

レガシーシステムの現在と未来

2015年10月16日
株式会社ワンサークル
代表取締役 星野 亨



会社の紹介

レガシーシステムに関する
課題解決の専門会社です

大手システムインテグレータでの
経験を通じて、レガシーシステム
ユーザー様への技術コンサルタント
として2015年2月に設立しました。

主な事業内容

- ・技術顧問サービス
- ・技術支援サービス
- ・情報発信活動

アジェンダ

1. レガシーシステムとは
2. 「いま」やるべきこと
3. 「これから」やるべきこと
4. まとめ

TAKE

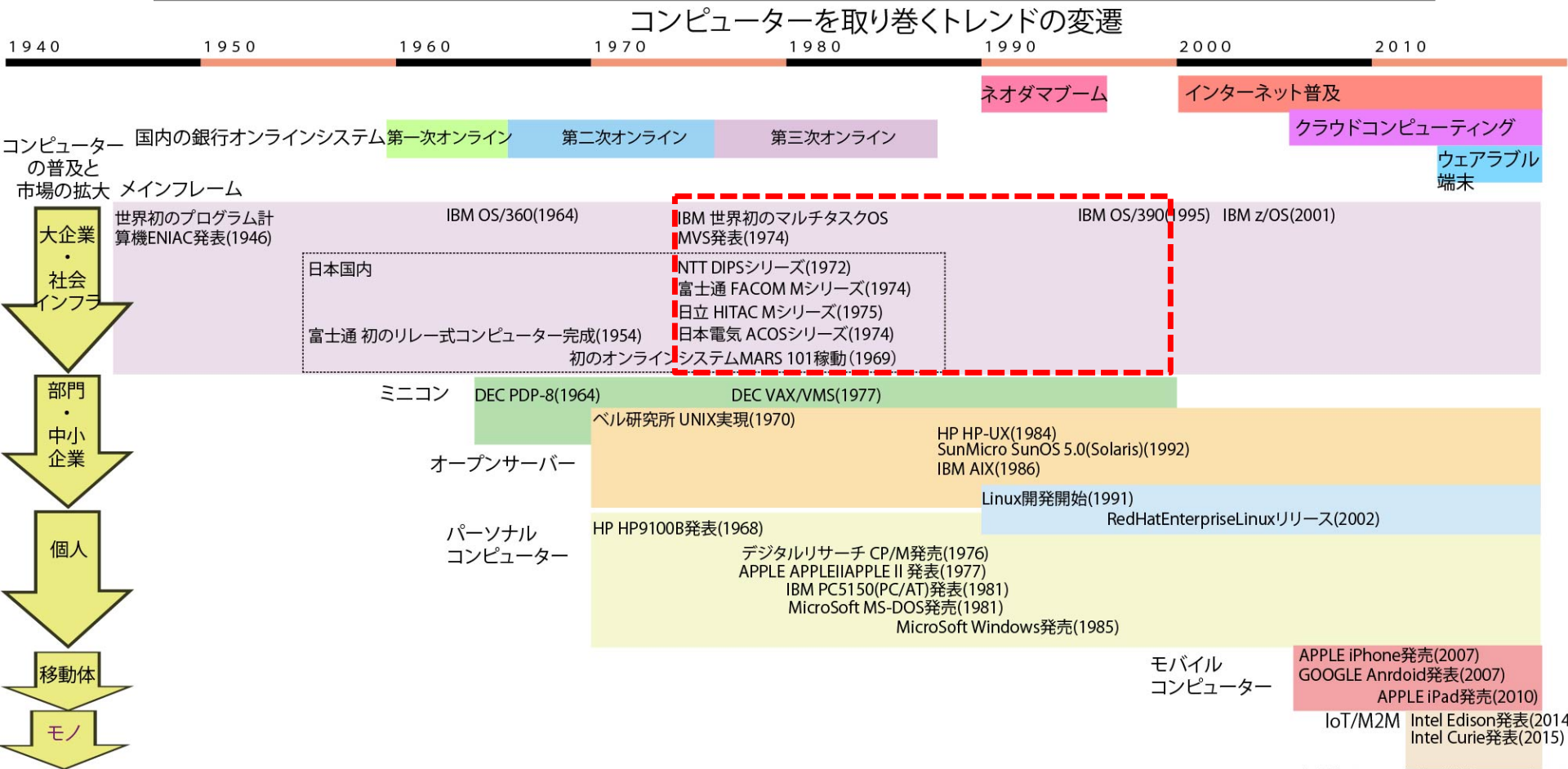
ACTION



1. レガシーシステムとは



レガシーシステムの歴史



2015.2 一般情報を元に当社が独自にとりまとめ

40年前頃から電算化が広まる(MVSの登場、国産メインフレームの普及)
国内景気回復～バブル景気で一気に電算化が加速 → レガシーシステム化

レガシーシステムのキーワード

「コンピュータによる手作業事務の再現」

おもてなしの心：

紙ベースの事務をコンピュータで忠実に再現
コンピュータで世の中を便利に(VAN、C&C)

