
当社の開発方法論と SIベンダーへの期待

E-AGILITY Conference 2010

協和発酵キリン株式会社

情報システム部 中山嘉之 (2010.11.19)

■はじめに

- 協和発酵キリンの概要
- 情報システム部門の組織
- パートナーとの役割分担

■エンタープライズ・アーキテクチャ

- 当社が目指すアーキテクチャ
- EAと当社のモデリングツール
- ※各種モデルによる図表現

■当社の開発方法論

- ☆開発全般のフェーズ概要
- 中計～準備～方向付け<上流>
 - 準備フェーズの手順
 - 方向付けフェーズの手順
- ※開発構想書作成のポイント
- ※ユースケース図への展開

※RFPへの記述内容

※ベンダーとの契約形態と発注方法

- 推敲～作成～移行<下流>
- ☆推敲フェーズ以降の反復計画
 - 推敲フェーズの手順(パッケージ)
 - 推敲フェーズの手順(スクラッチ)
- ※推敲フェーズのマイルストーン
- 作成フェーズの手順
- ※SIベンダーとのコミュニケーション

■SIベンダーへの期待

- ウォーターフォールの特徴と課題
- 反復型の必然性とリスクへの対応
- SIベンダーへのお願いと期待

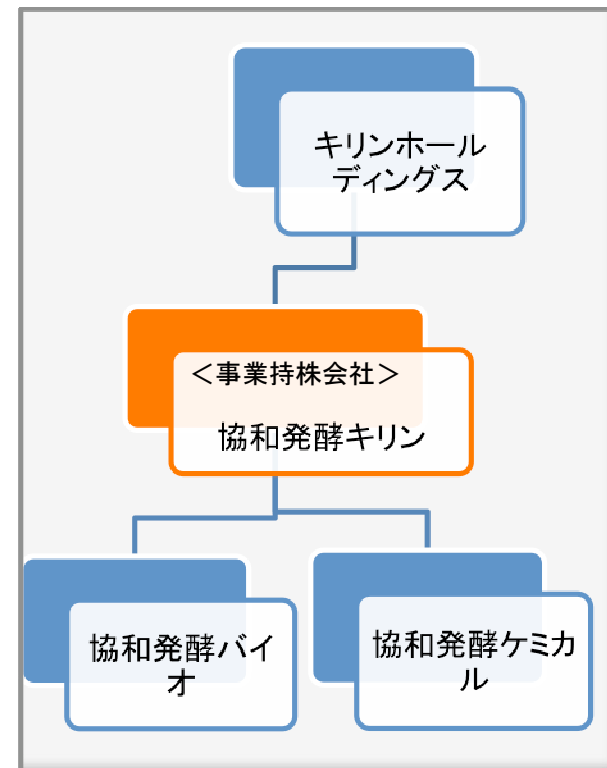


<はじめに>

協和発酵キリンの概要

KYOWA KIRIN

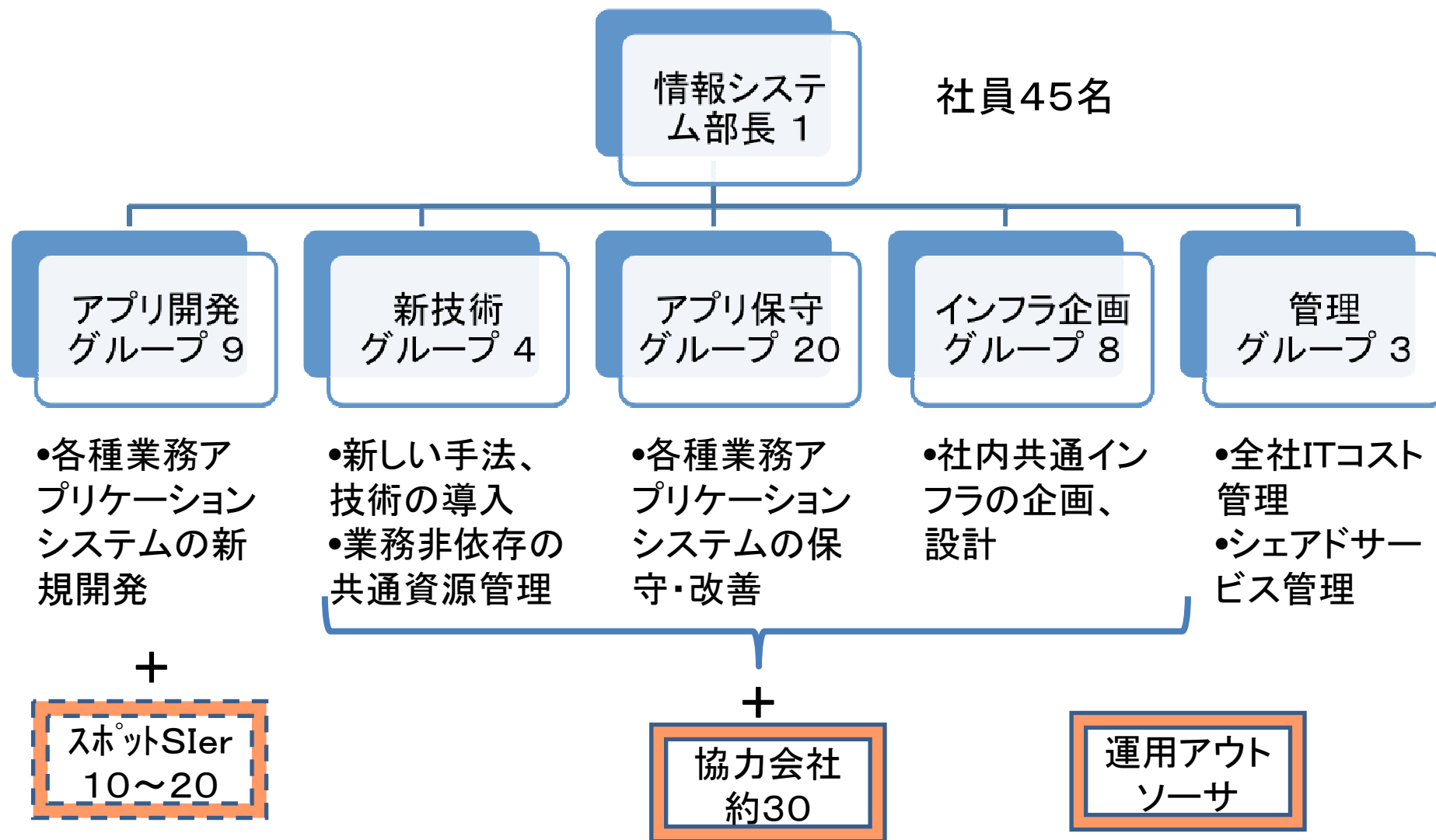
- **設立** : 2008年10月1日
- **資本金** : 26,745百万円
- **従業員数** : 7,436名(連結、2009年12月31日現在)
- **事業内容** : 医療用医薬品の製造・販売。バイオケミカル、化学品をグループ事業として展開。親会社キリンホールディングス。
- **売上高** : 2009年度 3,091億(4~12月、連結)



• **事業ビジョン**: がん、腎、免疫疾患を中心とした領域で、抗体医薬を核にした最先端のバイオテクノロジーを駆使して、画期的な新薬を継続的に創出し、開発、販売をグローバルに展開することにより、世界の人々の健康と豊かさに貢献する、日本初のグローバルスペシャリティファーマとなる。

