

探求者(問題フレーム活用)

価値ある問題をデザインしよう

Designing a Valuable Problem guided by Problem Frames 2009.6.18~19

はじめに

【探求者がやりたいこと】

大槻のマインドマップ

ソフトウェア経済学

価値と解のマトリクス

開発プロセス

見積り手法の例

ソフト請負開発会社の見積り手法

アジャイルプロセスの見積り手法

見積り手法の俯瞰

ドメインとインタフェース(共有現象)

予測問題: コンテキスト

アジャイルプロセスとドメイン

予測とは?

サービスとしてのアジャイルプロセス

正規分布、指数分布

モデルドメインと現実世界

モデルドメインの導入

顧客提案・交渉の局面

【探求者としての課題】

モデリングはモデルを作ることではない

おわりに

S. Otsuki



はじめに

宣教者(飯泉)の『問題は「問題」にある』の話を受けて、私は探求者として「問題フレーム」を活用して、私が取組んでいる「問題」の話をさせていただきます。

問題フレームは、形式手法でも、手順的な手法でもありません。まして、モデリング手法でもありません(たぶん)。

問題フレームは、ものの見方、考え方、認識の方法、分析の観点を提供しています。

私が取組んでいる問題は、強いていえば「プロジェクト見積り支援システム」に関する事項です。従来のウォーターフォールプロセスで仕様が固まってからの請負受発注時の見積り手法は、そこそこのものがありますが、インクリメンタルやアジャイルプロセスでは未だよい方法が確立されていません。そこで、この新しい見積り手法の問題を、問題フレームを活用して分析してみようと思います。

【探求者がやりたいこと】

ソフトウェア開発で、見積り予測をする機械（マシン＝コンピュータおよびその上で動くソフトウェア）を構築したい。

顧客要求の不確実性に対応したい。

開発リソースを決め、マネジメントに活かしたい。

顧客への提案、顧客との交渉に活かしたい。

優秀なマネージャ、リーダー、営業担当者等がやっていることを定式化し、機械に組み込みたい。

経験やデータの活用

ステークホルダ調整や説明論拠、ロジック

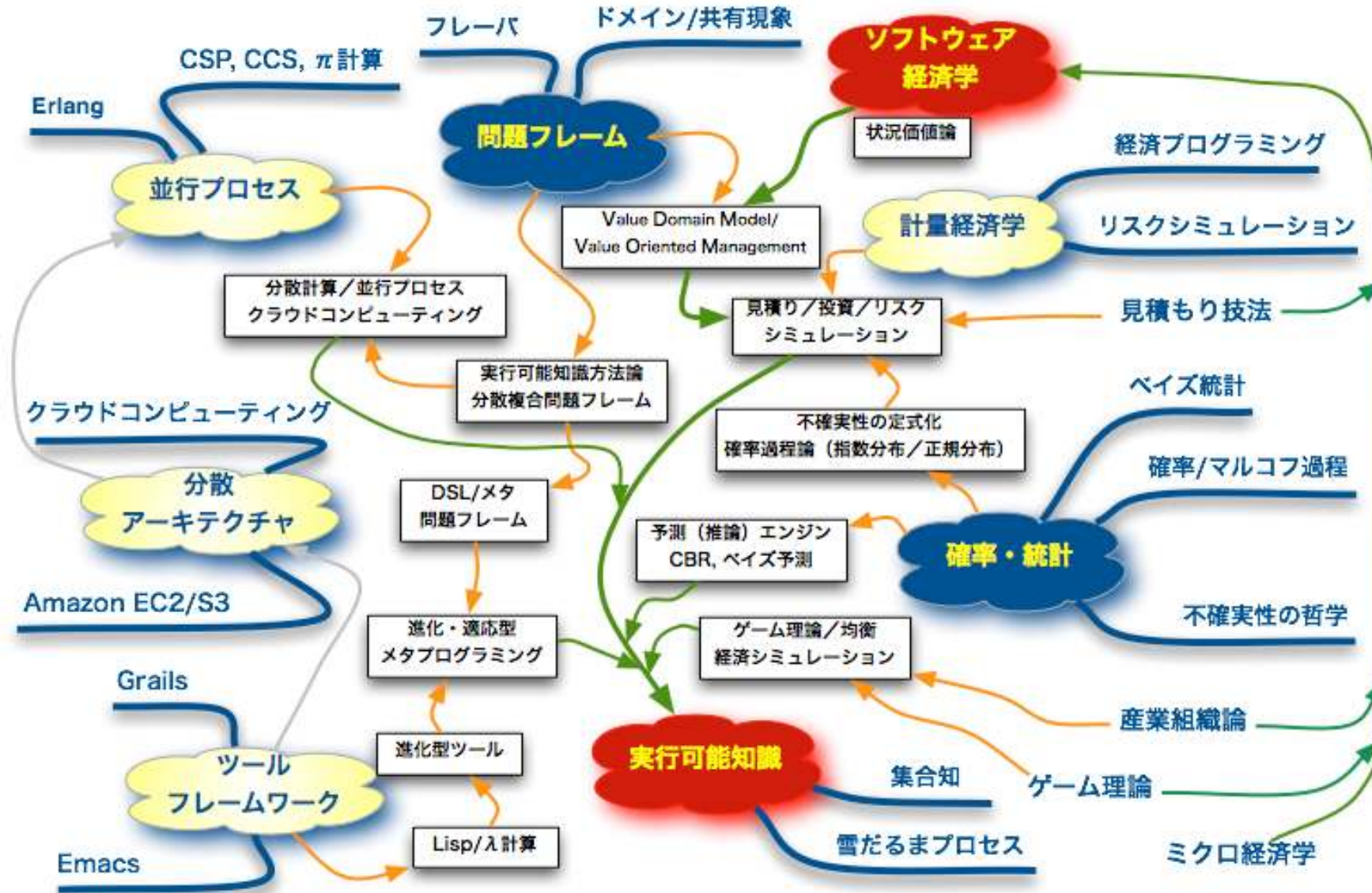
推論、見積り手法、メトリクスの活用

予測問題とは？

過去の経験やデータに基づき、
（将来の）不確実性を考慮し、
将来の事象（起こり得る事柄）を、
科学的・客観的に推論すること。

見積り予測を含む「ソフトウェア経済学」の知見を、 実行可能知識化(活きたソフトウェアの構築と進化)したい

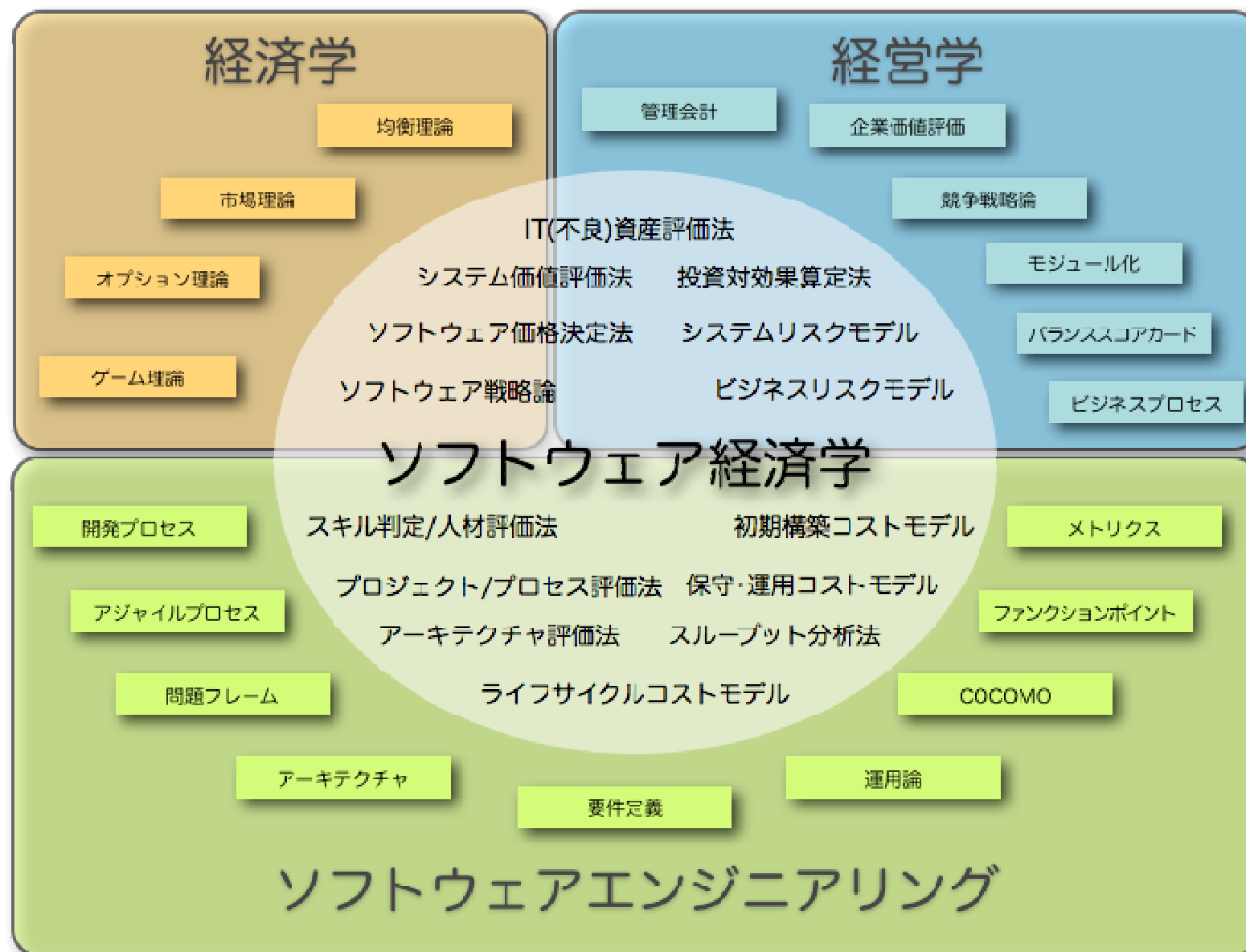
大槻の頭の中(文字通りのマインドマップ)



