

ITアーキテクト・サミット 2006 Summer
セッション【SB-3】

既存資産のSOA移行 (Legacy-to-SOA)

レガシーアプリケーションを近代化する
アプリケーションマイニング技術とは？

2006年8月3日
日揮情報ソフトウェア株式会社
常務取締役 岩田 アキラ

The Roadmap Strategy



アジェンダ

- BPMとSOA
- SOA開発の流れ
- サービス開発
- レガシーアプリケーションのサービス化
 - ▶ Legacy-to-SOA
- 既存システム再利用の問題点
- アプリケーションマイニング技術
- 影響分析ツール: ChangeMiner
- まとめ



BPMとSOAの共通目標

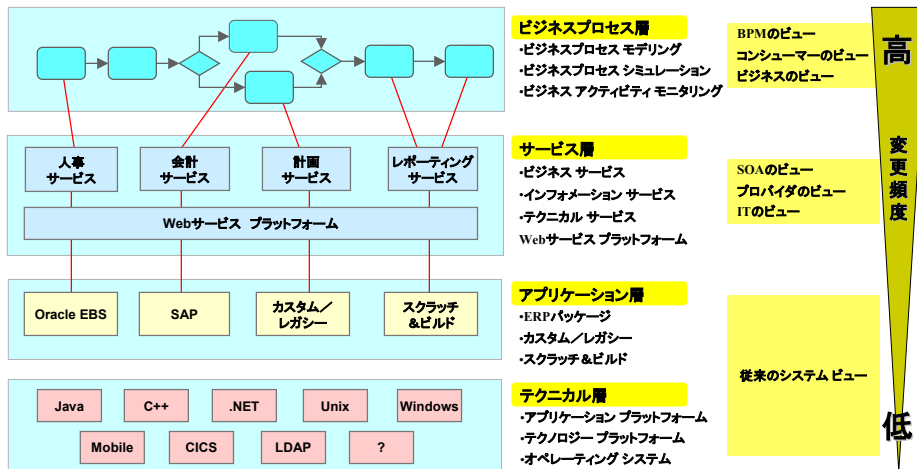
- BPMとSOAは同じ目標を共有する
- 共通目標: Agile Enterprise=俊敏な企業



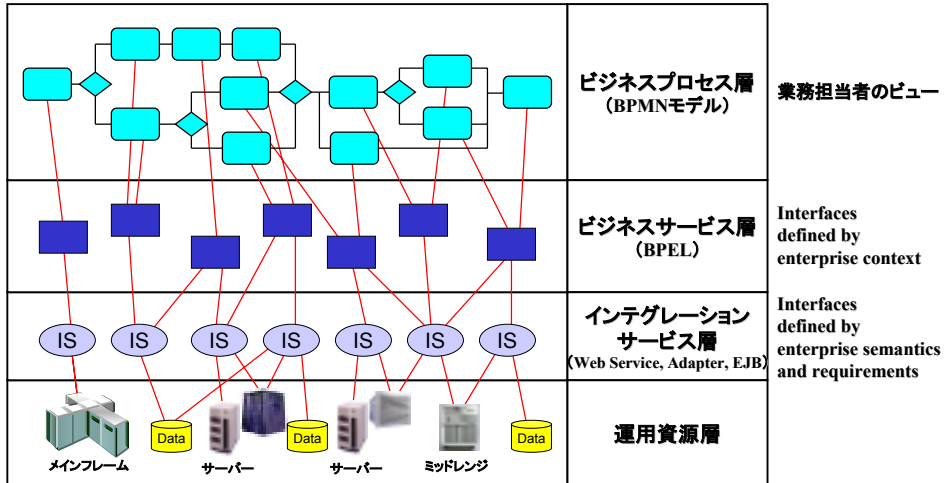
ソース: Gartner's Position on Business Process Management, 2006年2月