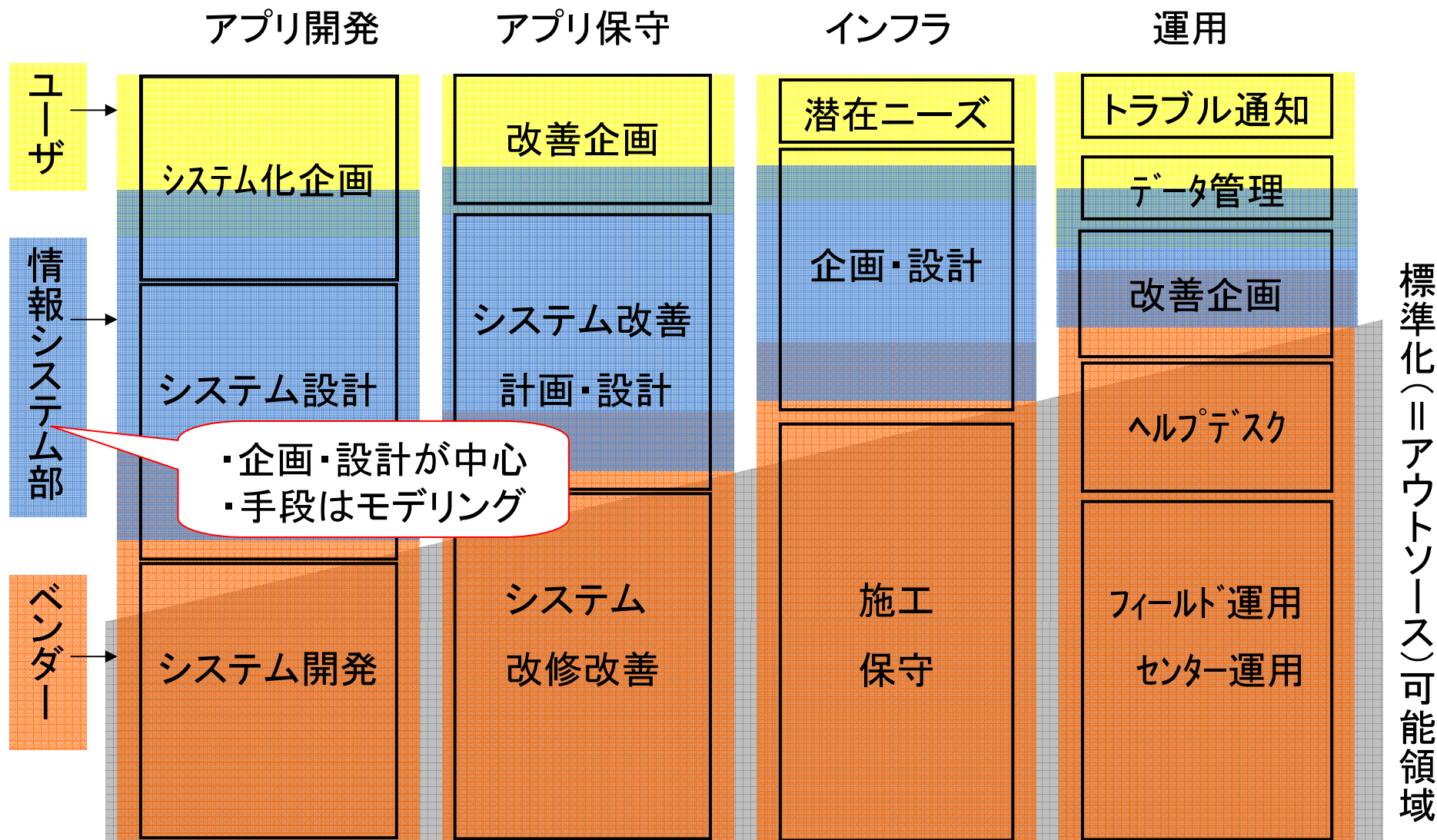
情報システム部門の組織

KYOWA KIRIN



パートナーとの役割分担

KYOWA KIRIN



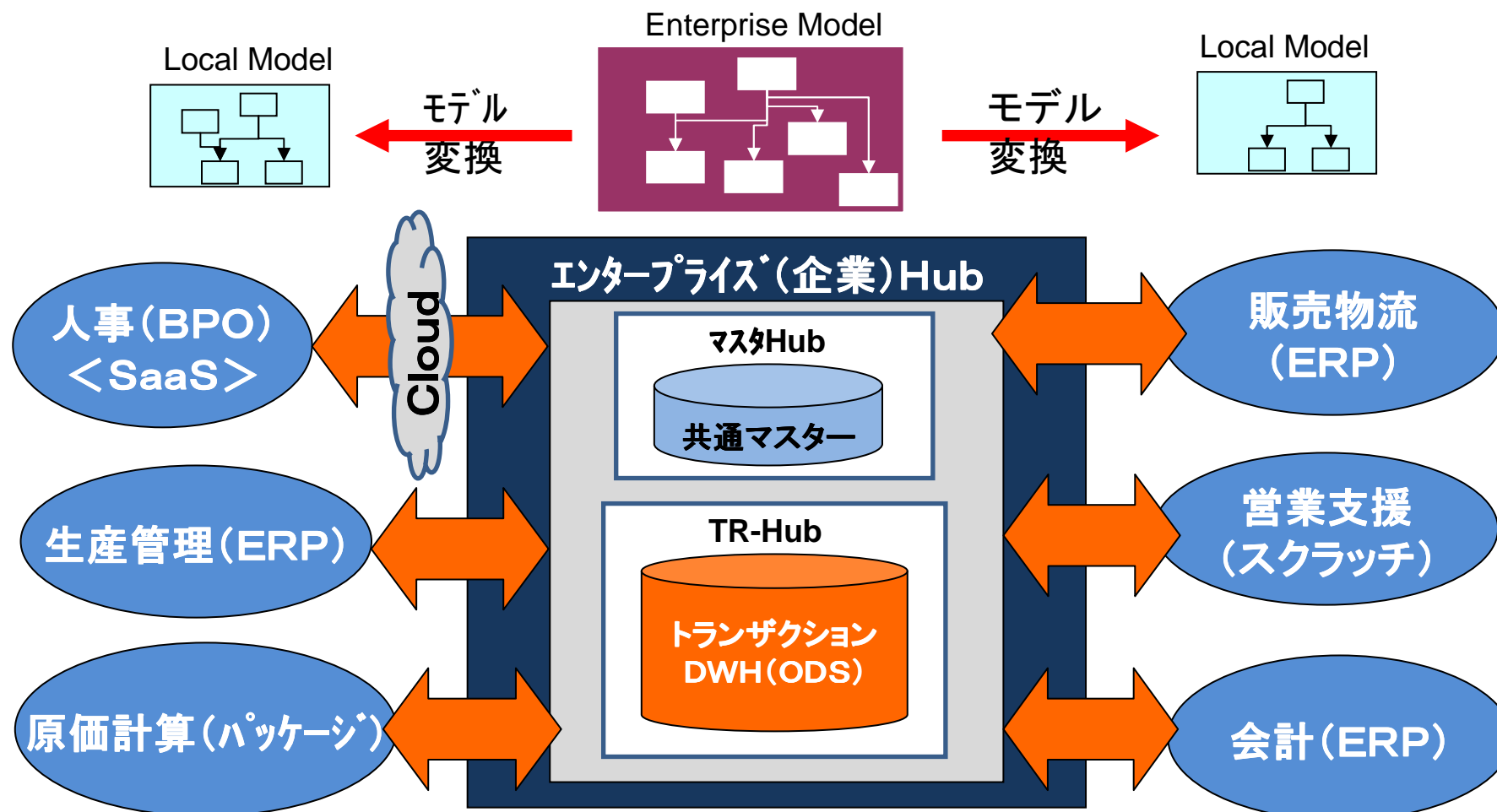
《エンタープライズ・アーキテクチャ》

- 当社が目指すアプリ・アーキテクチャは自社のエンタープライズ・(データ)モデルが中心で、市販ERPの(プロセス)モデルが中心ではない。
- アプリ開発(特に上流)では常にモデルを中心とし、モデル図は保守にも及ぶ永続的な成果物となる。