ソフトウェア経済学で明らかにしたいこと

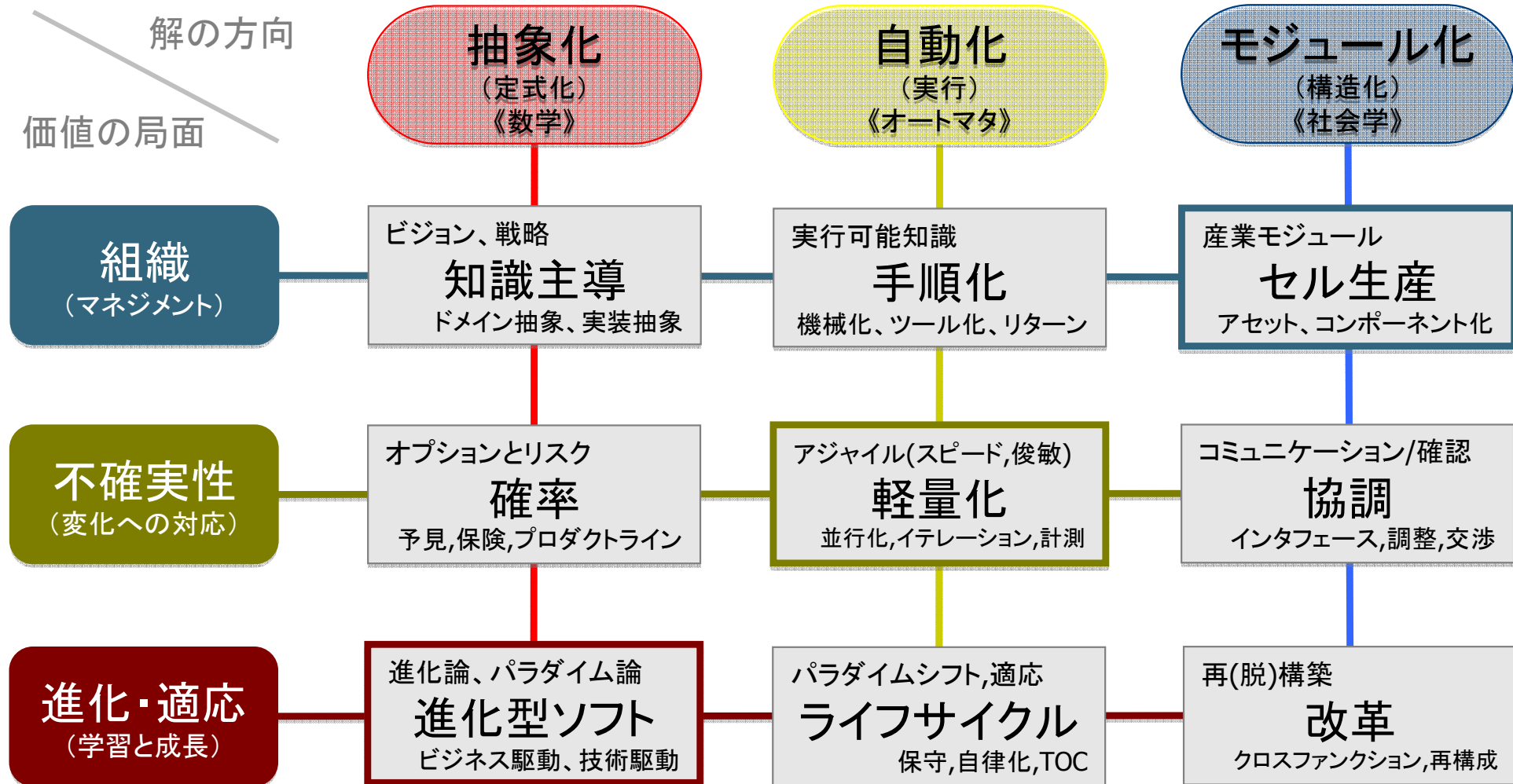
ソフトウェア
経済学

- ❖ ソフトウェア、システム、サービス等の無形財の利用、開発、保守、運用、破棄の総合的な社会/経済的な振舞い
- ❖ 市場、組織、部門、プロジェクト、チーム、個人の一貫した社会/経済的な振舞い
- ❖ 価値、価格、費用(コスト)の定式化と、これ間の関係



ソフトウェアが生み出す価値と それを生み出す方法(解)との関係

価値と解のマトリクス



いろいろな開発プロセス (アジャイルとか・・・)