BPMとSOAの境界線



サービスの2層概念

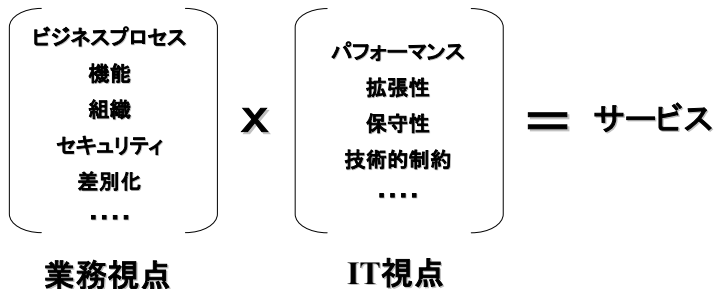


ソース: Michael Rosen

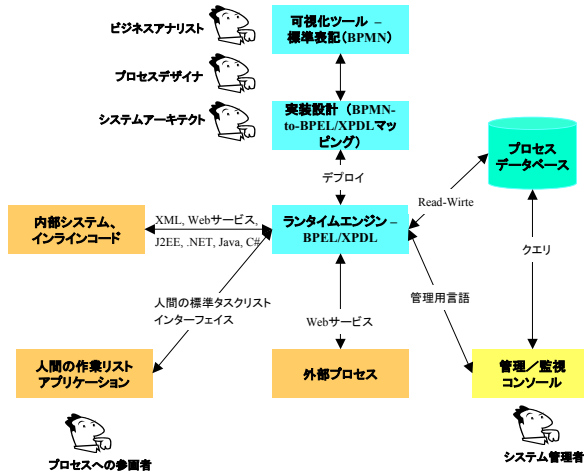


サービスの粒度

- 粒度論議をするとエンドレス
- 小さめに切り出しておけば、後で統合は容易
- 粒度を決められるセンスは、業務知識+IT知識



分析から実装までの流れ

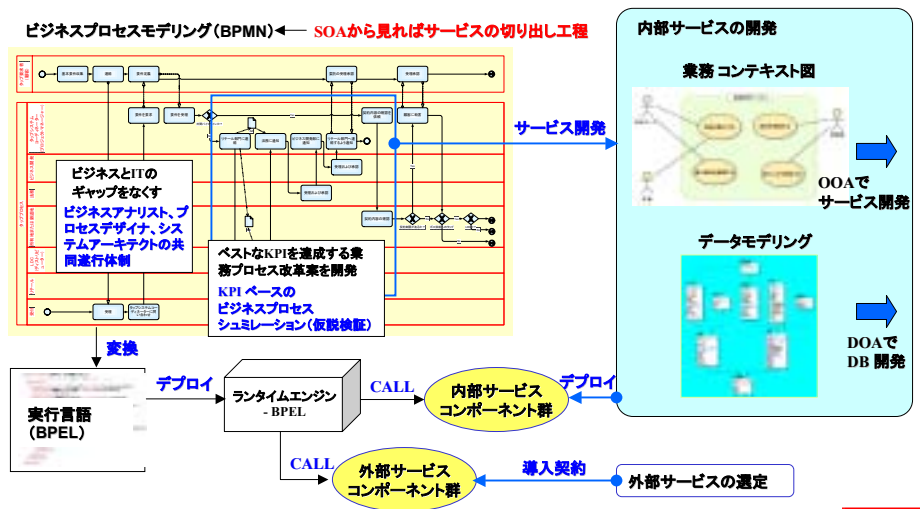


1. BPMNモデルを設計する。
2. BPMNモデルから BPEL/XPDLを生成する。
3. 要求されたサービス(人と内部および外部システムのインターフェイス)を開発する。
4. BPEL/XPDLコードとエンジンが必要とするインターフェイスをデプロイする。
5. 管理/監視インターフェイスを使ってプロセスの実行状況をモニターする。