ハード効率重視プログラミング：

当時の非力なコンピュータ機器のフル活用
1ビット・1ステップも無駄にしない設計

アセンブラ、マルチレイアウト、バイナリの多用
ヒト(人月)よりもモノ(計算機)の値段が高かった時代

レガシーシステムのオープン化が難しい要因

レガシーシステムのキーワード

「基幹業務システム」

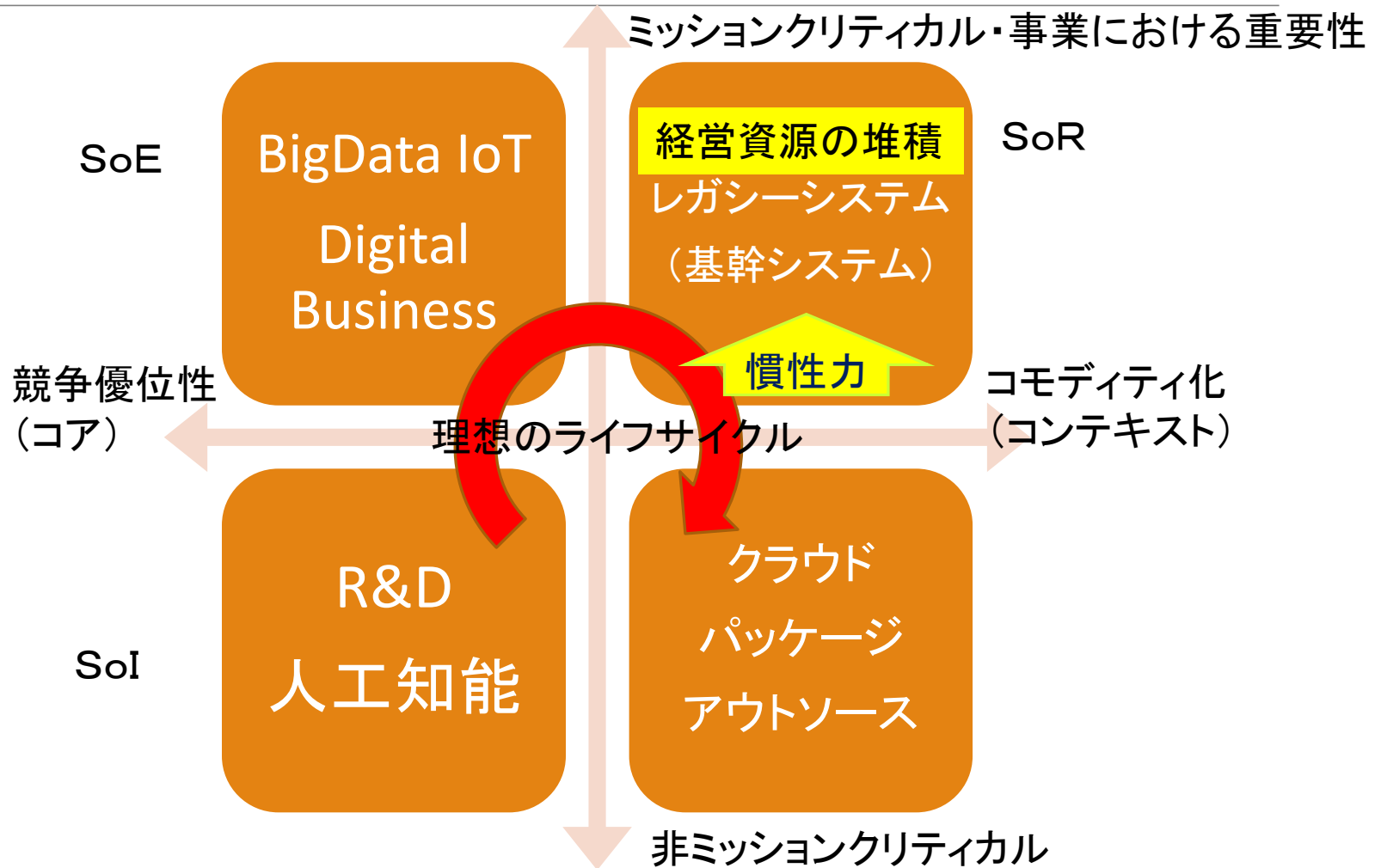
企業（社会）活動の中核となる業務

サービスが停止すると企業活動に甚大な影響
重要性は今でも変わらない

しかし、戦略面では競合優位性を追求する必要がない（コンテキスト）システム

重要だが、戦略的投資対象ではない
メインフレームの維持コストがITコストを圧迫

ITシステムのコア・コンテキスト分析



レガシーシステムのキーワード

「2007年問題(2012年問題)」

メインフレーム開発・保守の中核層である団塊の世代の定年退職に伴う、自社基幹システムを理解するエンジニアが企業内に存在しなくなる懸念

現場では、レガシーシステムを理解するエンジニアの大半は退職

更改・機能追加のためには、資産(ソース、定義、データ、等)を調査する必要がある。

エンジニアの高齢化と暗黙知の喪失が最大のリスク

レガシーシステム問題の考察

老朽化はハードやプログラムだけではない。
業務システムの構成要素全体がレガシー化

ハード、プログラムをモダナイズしても、問題の本質は変わらない(再レガシー化)

業務システムの見直し

- ・ミッションクリティカルな基幹業務の最小化
- ・残りの業務の非ミッションクリティカル化

レガシーシステムのリスク要素

短期的リスク

「メインフレーム」	(技術)
「2007年問題」	(ナレッジ)
「慣性力」	(組織)