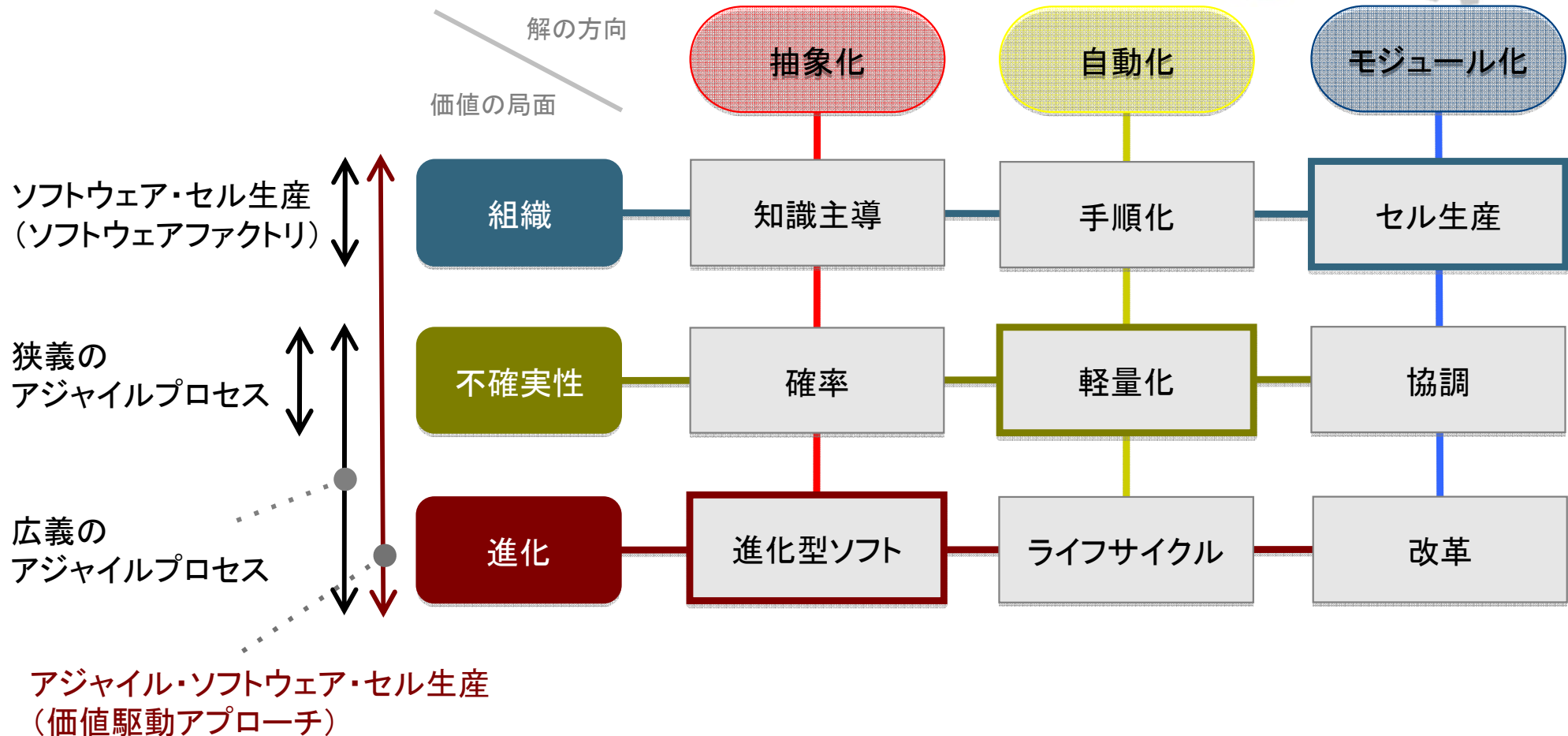
❖ 狭義のアジャイルプロセス

- 開発者側中心の視点
- 顧客側とのコミュニケーションや確認を重視してはいるものの、あくまでも受動的



❖ 広義のアジャイルプロセス

- 顧客側と開発側との同期
- 全体での最適化、ビジネスプロセスを能動的に考慮



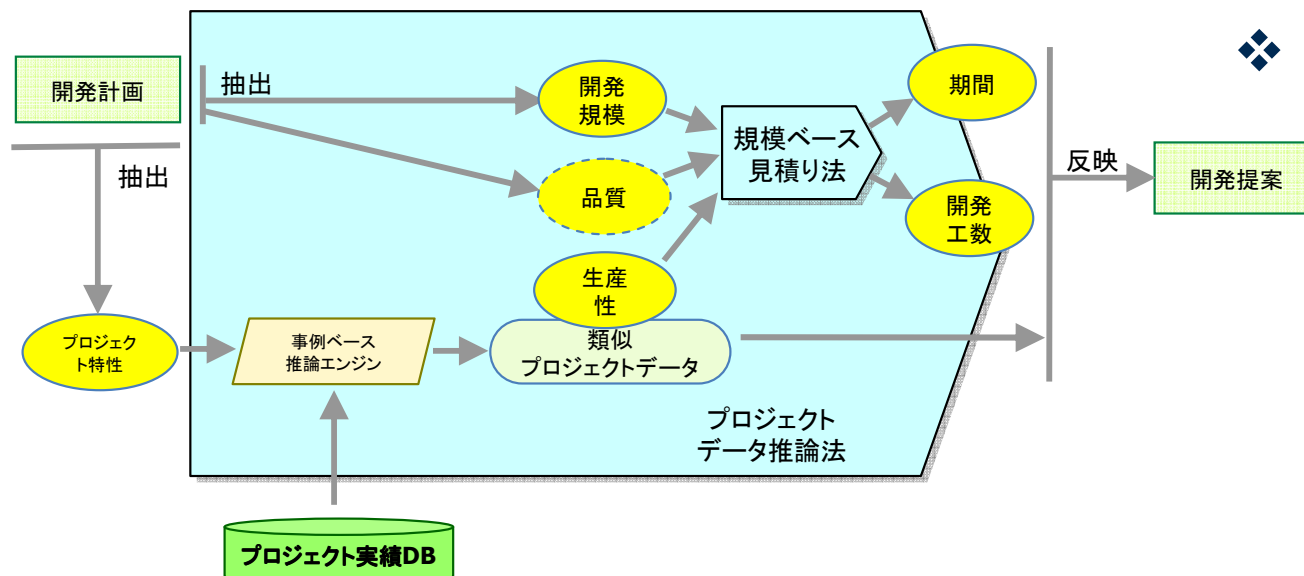
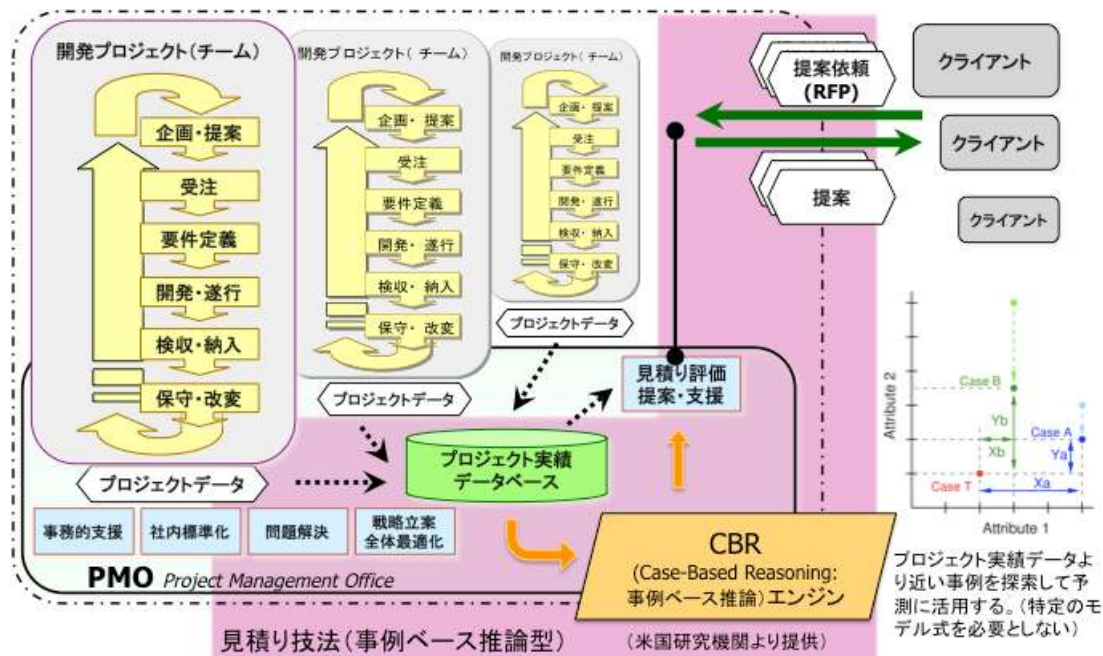
❖ 一般的な見積り技法の例

対象	分野	見積り技法/カテゴリ	概要
A社	プログラムプロダクト開発	エンハンス調整法 /マーケット主導型	開発項目要求に関する明確なステークホルダ間合意を優先順位付で行い、前回の開発における詳細WBSをベースに、経験と類推により、精緻な要求/実装対応の詳細WBSを作成する。開発期間優先で、リソースとの兼ね合いから、そのバージョンでの開発項目を決定する。
B社	アプリケーションソフト請負開発	プロジェクトデータ推論法 /事例ベース推論型	過去のプロジェクトデータから事例ベース推論エンジンを使って、類似しているプロジェクトを抽出する。類似プロジェクトの生産性から提案プロジェクトの工数と期間とを算出する。
C社	インフラ関連運用	効用ベース価格算定法 /従量価格算定型	提供ソリューションごとにトランザクション量、サーバホップ数、冗長化ルート数等からクライアントのインフラ投資に見合った従量型の価格設定を行う。インターネット計測統計のフィードバックも行う。
D社	Webシステム開発	タイムボックス法 /アジャイルプロセス型	提供フィーチャとプロセスとを統合したメタコンポーネントによってクライアントと開発範囲と開発期間を合意し、それらに基づいて、アジャイル・ソフトウェア・セル生産方式のセル月を見積もり、セル構成を決定する。並行セル数が多いほど、セル月単価は高く設定する。
E社	システム保守・運用	非機能要求調整法 /プロセス積上げ型	SLA(OLA)設定から導出された、複数組織にまたがる保守・運用プロセスから組織ごとに工数積上げを行い、その基準値をベースに運用特性によるコストドライバによって調整して見積もる。
F社	セキュリティ診断	標準診断差分法 /ベストプラクティス差分型	各種セキュリティ標準に従ったチェックリスト診断を行い、ベストプラクティスとの差異から改善項目を洗い出し、項目積算によって見積もる。
G社	組込みソフト開発	TOC改善調整法 /アセットベース型	システム開発の対象となる構成ドメインのアセット(モデル+プログラム)利用による部分(非規模ベース見積り)と、新たに改善(ボトルネック解析)する部分とから積上げ算出する。
H社	流通・サービス系ソフト開発	セル生産算定法 /プロダクトライン型	共通部、支援環境等の開発を高スキルの専門集団が行い、個別アプリケーション開発を定形・手順化して一定の工数見積り範囲に押さえる。派生製品数と期間から生産性向上の目標とその投入工数を算出する。

アプリケーションソフト請負開発会社の見積り手法

❖ 請負開発

- 開発提案
- 受発注価格交渉
- PMO等による組織的プロジェクトデータ収集



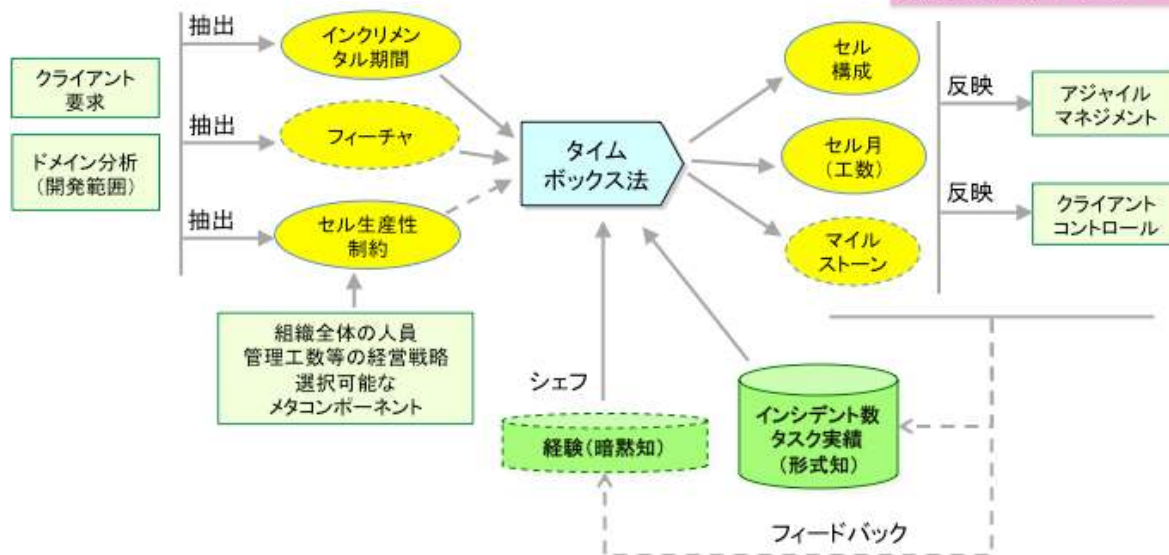
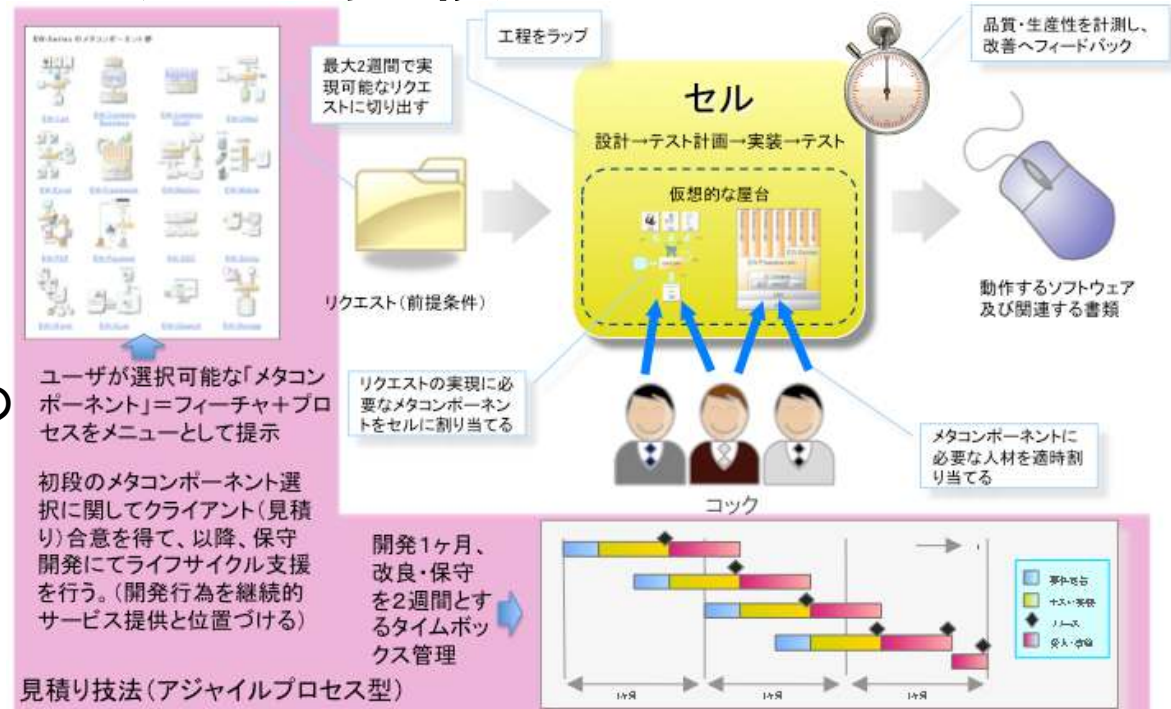
❖ 非規模ベースの特徴