当社が目指すアーキテクチャ

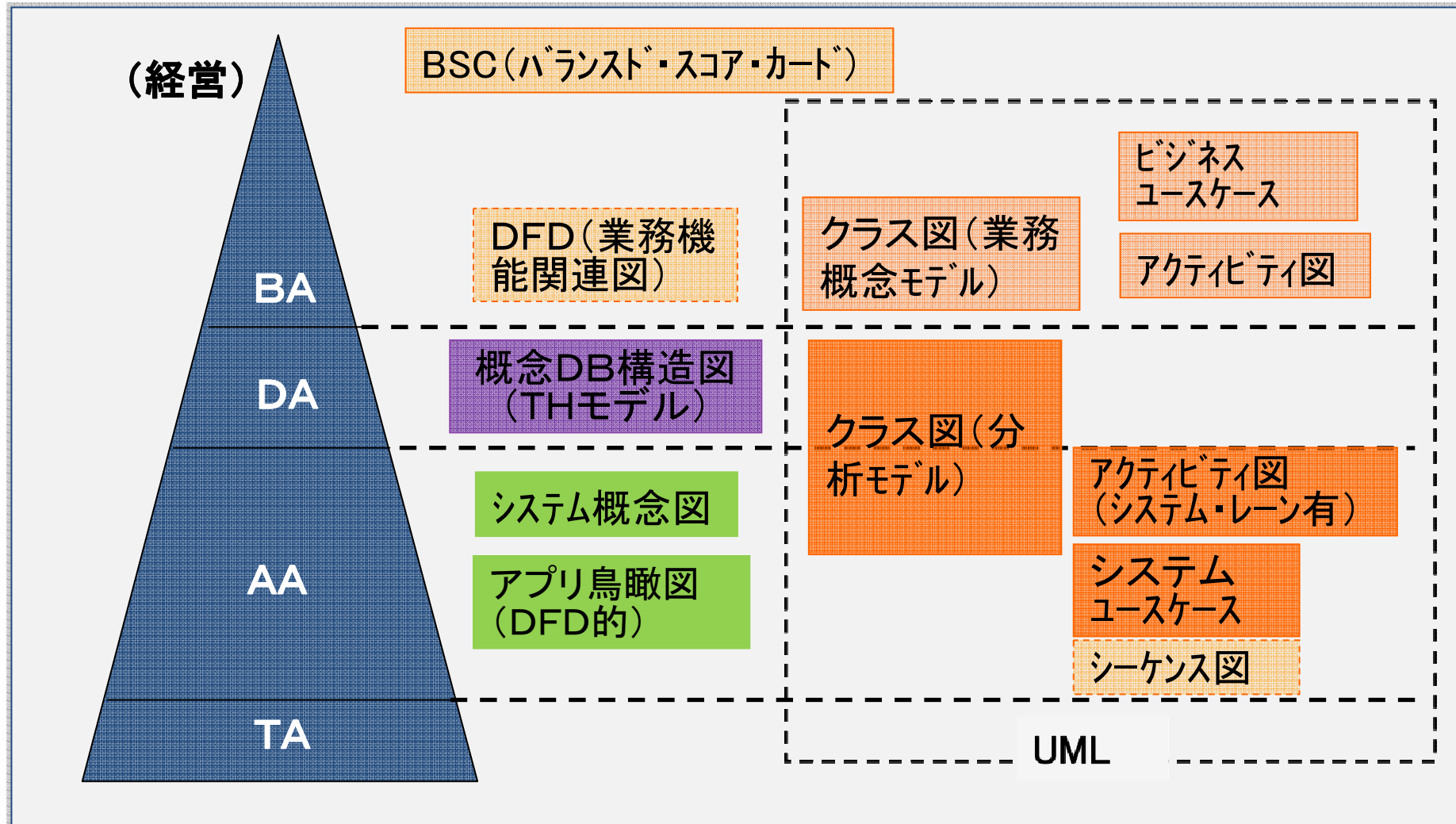
KYOWA KIRIN

- 企業独自のモデルに基づいたエンタープライズHubが中心。
- ERP等の周辺処理コンポーネントは、取り替え可能！

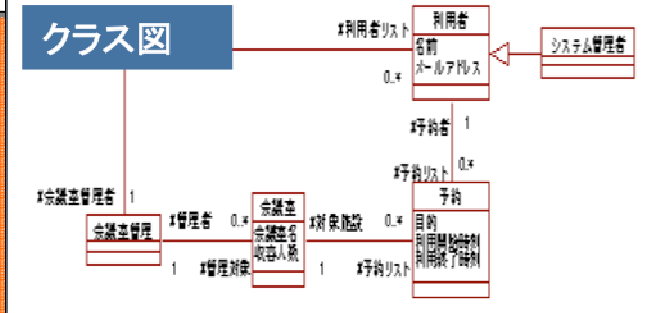
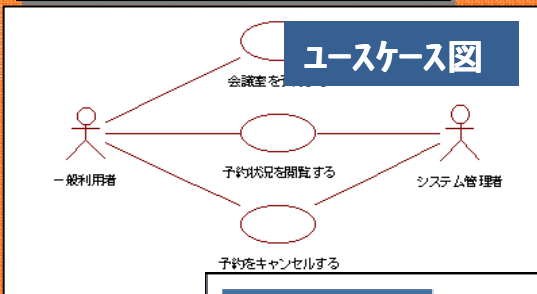
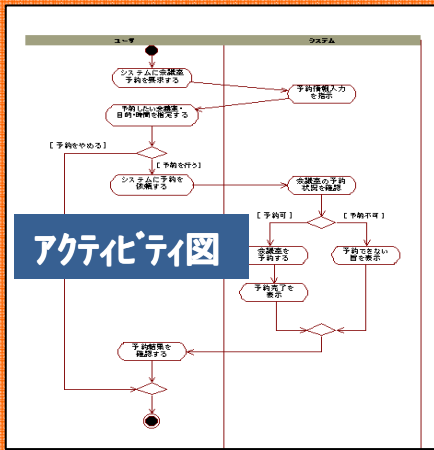


EAと当社のモデリングツール

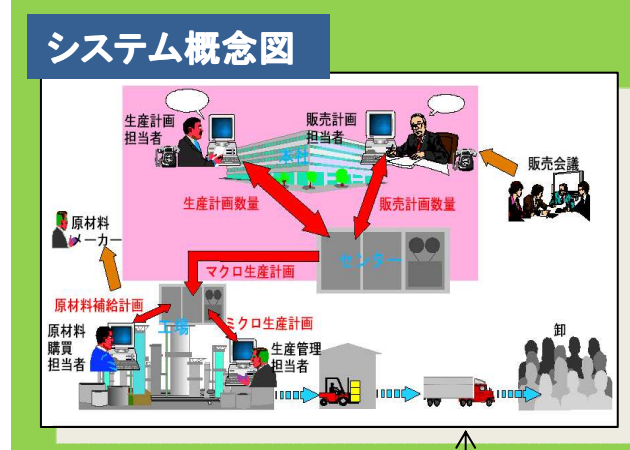
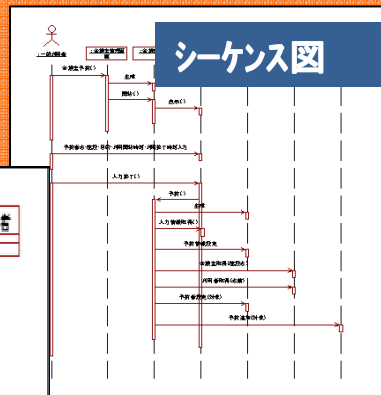
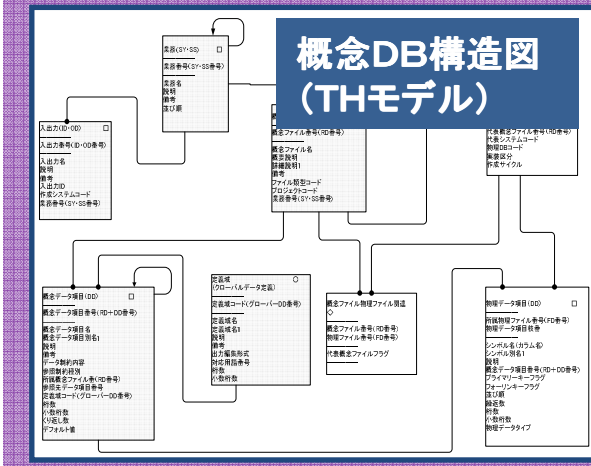
KYOWA KIRIN