プラットフォームの見直し

長期的リスク

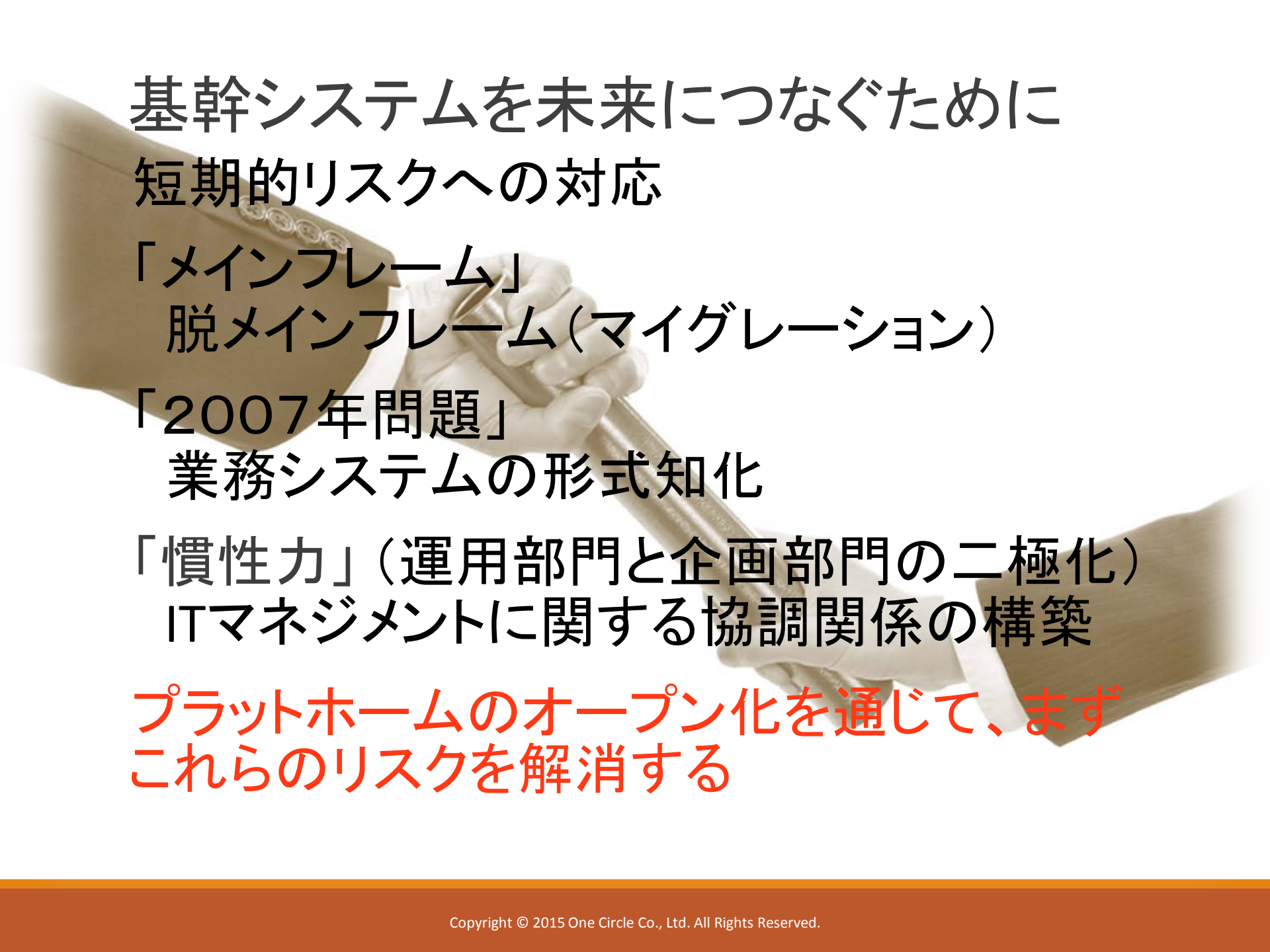
- 「再レガシー化」
- 「経営資源の堆積」
- 「技術の空洞化」

業務システムの見直し



ACTION

2. 「いま」やるべきこと

A background image showing two hands, one in a brown suit sleeve and the other in a white shirt sleeve, shaking over a rolled-up document tied with a string.