- 事例ベース推論エンジン
 - モデル(計算式)不要
- 類似プロジェクトの抽出
- 類似性の定義による推論方式の改善
- 規模ベース見積りとの併用

Webサービスシステム開発会社の アジャイルプロセス(セル生産)の見積り手法

❖ アジャイルプロセス

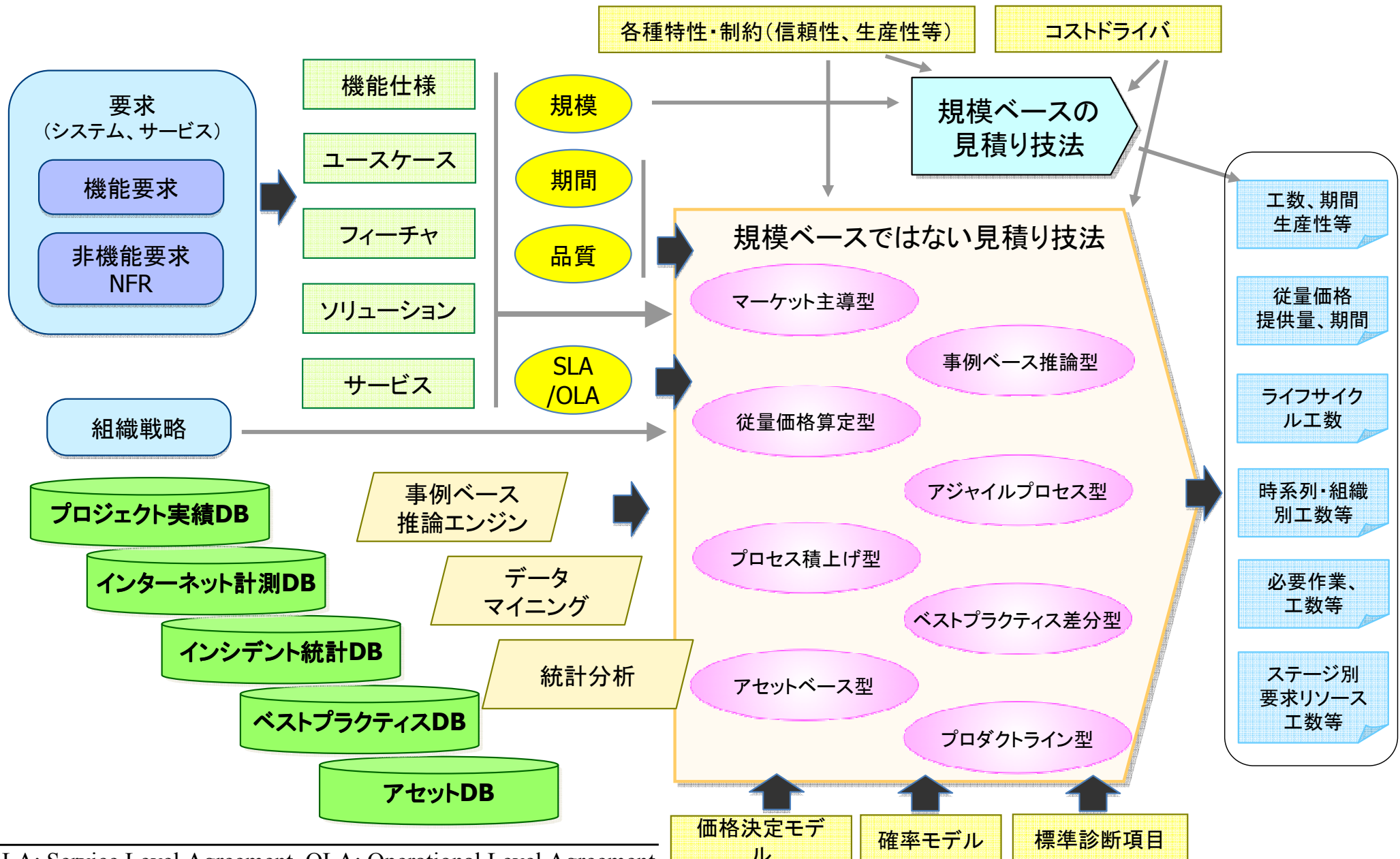
- Webシステム開発
- セル生産方式
 - 人月からセル月へ
 - セル = 工程をラップしたもの
 - 開発プロセスのパターン化
- 保守型契約に類似



❖ 非規模ベースの特徴

- セルの見積り
- 時間を区切って(タイムボックス化)、セルを割り付ける
- 顧客要求の時系列予測
- 不確実性に対応するバッファ(ゆとり)計算

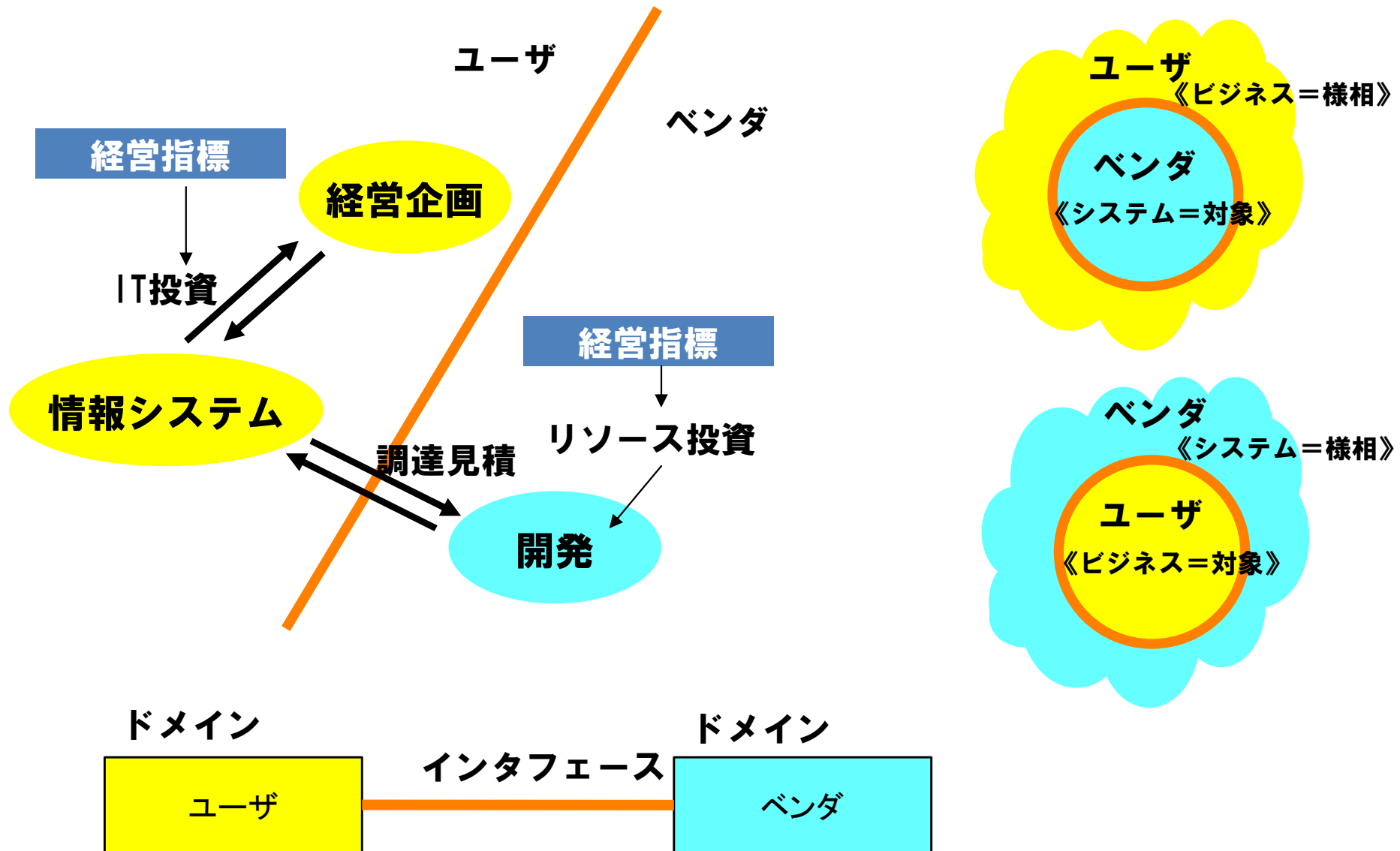
見積り技法の俯瞰



SLA: Service Level Agreement, OLA: Operational Level Agreement
NFR: Non-Functional Requirements