ソース: Mike Havey, author of Essential Business Process Modeling

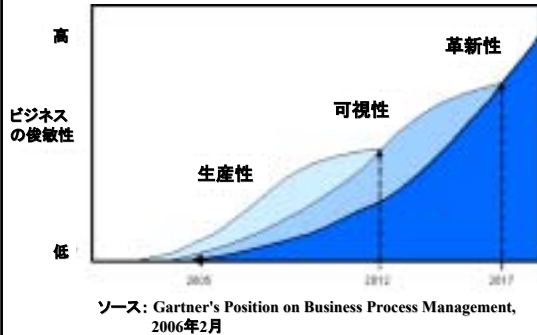


SOA時代の上流工程



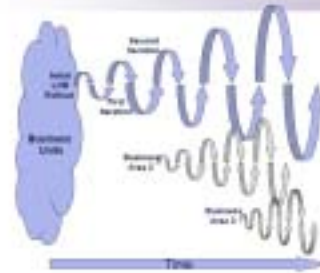
SOAへの取り組みは10年仕事

BPMが価値を生む発展段階

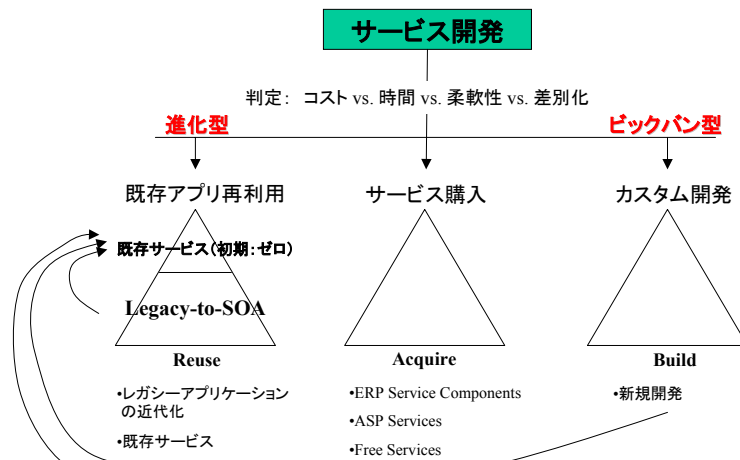


プロジェクトの進め方
=進化型アプローチ

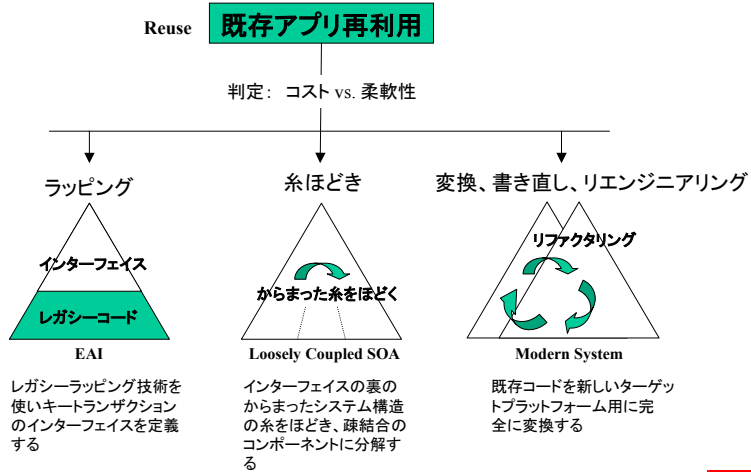
Think Big, Start Small - Iterate