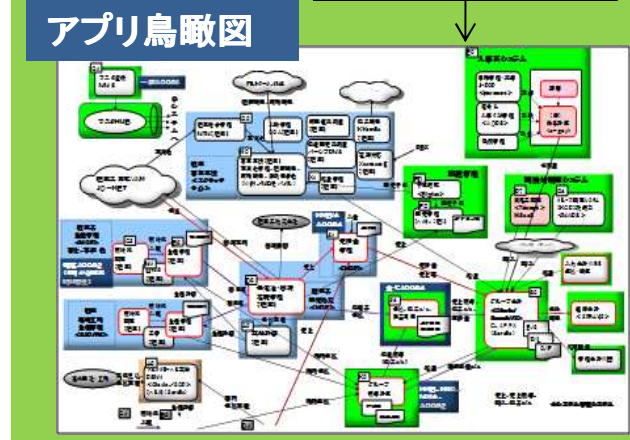
※各種モデルによる図表現



UML



当社独自モデル

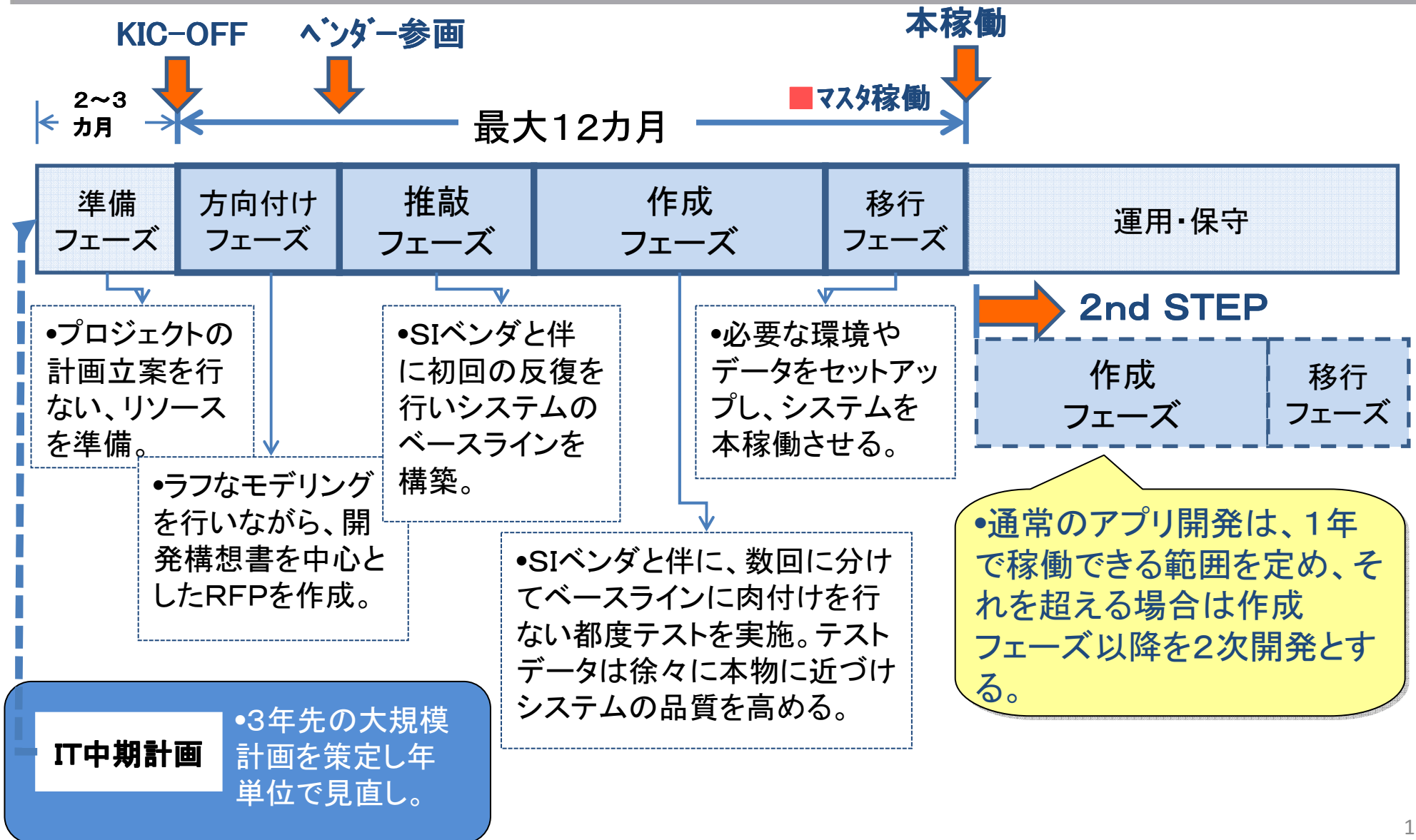


<当社の開発方法論>

•当社ではオブジェクト指向の台頭を背景に、2004年にRATIONAL(現IBM)のコンサルテーションを導入。UMLモデリングとともに、RUPによる反復型開発を実アプリで体験。翌2005年、従来のウォーターフォール手法に代わって、RUPを自社流にカスタマイズした反復型開発標準を制定、翌2006年、上流工程に特化した方法論を追加し、現在に至る。