ドメインとインタフェース(共有現象)

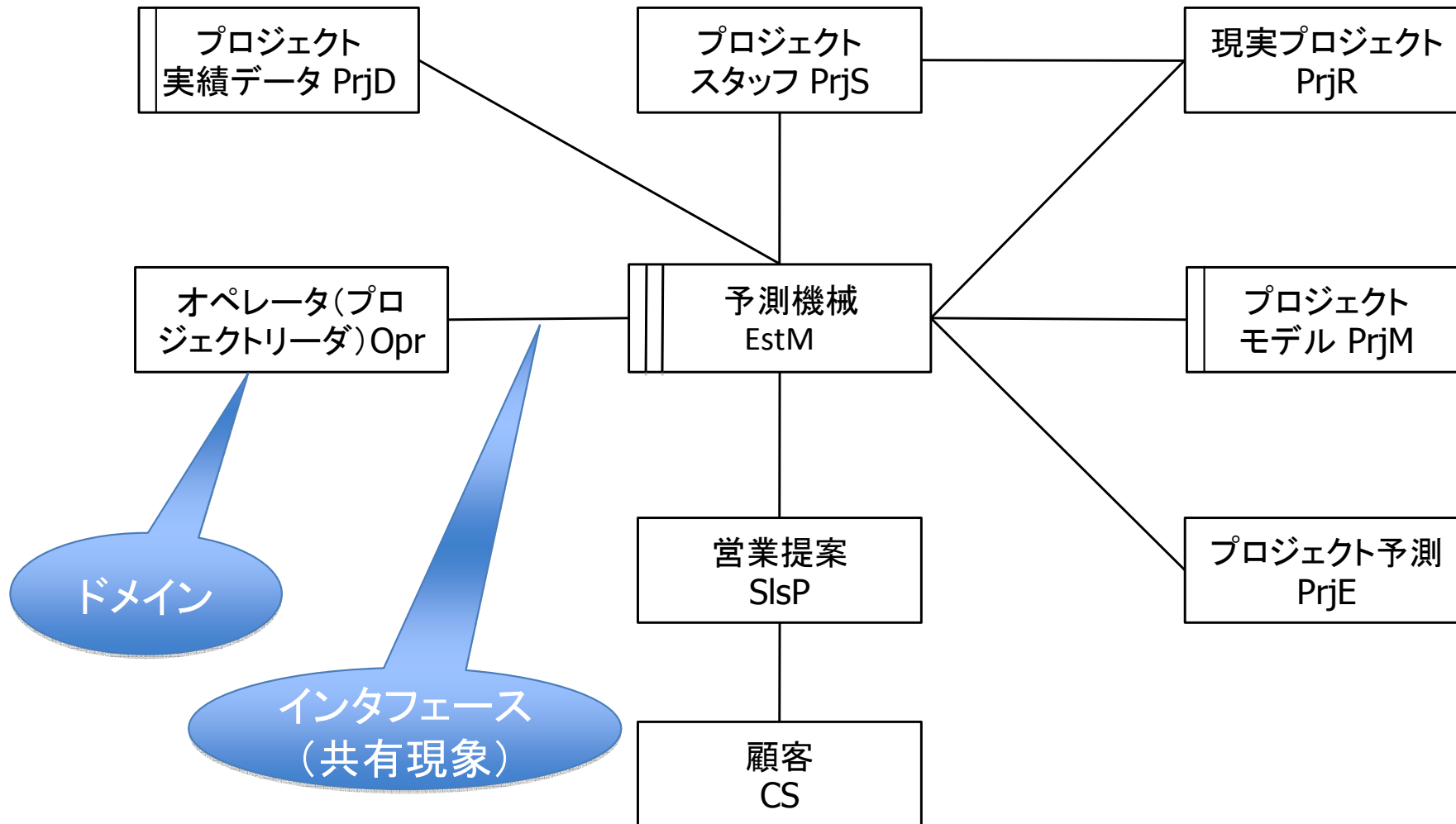
[宣教者スライドより]