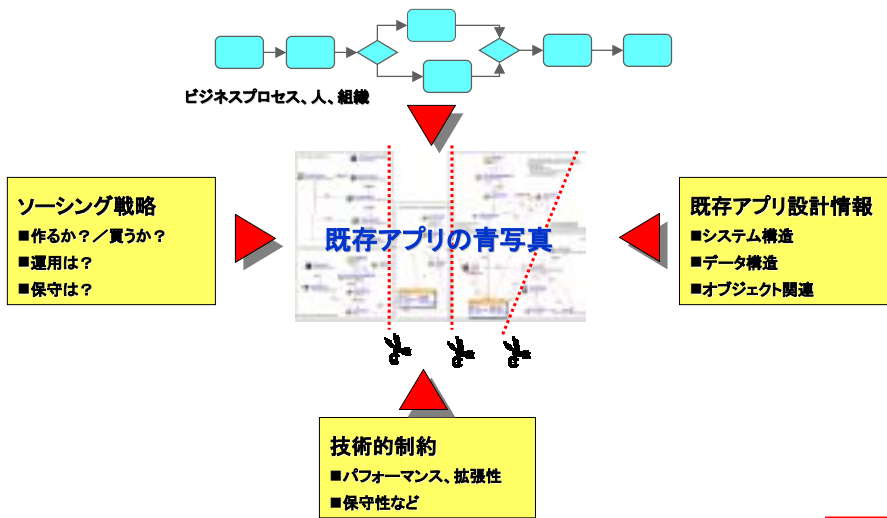
サービス開発の類別



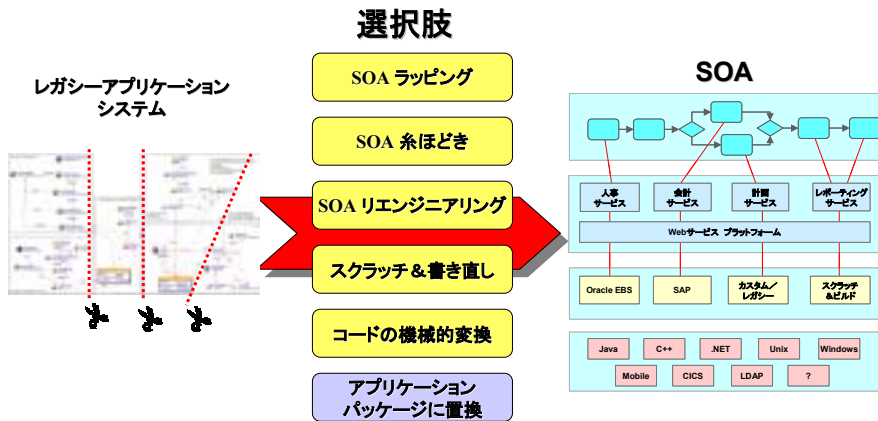
Legacy-to-SOAの類別



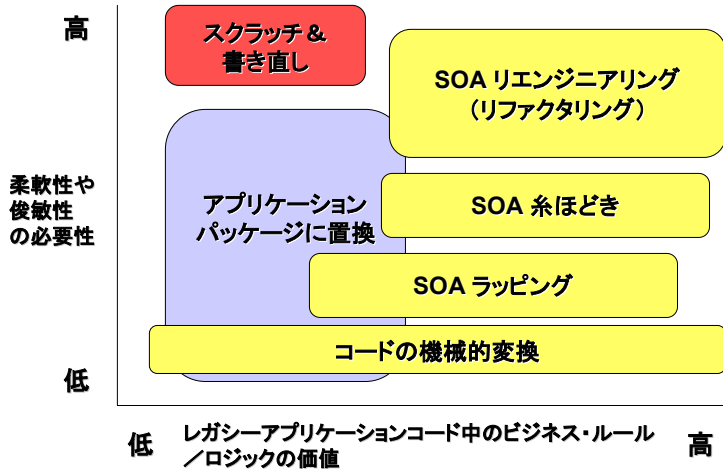
サービス境界線的设计



Legacy-to-SOA



戦略的選択

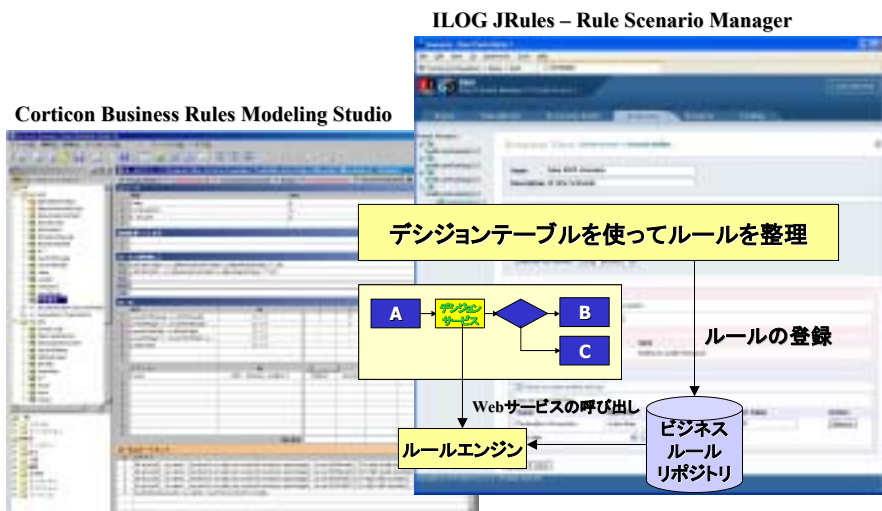


スクラッチ&書き直し

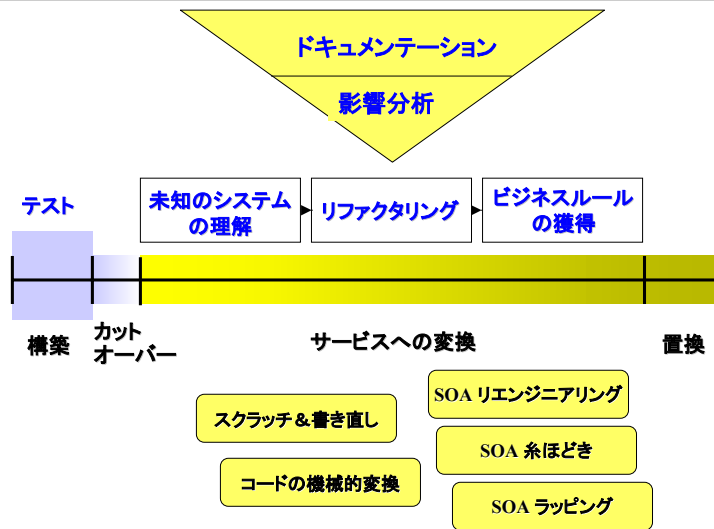
- SOAにとって最適な選択
 - SOA(柔軟性があり俊敏性があるサービスの提供)化
 - ターゲット・プラットフォームに最適な技術の取り込み
- しかし
 - コストがかかる
 - 時間がかかる
 - 失敗するリスクが高い
- 予算と時間が許せば選択したい
- コードレス・アプローチによる解決策
 - BRMS(ビジネスルール管理システム)の採用



代表的なBRMS製品とビジネスルールモデリング



サービスへの変換プロセス



レガシーアプリケーションのサービス化の障壁

- 既存システムの知識／ドキュメントの欠如
 - 開発責任者の不在
 - ベンダサポートの終了
 - 2007年問題
- 知識の継承
- 資産(コード)の継承
 - プログラムコードの棚卸しが必要
- サービス設計/実装技術の経験不足