基幹システムを未来につなぐために
短期的リスクへの対応

「メインフレーム」
脱メインフレーム（マイグレーション）

「2007年問題」
業務システムの形式知化

「慣性力」（運用部門と企画部門の二極化）
ITマネジメントに関する協調関係の構築

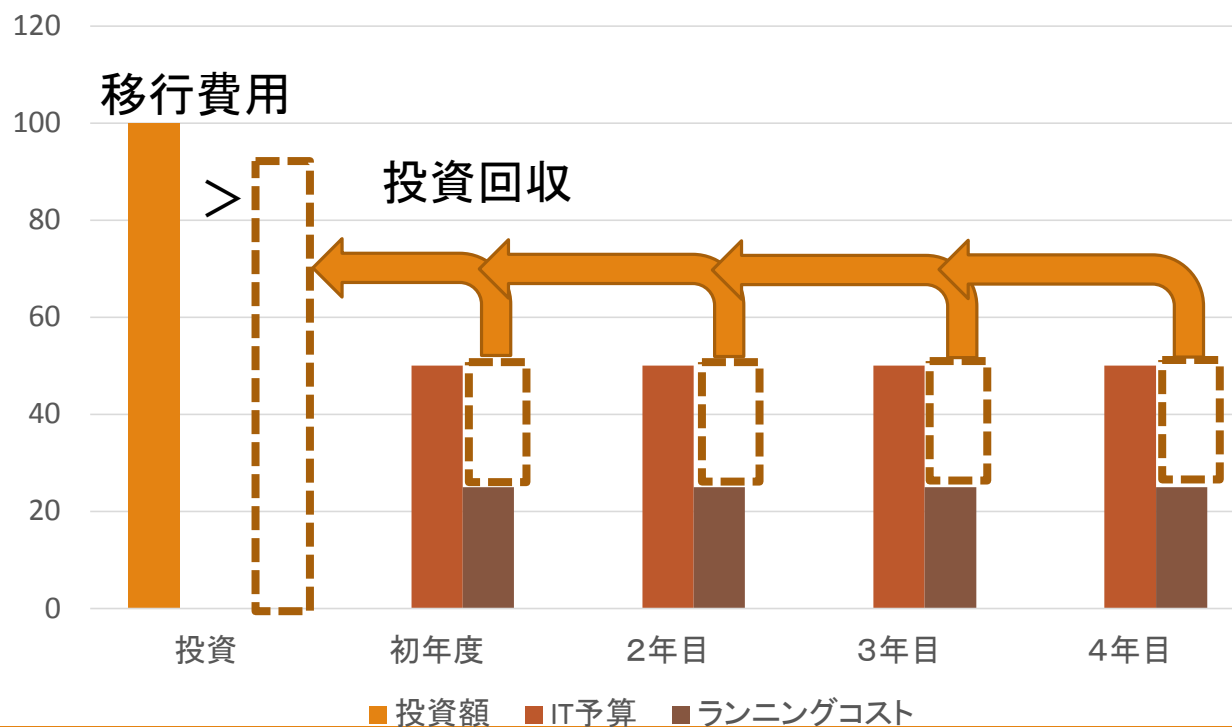
プラットフォームのオープン化を通じて、まず