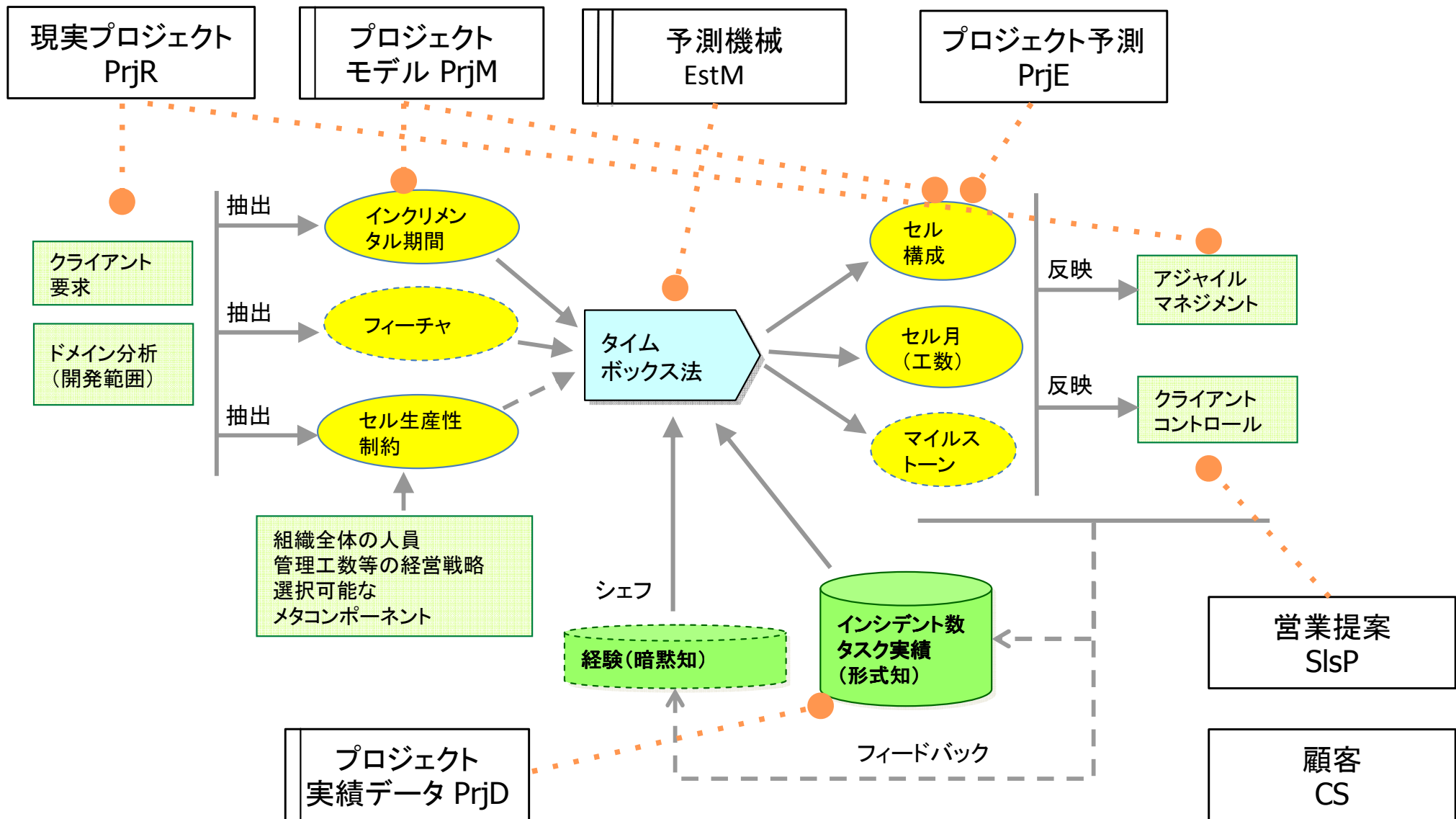
予測問題:コンテキスト

実行可能知識

問題フレーム／コンテキスト図



予測問題：アジャイルプロセス（セル生産）の場合

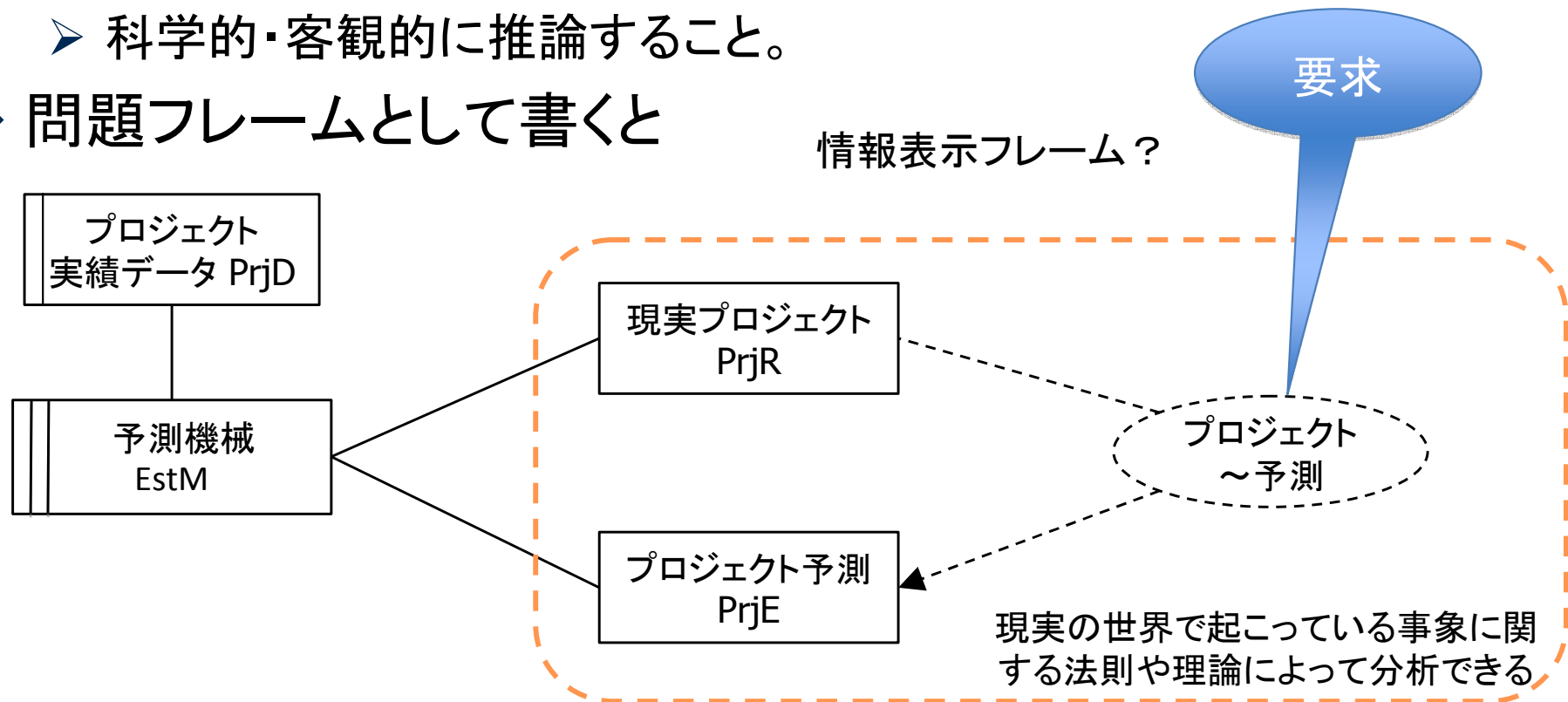


予測問題：予測するってどういうことよ？

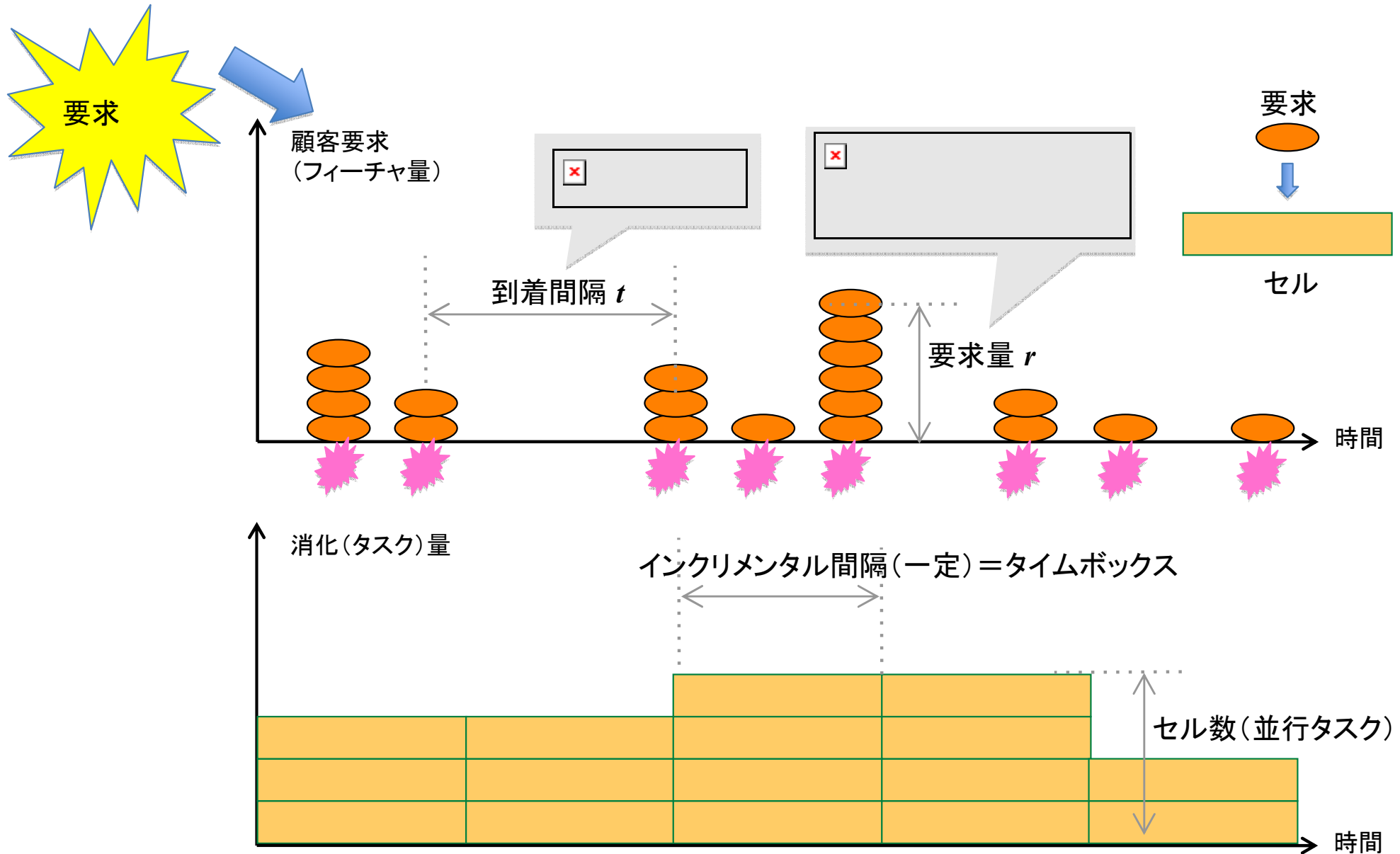
❖ 予測とは？

- 過去の経験やデータに基づき、
- (将来の)不確実性を考慮し、
- 将来の事象(起こり得る事柄)を、
- 科学的・客観的に推論すること。

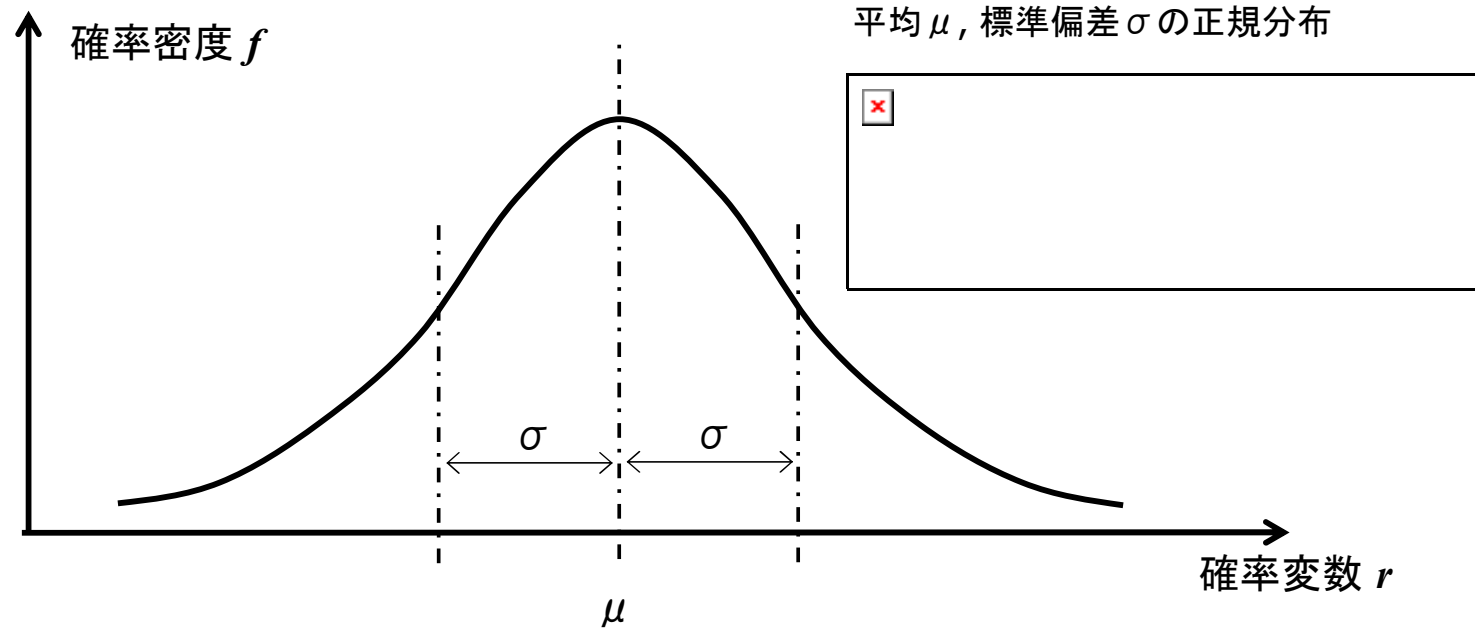
❖ 問題フレームとして書くと



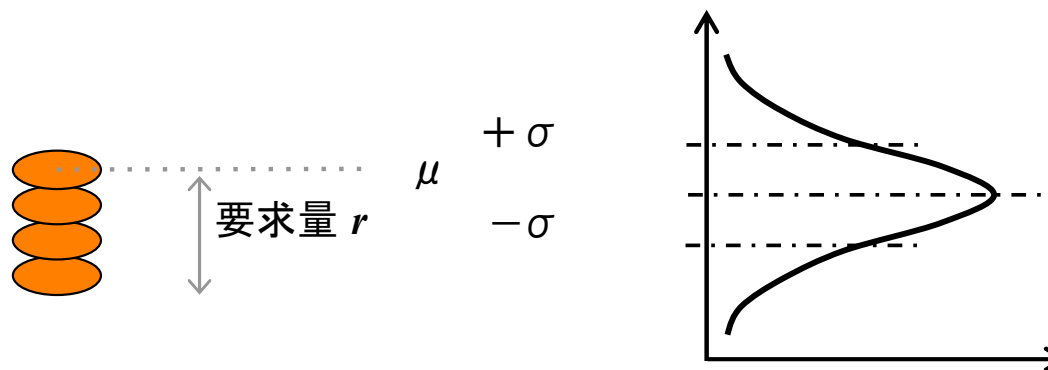
予測問題：サービスとしてのアジャイルプロセス



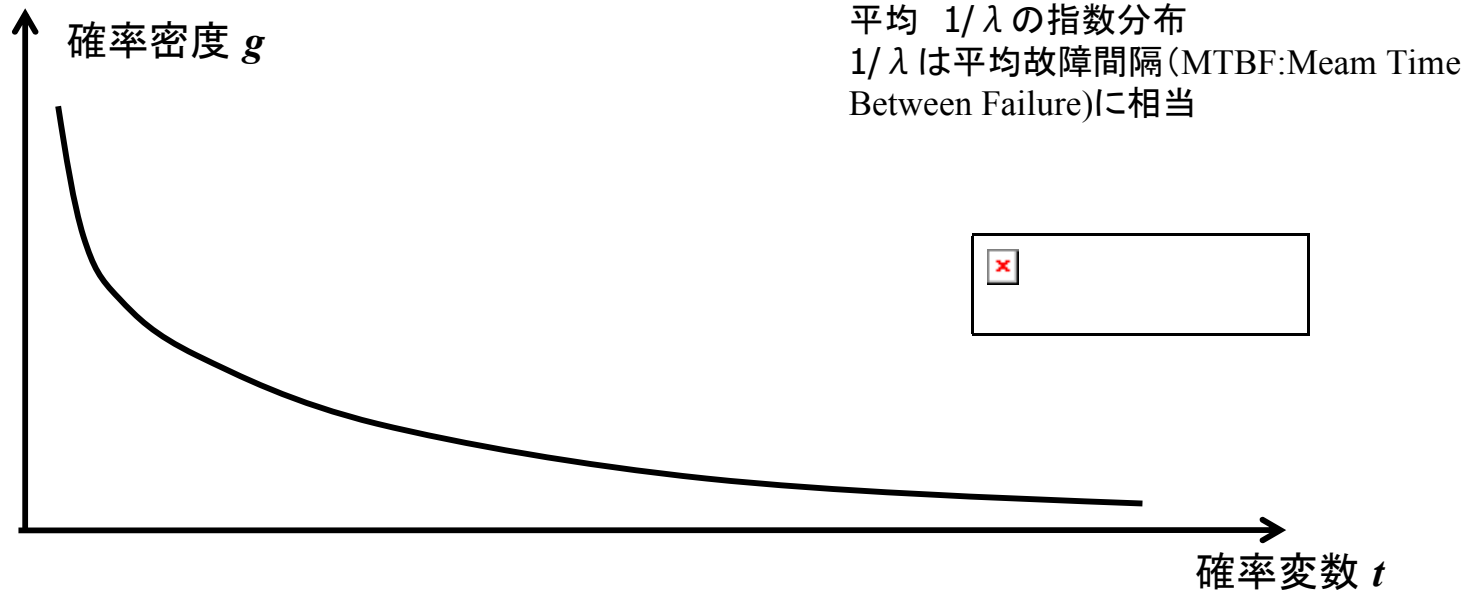
予測問題：不確実性（正規分布）