RUP: Rational Unified Processのとは、IBM社ラショナルブランドのオブジェクト指向型ソフトウェア開発プロセス

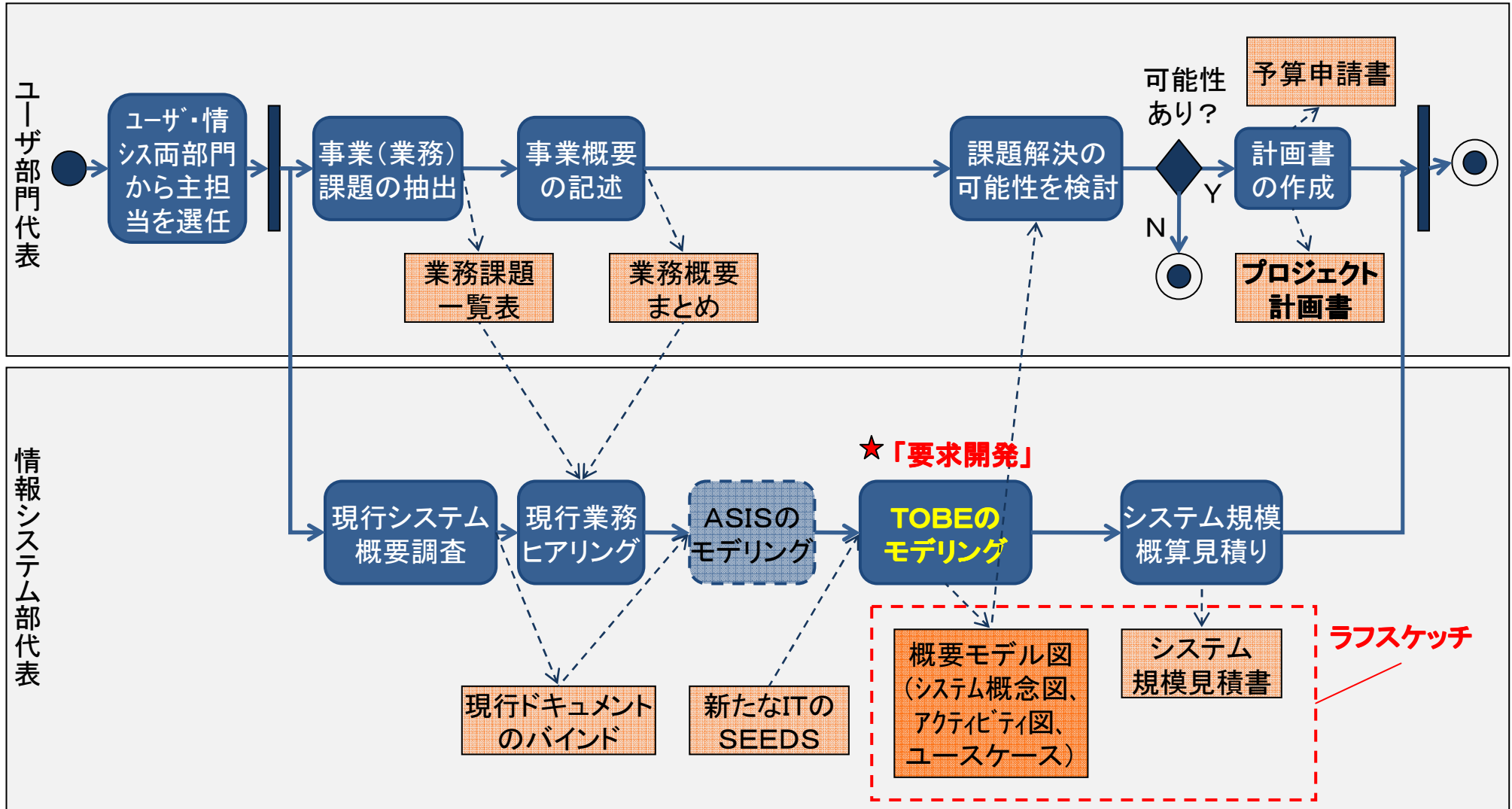
☆開発全般のフェーズ概要



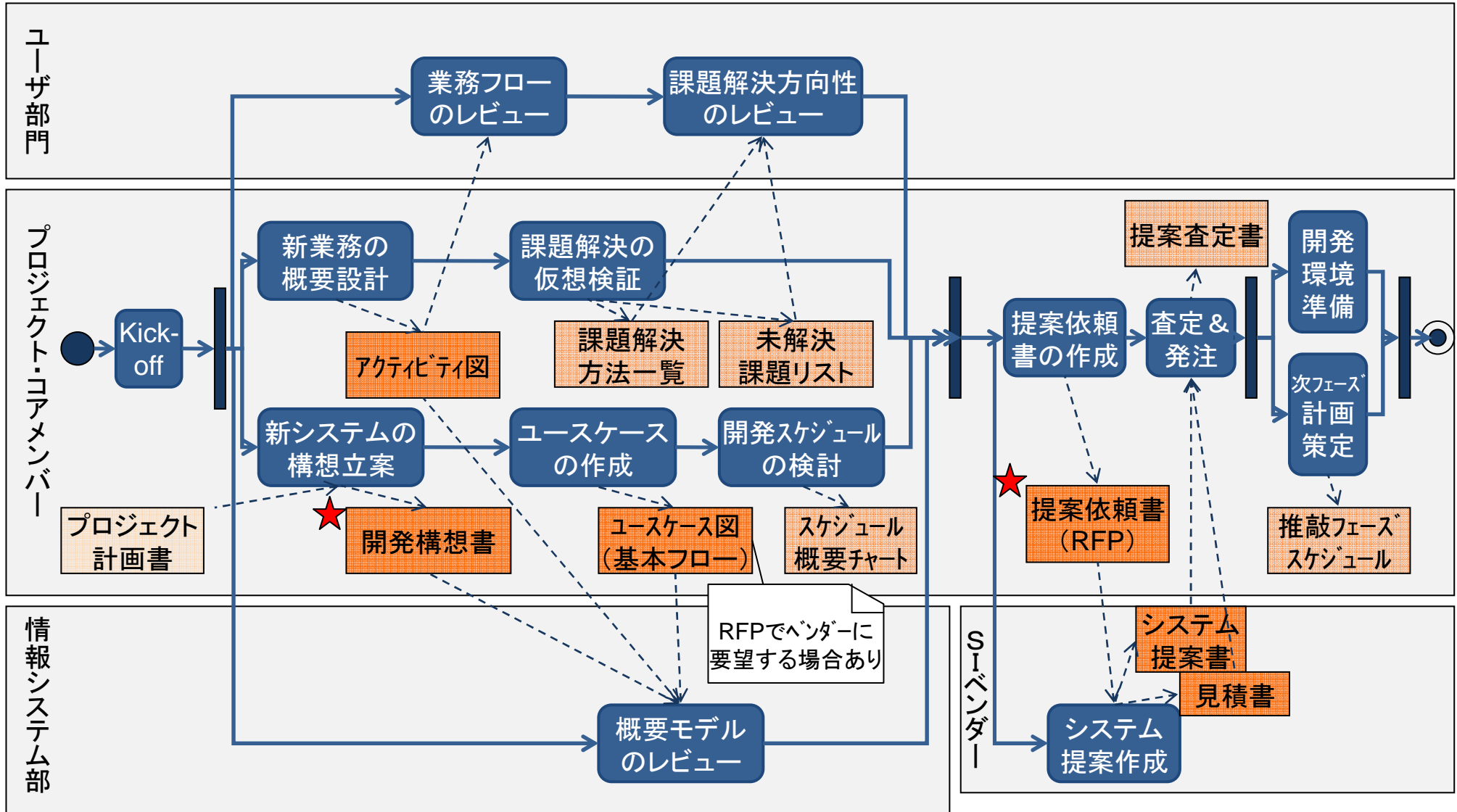
《中計～準備～方向付け》

•1／3Yで作成されたIT中画に基づき、毎年各プロジェクトが起案される。まず準備フェーズにて新システムのラフスケッチを行ない、プロジェクト計画を策定する。プロジェクトKICK OFFの後、最初の方向付け“フェーズで”開発構想書“の作成を行い、システムの概要設計を行なう。プロジェクトの成否はこの上流工程で決まる。

準備フェーズの手順



方向付けフェーズの手順



※開発構想書作成のポイント

1. システム化の範囲

- 開発スケジュール、優先順を考慮し、アイテム(What)・組織(Where、Who)・プロセス(How)の3つの軸で開発範囲(システムSCOPE)を明確化。

2. プロジェクトの位置づけ

- プロジェクト発足に至った社内外の環境要因や業務要件(Why)を、経営的視点から記述。また、現状業務の根本的問題とその影響及び、システム化による解決方向、有効性について記述。

3. 利害関係者とユーザ

- 利害関係者、ユーザという、システムに直接的、間接的に関係する登場人物についてその役割毎に抽象的に記述。

4. システムの概要

- 周辺システムも含めた当システムの位置付け及び、システム構成を図式化。文書記述は新システムの各機能詳細と、非機能要件等の前提及び、予算。

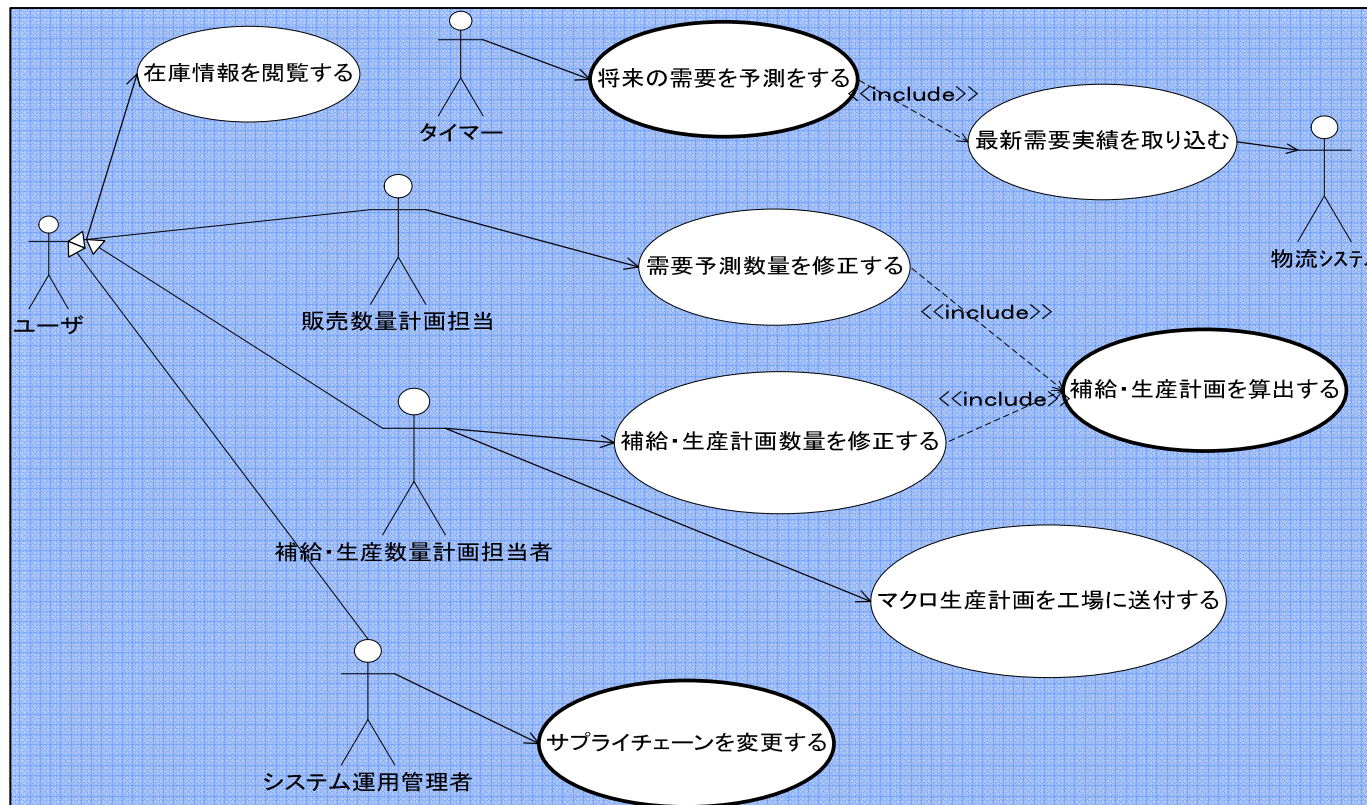
5. システムの実現手段(※)

- スクラッチ開発orパッケージ適用等の実現手段を社内標準に即して記載。
※標準のRUP開発構想書にはない

※ユースケース図への展開

- ・開発構想書の“ユーザ”をもとに“アクター”を描画（汎化されたアクターも）
- ・ “ ” “各種機能”をもとに“ユースケース”を描写（UCの数は適度に）
⇒あくまでユーザの視点に立ったシステムの“**使い方例**”を表現（＝外部設計）

➤ SCP(需給計画)システムのユースケース図 例



※RFPへの記述内容

- RFP提示内容

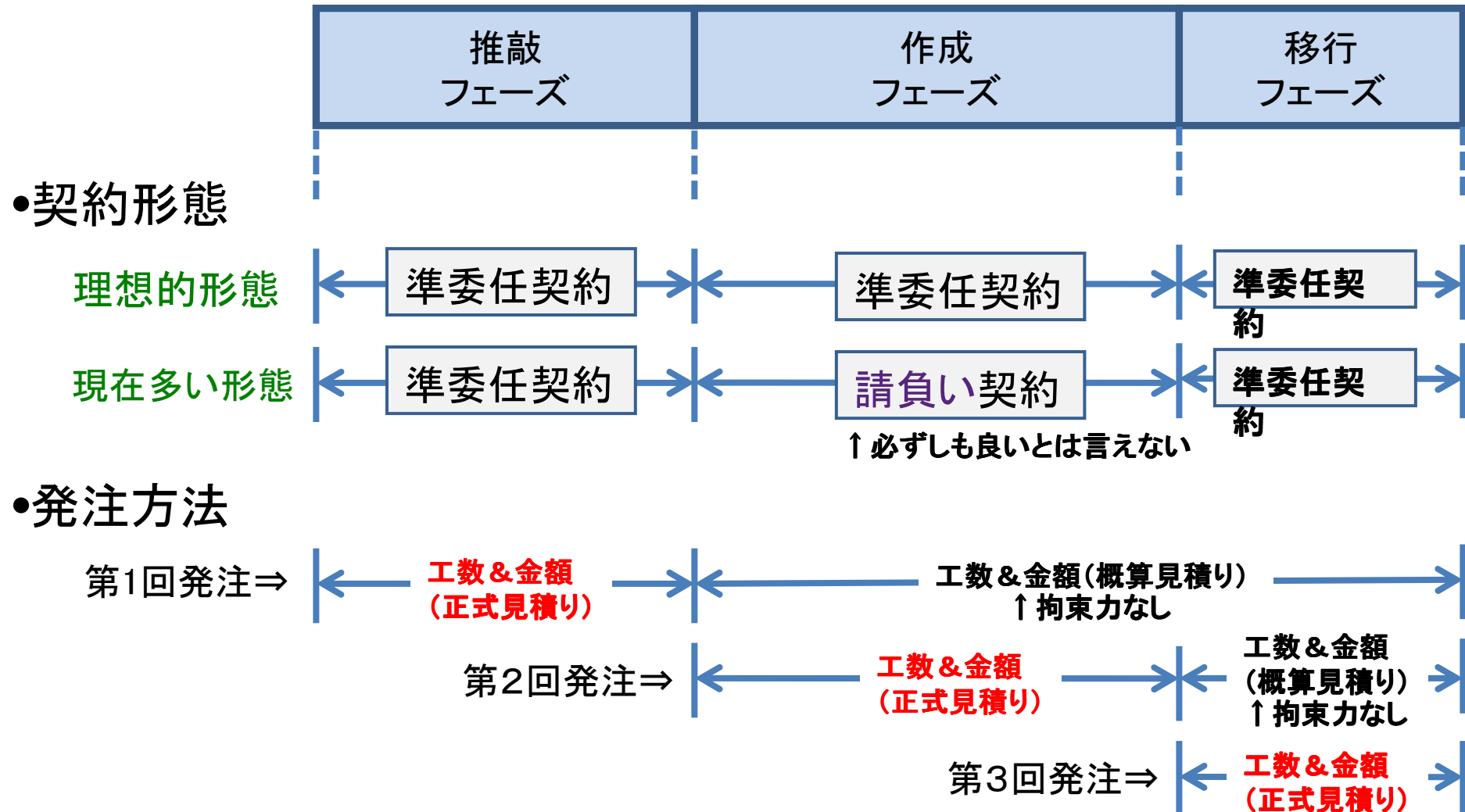
[本文] 1.システム化の概要 2.目的／効果(ビジネス要求) 3.プロジェクト体制図
4.プロジェクトのスケジュール 5.業務の概要(ASIS、TOBE)
6.システム機能要件のサマリ 7.前提となるインフラ 等