これらのリスクを解消する

オープン化によるTCO削減効果

プラットフォームのオープン化でTCOが削減できるケースは極めて稀

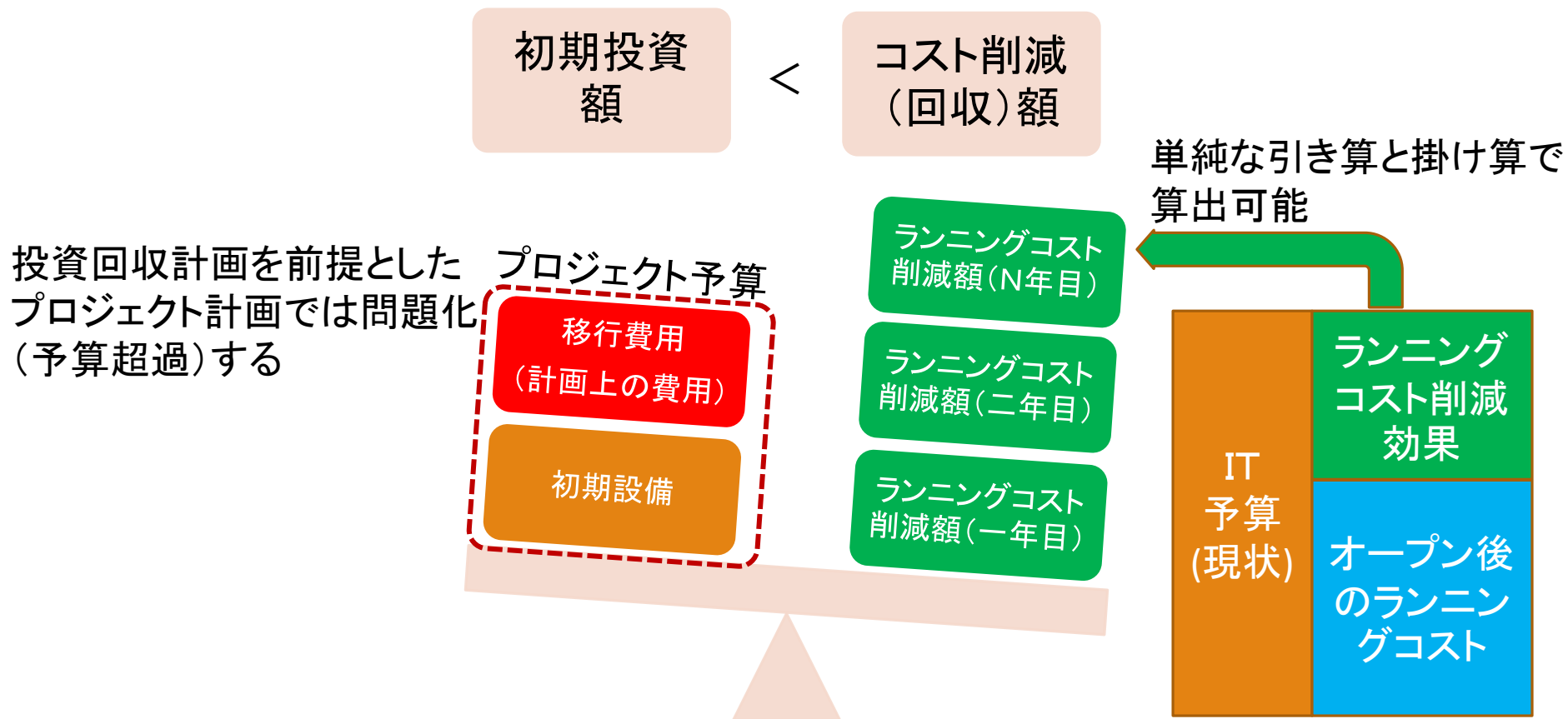
TCO削減＝オープン化に伴う投資額 vs IT費用削減費用 × 回収期間
(削減額が投資額を上回れば、投資価値はありと判断)