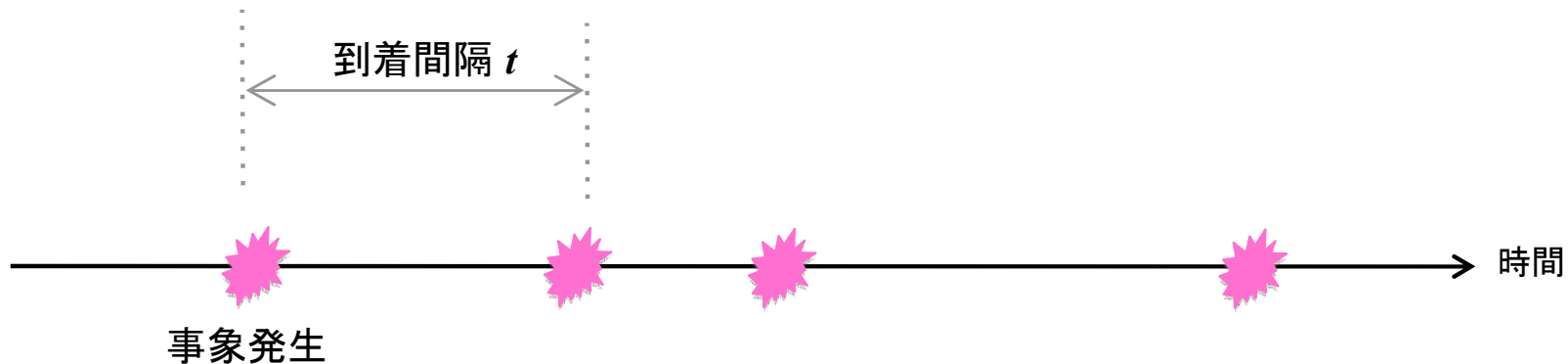
正規分布は、最も素朴な不確実性の表現
要求がどれくらい発生するかは不確実なので、これを確率で捉える



予測問題：確率過程（指数分布）

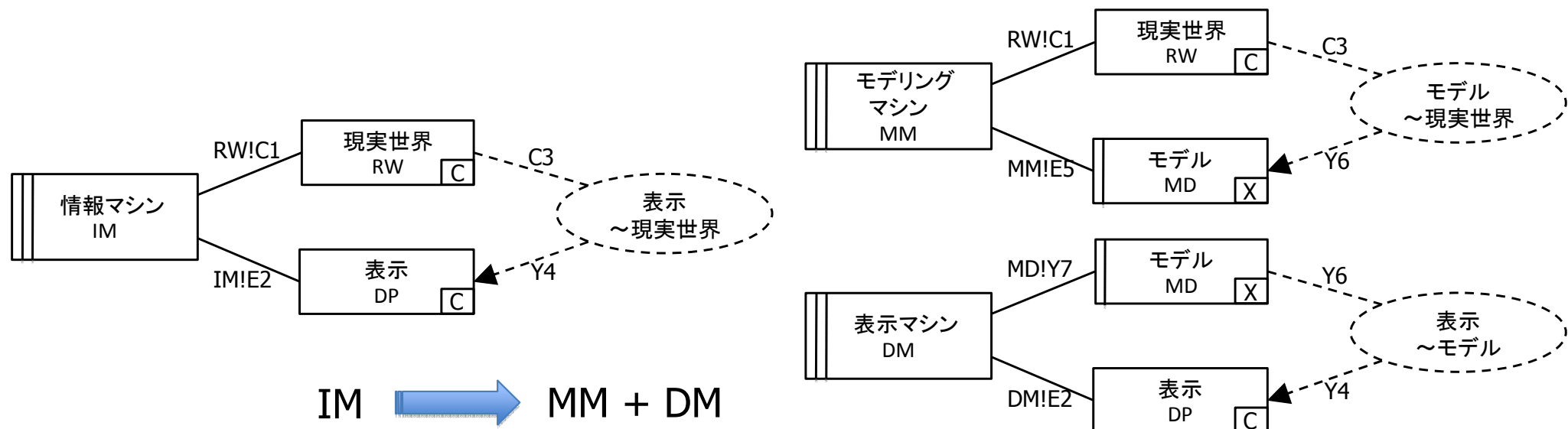


指数分布は、最も素朴な不確実な事象発生表現
要求がいつ発生するかは不確実なので、これを確率過程で捉える



モデルドメインと現実世界

- ❖ モデル: 類似モデル
情報問題の現実世界ドメインのこと
- ❖ モデルドメインは、情報機械内部の非共有現象の一部、すなわち、表示出力の計算に使用する変数のセットを分離して明確にします。
- ❖ モデルドメインを使用する際には、問題は2つの副問題: モデル構築部分と、モデル利用部分とに分けられます。
- ❖ モデルドメインの役割は、過去を要約すること、現実世界の推論を裏付けること、非常に長い計算の結果を予想すること等があります。
- ❖ モデルドメインは、現実世界とは乖離する可能性もあります。