[添付資料]①システム概念図 ②業務アクティビティ図 ③開発構想書
④ユースケース図 ⑤未解決課題リスト ⑥開発スケジュール 等

- 提案依頼内容

- システム概要 : アプリケーションデザイン概要、ビジネス要求の解決方法
- システム設計 : 実現手段(スクラッチ or パッケージ)、実現方法(要件⇒実装のマッピング)、各種モデル図(パッケージ: 概念クラス図、概念ER図等、スクラッチ; ユースケースモデル(基本フロー迄)、概念クラス図など)
- 開発工数(推敲フェーズ～移行フェーズ迄のフェーズ別工数(SE、PG))
- インフラ概要: ハードウェア/ミドルウェア、クラウド環境の有無
- プロジェクト管理: プロジェクト体制、開発方法論、コミュニケーション/リスク管理の手法
- システム保守: 稼働後のシステム運用・保守体制

※ベンダーとの契約形態と発注方法

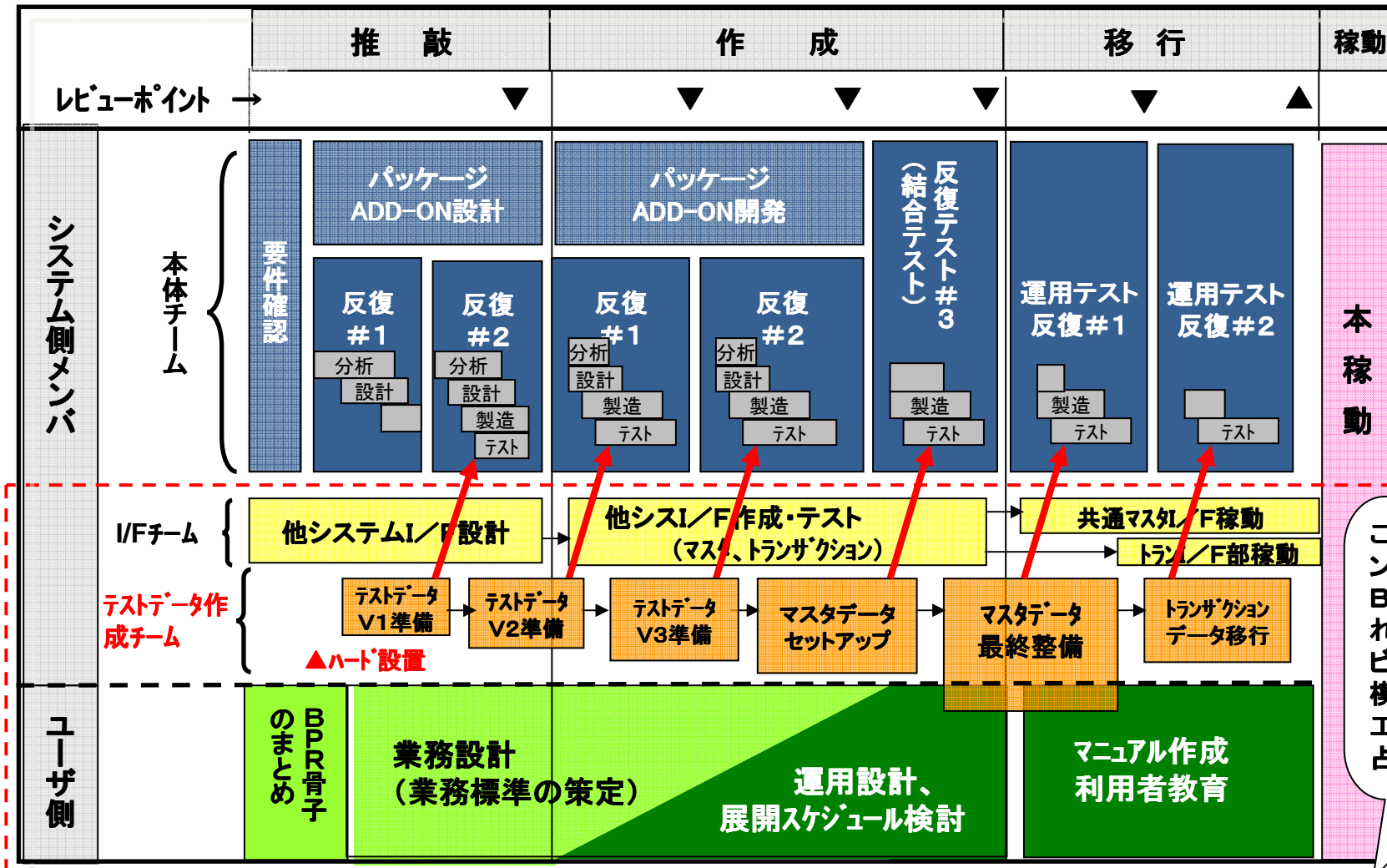


《推敲～作成～移行》

•SIベンダーが合流し開発下流工程が開始される。推敲フェーズから実際にシステムを稼働させながらの反復型開発が始まる。作成フェーズでは、推敲フェーズで完成したベースラインに肉付けする形で、本格的な反復が実行される。ここでのポイントはテスト行程にあり、とりわけ用いるデータがシステム品質を決定付ける。