オープン化の投資回収計画イメージ

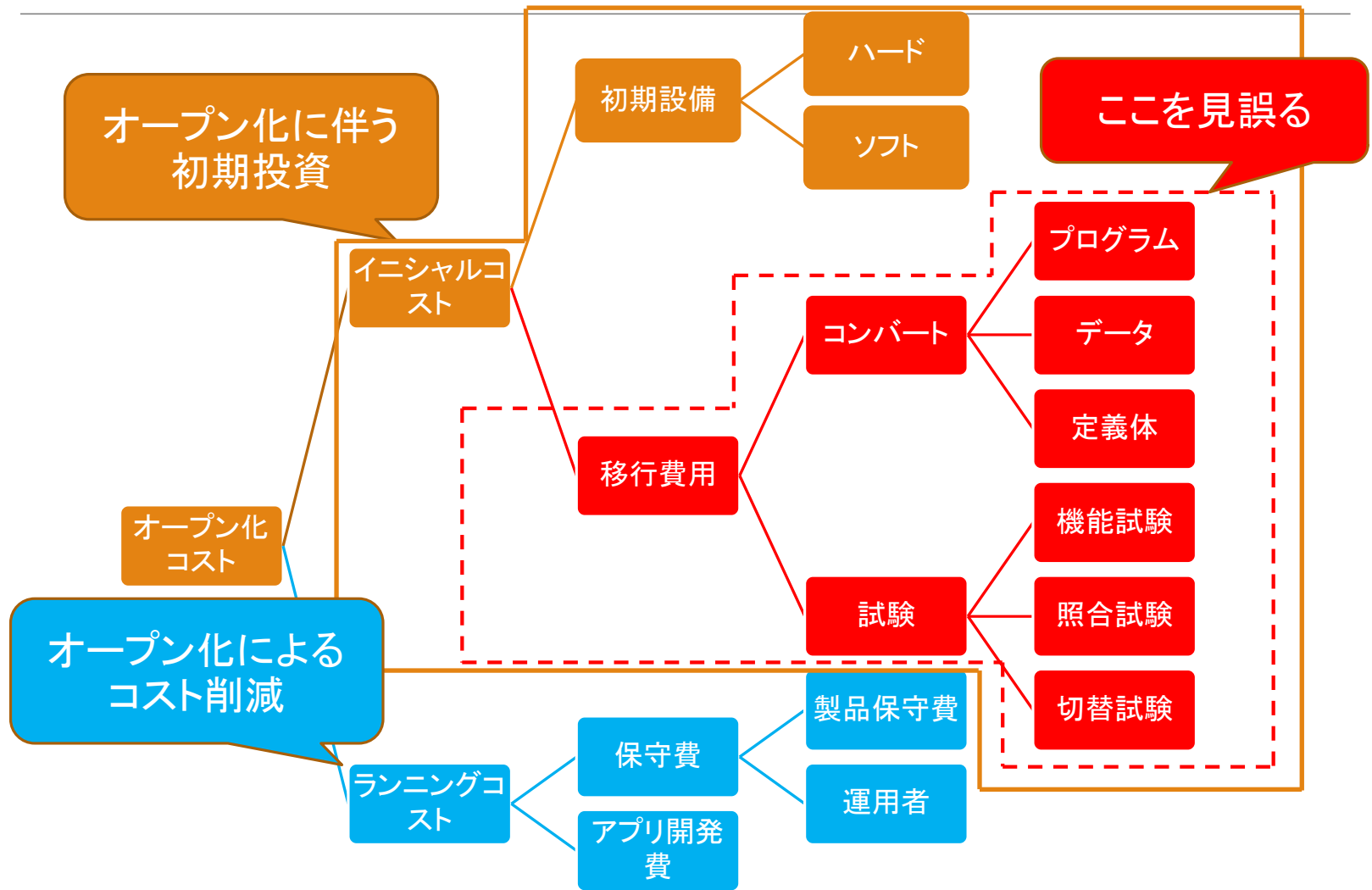


オープン化の失敗ケース

トップダウン(投資回収計画)でオープン化の投資額を設定する



オープン化のコスト構造



こんなはずでは



レガシーシステムの構造と特徴

レイヤ	技術領域
ユーザー層	業務処理の実現に必要な技術領域
メカニズム層	業務プログラムの実行に必要な技術領域
インフラ層	業務システムの稼働に必要な技術領域



レガシーシステム各レイヤの特徴

大分類	技術領域	特徴
ユーザー層	業務処理を実現するために必要な技術領域	・ユーザーの資産である ・<u>移行前後での等価性が求められる</u> ・資産量が膨大かつ管理元が複数存在するケースがある ・調査、検証には業務知識が必要である ・プログラム資産からの調査が可能である
メカニズム層	業務プログラムを実行するために必要な技術領域	・主に製品ベンダ提供の機能である ・ユーザー層に対する互換性が求められる(レベルは方式毎に異なる) ・<u>レガシー特有の機能が含まれている</u> ・<u>機能互換が保証されているオープン製品は無い</u> ・機能バリエーションが豊富かつ複雑である ・プログラム資産調査でも完全な把握は難しい ・ユーザー層が標準化、共通化されていない場合、機能バリエーションが膨大になる危険がある
インフラ層	業務システムが稼動するために必要な技術領域	・主に製品ベンダ提供の基盤機能で実現されている ・<u>オープン系システムと根本的に実現方式(アーキテクチャー)が異なる</u> ・現行システム資産調査では把握できない ・システムの運営を通じた背景や経緯が要件として蓄積されている ・現行SLA、設計書、運用ドキュメント、ヒアリング、実査等で多面的なアプローチで把握が必要 ・過去からの経緯に基づく固有の要件、価値観が反映されている ・過去の経緯は散逸しているケースが多い

オープン化プロジェクトの流れ

稼動中システム