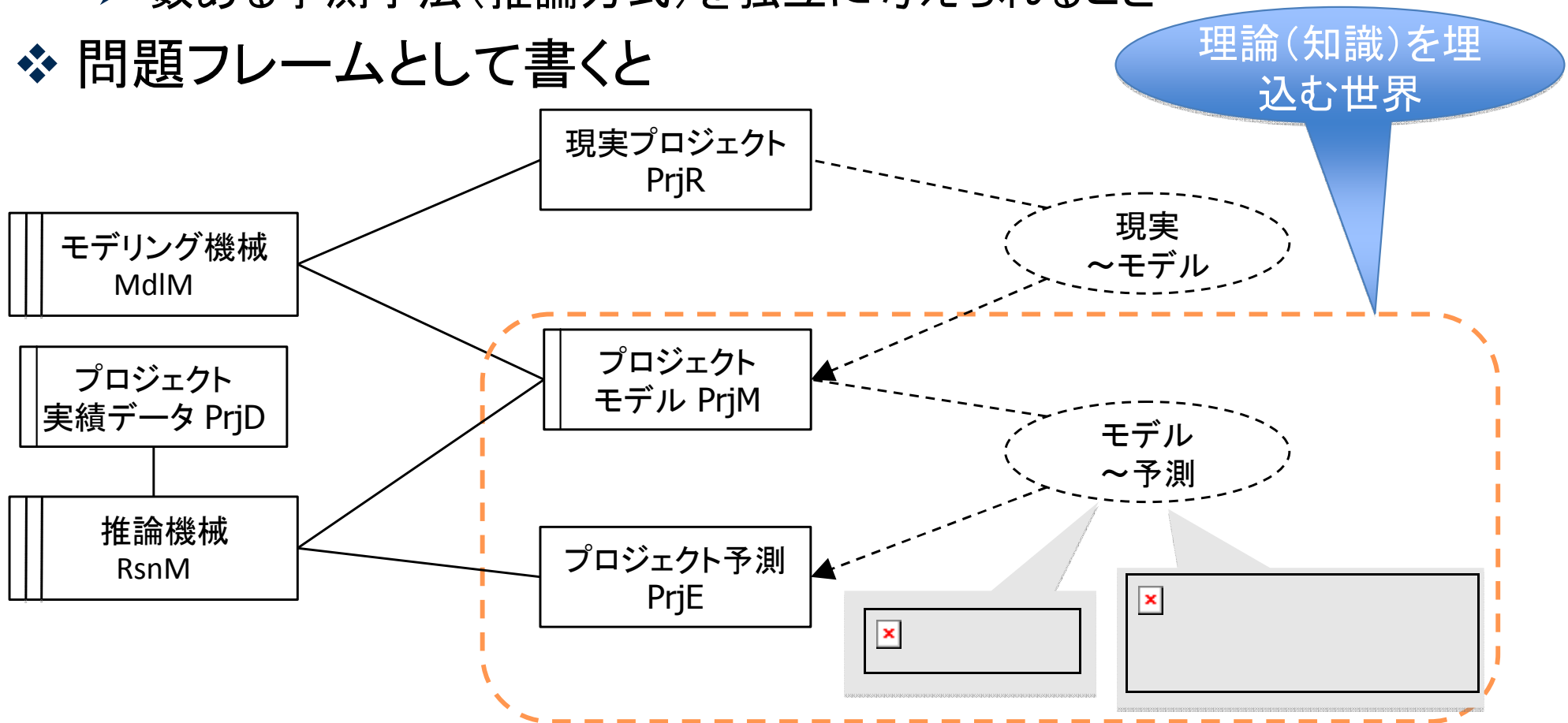


予測問題：モデルドメインの導入

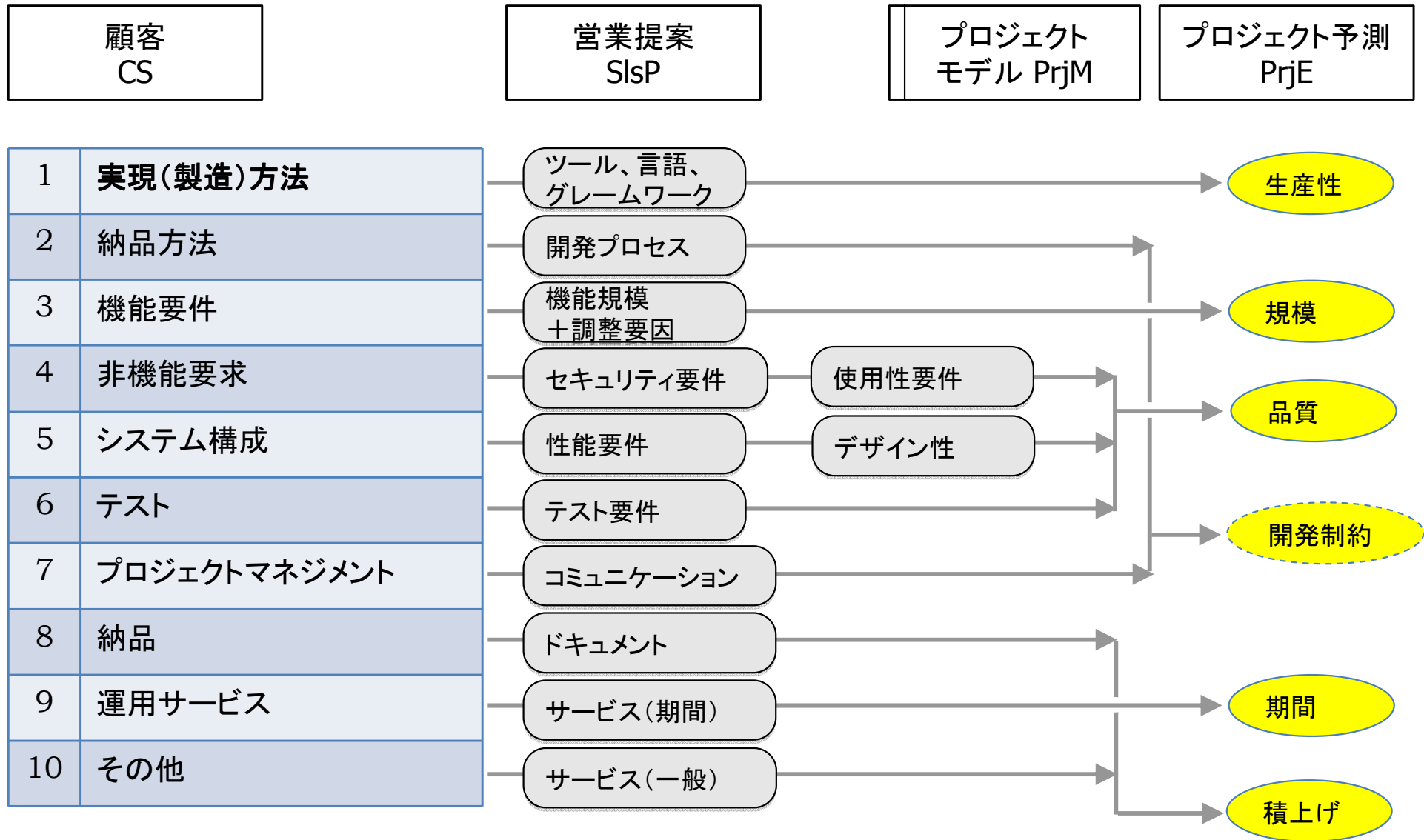
❖ モデルドメインの利点

- プロジェクトを抽象的(予測に必要な観点のみ)にとらえることができること
- 数ある予測手法(推論方式)を独立に考えられること

❖ 問題フレームとして書くと



予測問題：顧客提案・交渉の局面



【探求者としての課題】

課題1: 理論組込み

実世界にある問題を解くソフトウェアを、問題を解く理論を適用して実現したい。

ビジネスや組織活動では、問題を解き続ける必要がある。

蓄積された理論、ノウハウ、定式化、定理・法則等を組込んでいく方法がほしい。

〔問題フレームは理論領域を含む「概念」については、手法を示してはいない。〕

課題2: 複合世界

ソフトウェアに関わる複数の世界(私的世界／主観世界)を扱いたい。

世界は、独立した領域から成り立っている。

それぞれの領域から他の領域は観測できない(情報の非対称性)。

〔問題フレームの「ドメイン」の考え方は、主題と文脈を整理するのに好適〕

課題3: 進化・適応

状況に適応しライフサイクルの長いソフトウェアを実現したい。

機械は、実世界に投入することによって、実世界に影響を及ぼし、認識を変えてしまう。

進化そのものを予測しデザインすることは難しい。「第三の目」があるとよいかも。

〔メタ問題フレーム(問題そのものの発生のパターン?)が提唱できるとよい。〕

モデリングはモデルを作ることではない

〔自由な闘い（宴会）用〕

命題1: ソフトウェアエンジニアリングの定義

ソフトウェアエンジニアリングとは、
ソフトウェアの構築・保守・運用等に関わる《記述》に関する知識体系である

〔《記述》できない(語り得ない)世界も存在する。たぶん〕

〔記述のみならず、行為(言語行為)、記号(しぐさとかモード)まで入れるべきか迷うところ。〕

命題2: モデルの定義

モデルとは、類似の特性を持つと信じられる別の実体(類推モデル)のことである

〔分析モデルは、単に「記述」あるいは「分析的記述」と呼ぶ。〕

命題3: モデリングの定義

モデリングとは、言語によって記述すること

〔言語には、自然言語と規約言語とがあり、規約言語のうち数学的に厳密に意味が定義されている言語を「形式言語」と呼ぶ。プログラミング言語は形式言語の一種である。〕

命題4: モデリングの目的

モデリングの目的には、機械(汎用計算機)への命令、分析(定式化)、および、
伝達(合意形成)がある

〔伝達(コミュニケーション)による《合意形成》とは、《両立可能》であることを示し、意味が伝わることではない〕

おわりに

「問題フレーム」の考え方は、得てして陥りがちなエンジニアリング領域にはまり込んでしまうことを避けることができ、本来の「問題」を考えることに導いてくれます。

価値のないソフトウェア、既に解けている問題を解いても何の意味もありません。人間の叡智、知的資産をソフトウェアという機械に組込み、進化させていくことができれば、この業界ももう少々元気になっていくものと期待しています。