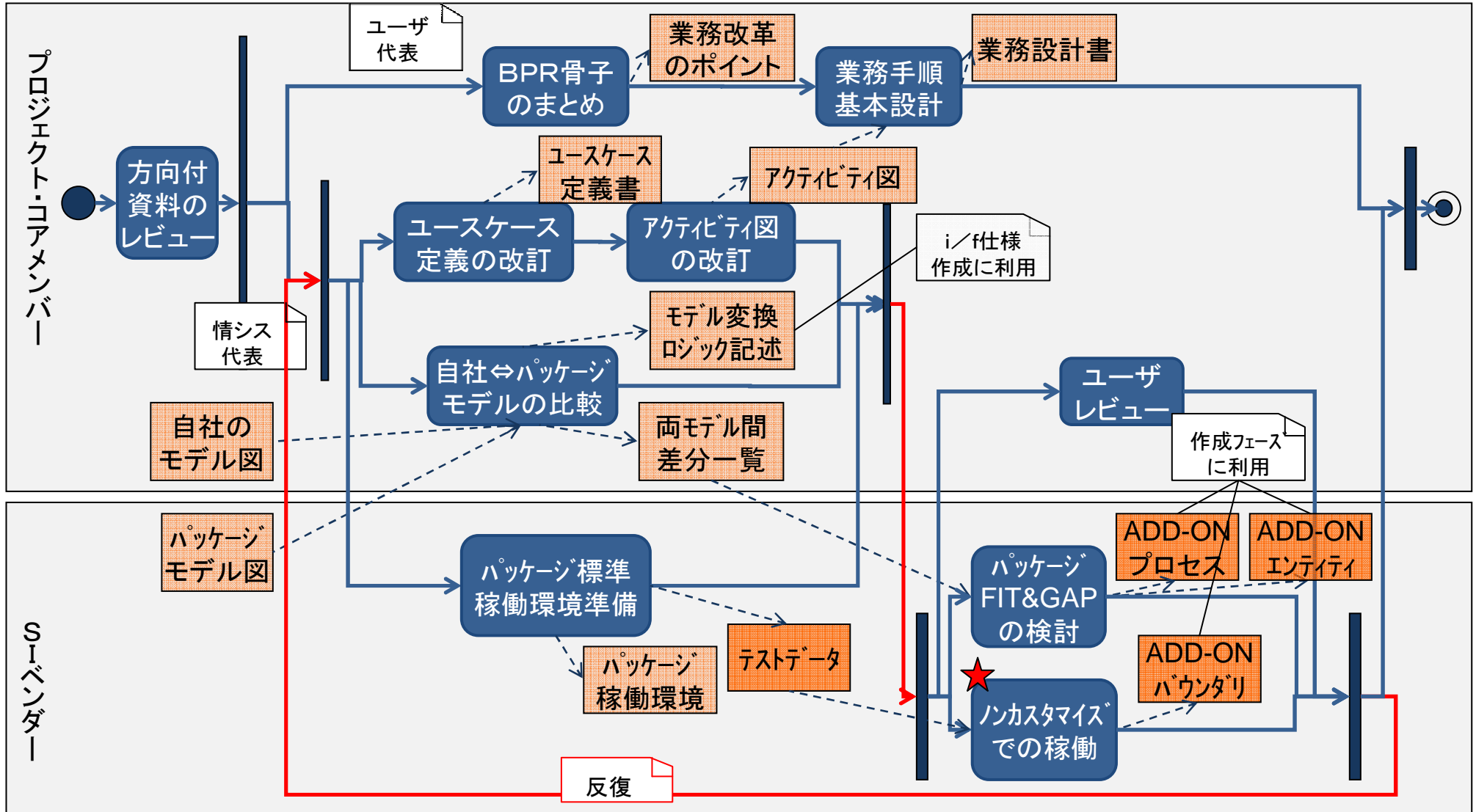
☆推敲フェーズ以降の反復計画

■ ”リスクドリブン“ & ”テスト駆動“ の反復型開発が標準

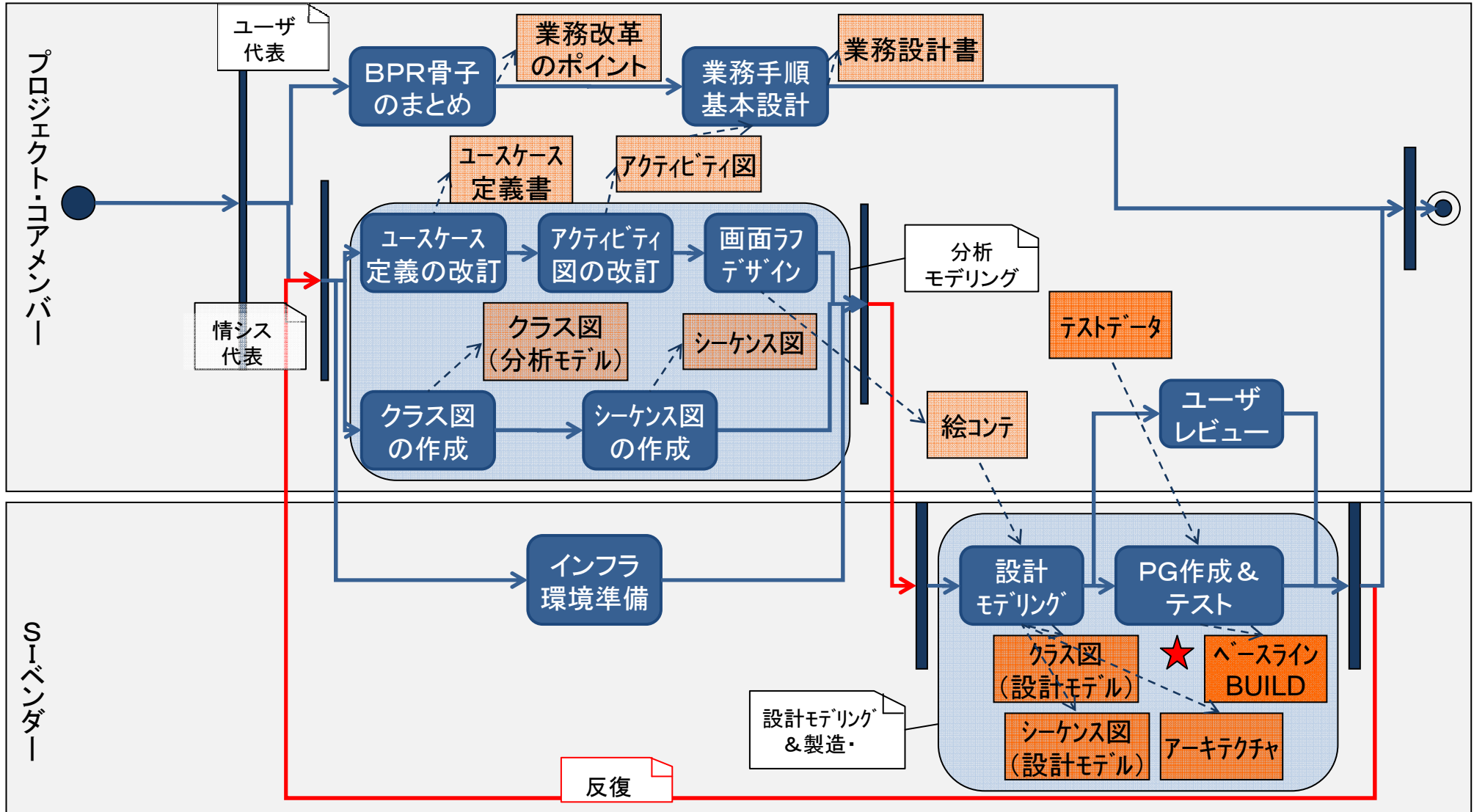


この部分はSIベンダー作成のWBSには殆ど示されることはないが、ビジネス系大規模アプリでは全工数の約半分を占める

推敲フェーズの手順(パッケージ)



推敲フェーズの手順(スクラッチ)



※推敲フェーズのマイルストーン

● システム分析

＜パッケージ適用の場合＞

- 分析モデル; PKGモデルへの変換ロジック&ADD-ONエンティティ、ユースケース基本フローレベルのADD-ONプロセスが固まる
- 画面設計; 主要なアドオン画面・入出力データ項目が固まる
- 実装; PKG標準部分がテスト駆動する