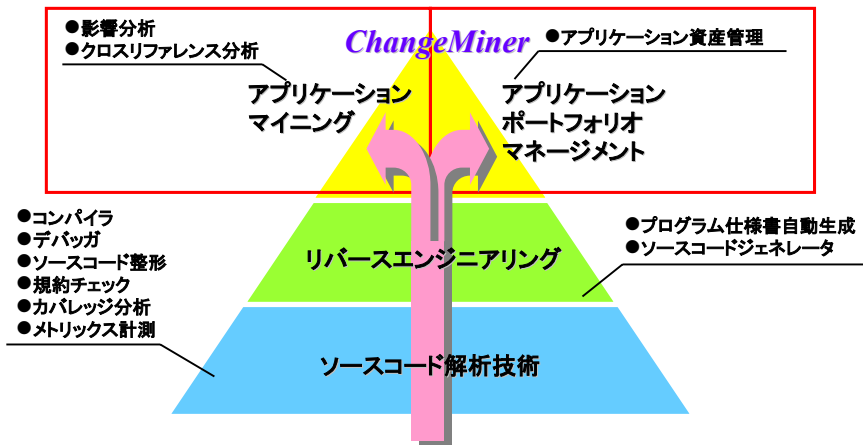
障壁の撤去:レガシーアプリケーションの設計情報整備

□ マイニング(採掘)の実行

アクション事項	内訳
アプリケーション 棚卸し管理	<ul style="list-style-type: none"> □ 管理基準によるプログラム棚卸し一覧管理 □ プログラムと関連したDB カタログ情報管理
影響分析管理	<ul style="list-style-type: none"> □ プログラム内部構造情報、ソース連係 □ プログラムとプログラムの呼び出し関係 □ プログラムとDBテーブル、カラムの影響分析, CRUD Matrix □ 他システムとの関連情報分析
標準化及び ドキュメント化支援	<ul style="list-style-type: none"> □ プログラムコーディング標準遵守可否及びコメント標準検査 □ 棚卸し、影響分析、統計資料ドキュメント化支援 □ 多様な形式のドキュメント支援
システム 管理指標設定 (アプリケーション ポートフォリオ管理)	<ul style="list-style-type: none"> □ 品質関連指標(コンポーネント再利用率、プログラム複雑度)管理 □ 管理及び企画管理指標



ソースコード解析技術を基盤とした2つの新しいツール分野



ソース: ITソリューションフロンティア2006年5月号「ITマネージメントのためのアプリケーションポートフォリオ管理」
http://www.nri.co.jp/opinion/it_solution/2006/pdf/IT20060508.pdf



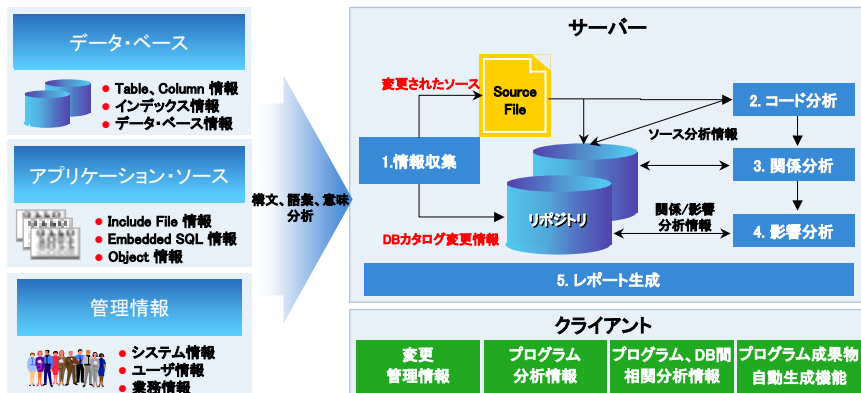
ソースコードという「動かぬ事実」から アプリケーションを客観的に評価

- 影響分析 (Application Dependency Mapping)
 - 受益者: PMなどのIT管理者層、開発者、保守業務管理者/担当者
 - プログラム構造、オブジェクト関連性の可視化
 - 未知のコードの探索、ナビゲーション
 - ドキュメントの代替

- アプリケーション資産管理 (Application Portfolio Management)
 - 受益者: CIO、PMなどのIT管理者層
 - 適切なIT資産の品質評価
 - アウトソースの管理レベル向上

影響分析ツール: ChangeMinerの仕組み

アプリケーションソースコードの構成要素をコンパイラレイヤで分析することにより、波及される部分の自動検出。さらに、DBカタログに直接アクセスし、構成情報をリポジトリへ自動抽出。この情報を基に波及される影響要素の相関関係分析を行い、**アプリケーションの変更管理、影響分析、成果物の自動生成、品質管理**を可能にする。

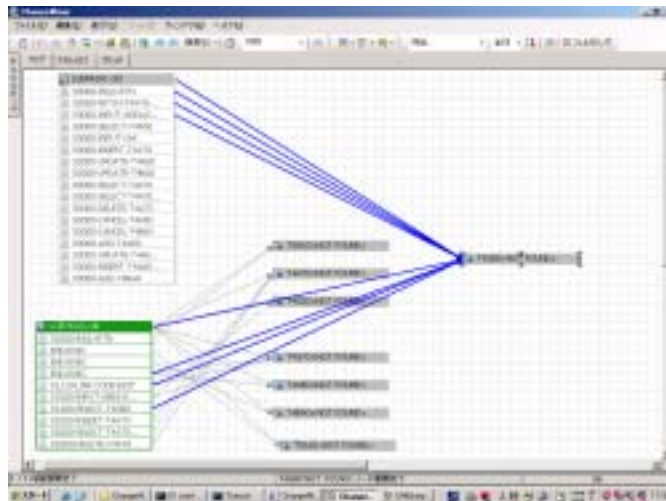


ChangeMiner: オブジェクト関連マップ (Javaの例)



