入においては、BPEL,ESB,WSといったアーキテクチャを優先するのではなく、適用形態、適用方針などを、まずは明確にすべき
- そのためにはSOAの目的や効果をキチンと捉えておくことが重要
- SOAはサービスとビジネスプロセスという一般的なキーワードに拘らず、段階的なシステム統合の手段として捉えることも必要

- Oracle 及び Oracle EBS は、米国オラクル・コーポレーション及びその子会社、関連会社の米国及びその他の国における登録商標または商標です。
 - DataStageおよびQualityStageは、IBM Corporationの商標です。
 - Java及びすべてのJava関連の商標及びロゴは、米国及びその他の国における米国Sun Microsystems,Inc.の商標または登録商標です。
 - Microsoft® は、米国及びその他の国における米国 Microsoft Corp.の登録商標です。
 - SAPは、SAP AGのドイツ及びその他の国における登録商標または商標です。
 - R/3は、SAP AGのドイツ及びその他の国における登録商標または商標です。
 - WebSphereは、米国における米国International Business Machines Corp. の登録商標です。
 - Windows® は、米国及びその他の国における米国 Microsoft Corp.の登録商標です。
- その他記載の会社名、ブランド名および製品名は、それぞれの会社の商標、もしくは登録商標です。