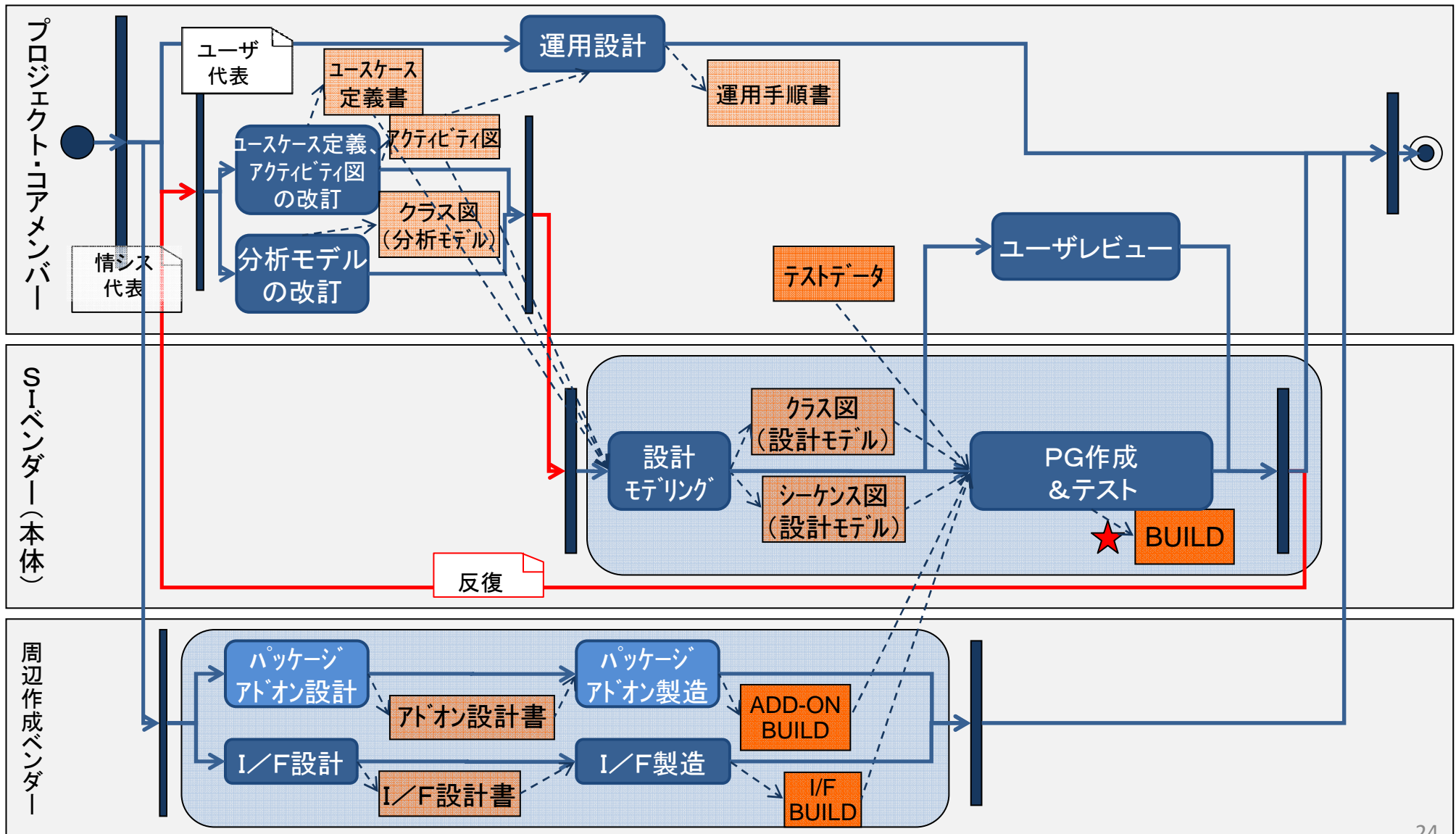
＜スクラッチの場合＞

- 分析モデル; 主要クラスを捉えた概念モデル(クラス・シーケンス図)が固まる
- 画面設計; 主要画面の主要入出力データ項目が固まる(デコレーションなし)
- 実装; 主要ユースケースのベースライン・アーキテクチャが固まり、テスト駆動する

● 業務設計

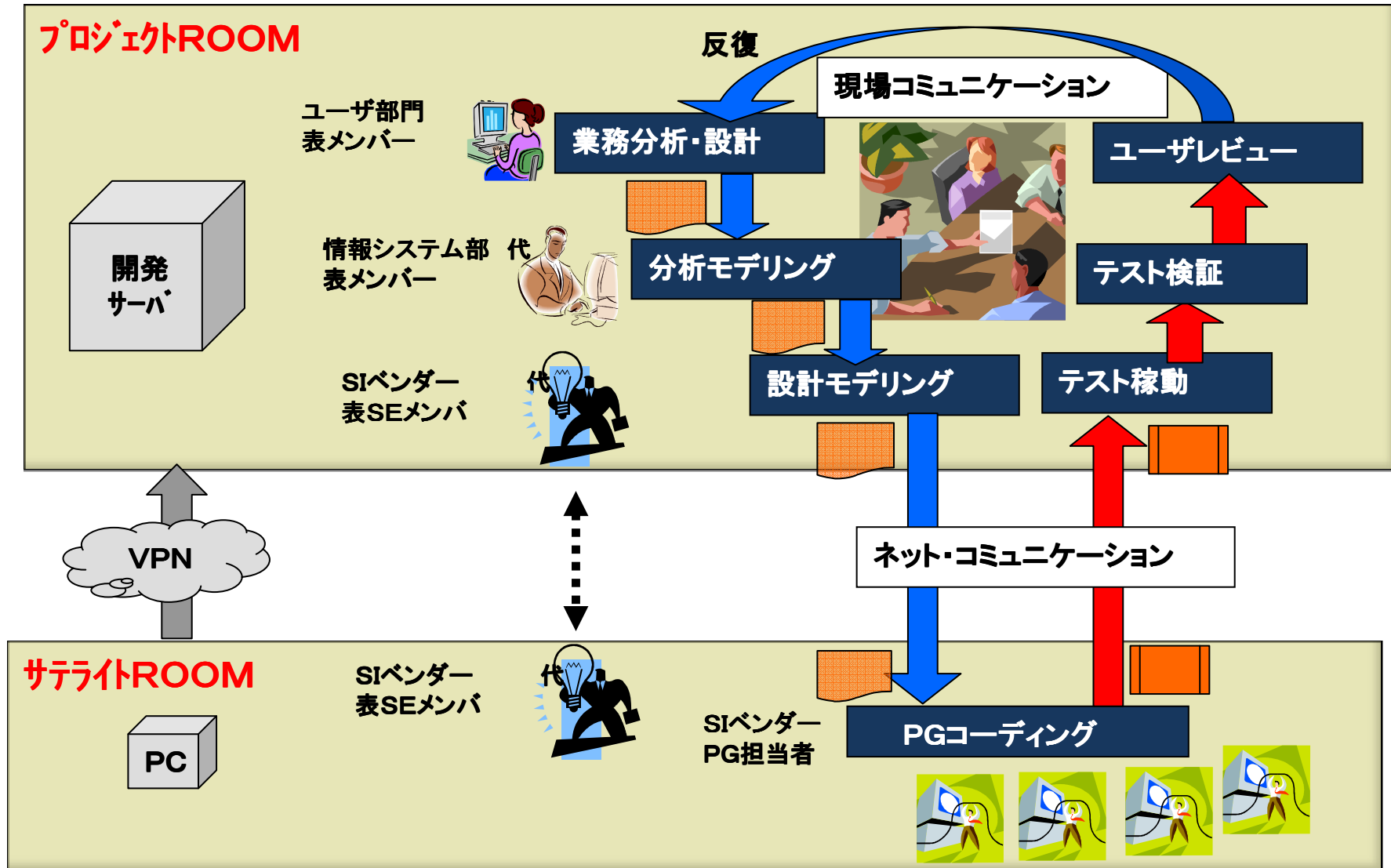
- 業務・組織改革の“ゆるぎない骨子”が固まる
- 業務フロー概要がアクティビティ図のレベルで固まる(アクター/実組織の対応表も)
- 画面モックアップを通して業務のボリューム、実現性を確認する

作成フェーズの手順



※SIベンダーとのコミュニケーション

KYOWA KIRIN



<SIベンダーへの期待>

ウォーターフォールの特徴と課題

<特徴>

- 前行程で、**次工程のゆるぎない仕様**が確定している
 - 製造工程の一括アウトソース(オフショア含む)が可能となる
 - セオリーが確立している業務システム(制度会計など)に適する
- 通常、**請負型契約**の下で適用し、ベンダー・ユーザ企業双方のリスクをヘッジ
 - ベンダー側は、前行程への揺り戻し分などのリスクを予め見積りに反映
 - ユーザ企業側は、万が一のベンダーのスキル不足でも追加費用はなし

<課題>

- ドキュメントやモックだけでは仕様のレビューに限界がある
- **仕様変更**(と言われるもの)が開発後半に頻発することが多い
 - 開発序盤で無理に仮置きした詳細仕様は“変更”の発生が必然?
- 終盤の**実装工程**で**致命的問題**が発生する可能性がある
 - 難易度の高いロジックや非機能要件の検証に十分な時間が取れない
- 近年、リスク管理により見積りが安全サイドに流れ、**過剰コスト**の傾向あり
 - 現に見積りの段階でリスクファクター(1. x)を乗じていると聞く

反復型の必然性とリスクへの対応

KYOWA KIRIN

<反復型の必然性>

- ウォータフォールの一歩の狙いは“プロジェクトを失敗しない事”にあり
 - 仕様変更による納期超過を防ぐ事や、請負契約で際限ない予算超過を防ぐ等
- 序盤で仕様を凍結するのではなく、臨機応変な変更を可能にしたい
 - 納期 & 予算通りである事も重要だが、“良いシステムである事”がもっと大事！
- 実際に動くソフト(基本フローのみ)を用いて、ユーザ・レビューが可能
- 難易度の高いロジックや非機能要件の早期着手によりリスクを最小化
- 早い段階からテストを繰り返す事で、十分なシステム品質を作り込む

<リスクへの対応>

- エンドレスなスパイラル開発に陥るリスク
 - 方向付け、推敲、作成、移行の4フェーズを設け、予め各々の期間を設定
- 各フェーズのマイルストーンが不鮮明となるリスク
 - フェーズ毎の開発ドキュメントやモデル図と言った成果物を明確化
- SIベンダーの開発費用が不明確になるリスク
 - 全てを準委任契約とする事で、機能追加時は人員増(費用増)で対応

SIベンダーへのお願いと期待

KYOWA KIRIN

<お願い>

- 自社のウォーターフォール型開発方法論に固執せず、反復型などのユーザ企業が望む開発方法論にも対応して欲しい
- 大規模ビジネス・アプリ開発にも準委任契約を適用して欲しい
- UMLをはじめとする各種モデル表記が理解できていて欲しい
- 売買の関係にあっても目的を共にするパートナーと思って欲しい

<期待>

- 物作りが海外へ出てゆく中、国内での高密度なコラボに期待します！
- クラウド時代に突入し、各種アプリサービスの充実に期待します！
- ERPにはない企業競争力を支援する独自サービスに期待します！
- ITの世界で日本発グローバルスタンダードが出ることを期待します！

KYOWA KIRIN