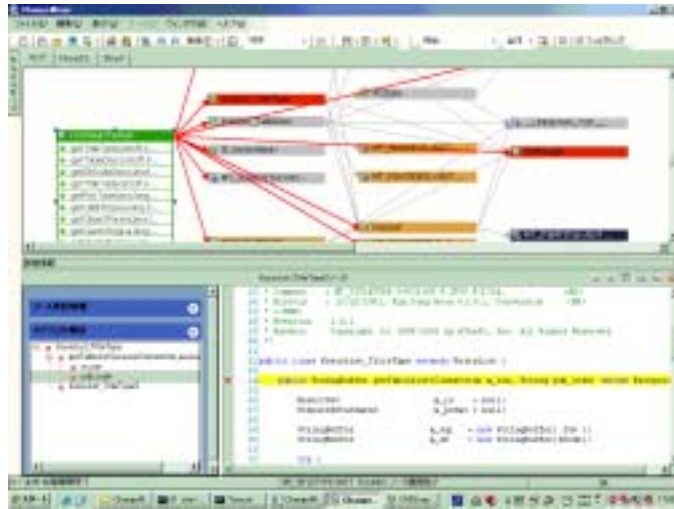
J-SYS
SOFTWARE

ChangeMiner: オブジェクト関連マップ (COBOLの例)



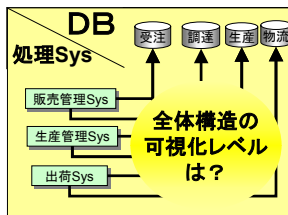
J-SYS
SOFTWARE

ChangeMiner: オブジェクトとコードの紐付け



ChangeMiner: データとプロセスの紐付け

全システムの全体図(イメージ)



ChangeMiner

CRUD Matrix(相関図)

処理Sys	受注		調達		生産			出荷			
	商品	納期	価格	材料	納期	価格	製品	工程	仕掛	納先	検収
販売管理Sys	C						C				
商品管理SS	U		C								
受注管理SS	R C R		R C								
.....SS	D	D	D							D	
生産管理Sys				C	C	C			C		
調達管理SS		R		U	U	U	R	R	R	R	
工程管理SS		R		R	R		R	C	U	R	
原価管理SS	R	R	R	R	R	R	R		R	R	R
.....SS				U	R	U	U		R	C	
出荷Sys											C
発送管理SS	R	R	R							U	R
.....SS	R						R			R	R

※ CRUD(クラッド)とは、C:生成、R:参照、U:更新、D:削除

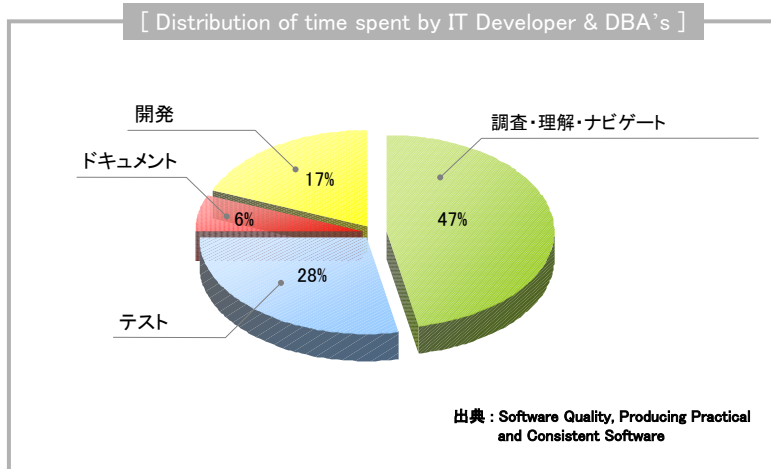


ChangeMiner: アウトプット

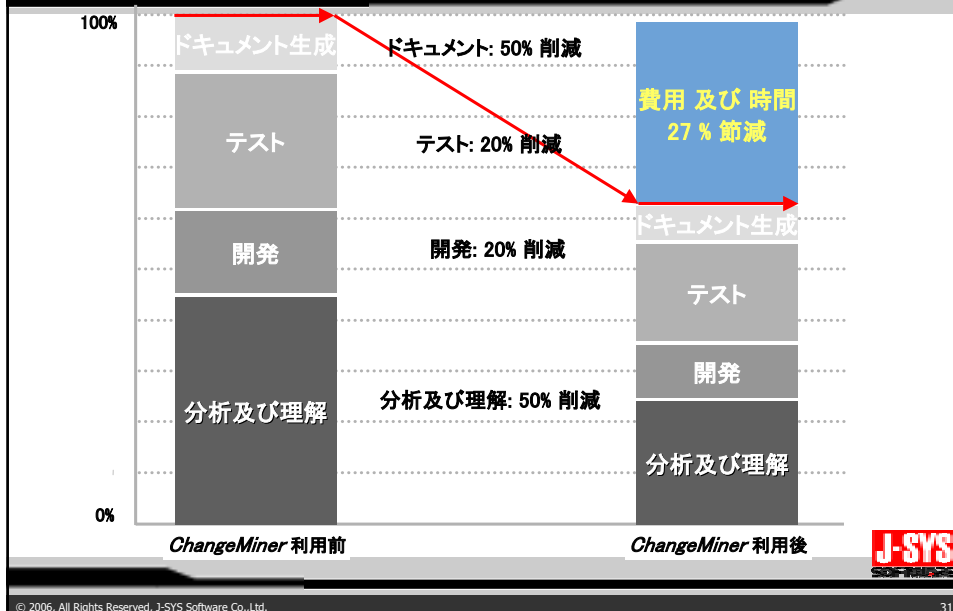
ソース・コード分析	相関分析	変更管理及びアウトプット生成
 <ul style="list-style-type: none"> ・プログラム別コード分析 ・プログラム別テーブル分析 ・プログラム別モジュール分析 ・業務別プログラム分析 ・担当者別プログラム分析 ・プログラム/モジュール検索 ・プログラム・モジュール統計 ・業務別テーブル統計 (兄弟製品: APM) 	 <ul style="list-style-type: none"> ・プログラム対テーブル相関分析 (CRUD Matrix/SQL照会) ・プログラム対モジュール相関分析 ・業務やプロセス別Entity相関分析 ・プログラム構造図/ダイアグラム 	 <ul style="list-style-type: none"> ・プロジェクトWorkFlow 管理 (担当割り当て及び照会) ・プログラム開発/変更詳細状況管理資料 ・プログラム開発進捗管理 ・プログラムテスト管理 ・IPO レポート ・プログラム仕様書及びテーブル管理資料 ・SQL 詳細リスト ・プログラム変更詳細リスト ・プログラムとテーブル相関図 (CRUD Matrix) ・プログラム構造図/ダイアグラム <p>(これらの機能はChangeFlow と Flow Chart 組み合わせたり、カスタマイズにて実現いたします。)</p>



メンテナンス作業の構成(一般論)



影響分析ツールの導入効果（韓国現代自動車の事例）



まとめ

- なぜ、SOAか？
 - 変化に強いITを目指す
- その方法は
 - 既存システムの再利用(進化型)が中心
 - ビックバン型は実行困難
- Legacy-to-SOAの実行障壁
 - 既存システムの知識/ドキュメントの欠如
- SOAへの取り組みは10年仕事
 - First Small→iteration →iterationの進化型で発展させる
- まず行うべきことは
 - 既存システムの資産整備とメンテナンスの近代化
 - ✦ 手段: アプリケーションポートフォリオ管理、アプリケーションマイニングツールの導入

ご清聴ありがとうございました

製品に関するお問合せ先：
日揮情報ソフトウェア株式会社 営業本部
TEL : 03-5778-0262
E-Mail : prodpost@jsys.co.jp

製品資料の参照先：
<http://www.jsys-products.com/product/ChangeMiner/>
<http://www.jsys-products.com/product/MetaMiner/>

THE INFORMATION SECURITY