= インプット



計画策定

資産の抽出

全てを洗い出す

方針策定・PoC

取捨選択、実現性の評価

設計～開発

試験

ユーザー層

マイグレ手順決定
データ移行方針
移行品質基準

プログラム変換
データ変換
データ整合確認

照合試験
業務運用
切替試験

メカニズム層

ユーザ層に合わせた
製品検証・選定
不足機能の扱い

製品機能検証
機能開発
ユーザー層との
結合確認

インフラ層

現行SLAの調査
オープン化で継承
する要件の取捨選
択

システム再構築

SLA試験
運用試験
外部シス
テム連動

このプロセスを経て
初めてコスト算出が可能に

レガシーシステム各レイヤの要素技術

大分類	中分類	技術領域	具体例(一部)
ユーザー層	プログラム	業務仕様、プログラム言語	COBOL、PL/I、4GL
	入出力形式	業務データの格納先、出力先フォーマット	テーブル、帳票、画面
	データ	業務データ実体、符号化方式、固有データ型	EBCDIC、COMP-3
メカニズム層	API	業務プログラムが呼び出している機能	DBアクセス、電文送受信
	フレームワーク	業務プログラムを実行するために必要な機能	オンライン、バッチ
	入出力媒体	業務データの格納先、出力先媒体	DB、VSAM、磁気テープ
	外部連携	外部・周辺システムの連携機能	EDI、情報系、Web
	ユーティリティ	ベンダー提供、個別開発ツール	ソート、DB、運用・保守
インフラ層	性能・スケーラブル	システムに求められる性能条件	トランザクション数、バッチ処理時間
	可用性	システムの運転継続性に関する条件	稼働率、定期メンテナンス
	運用	業務処理、システムの運転、運用に関する条件	自動運転、システム監視
	保守	システム、業務プログラムの維持管理に関する条件	アプリ開発、リリース手順
	セキュリティ	各種認証、アクセス認可に関する条件	端末認証、アクセス認可
	他	システム固有の要件、背景、過去の経緯	運用ルール、管理ルール

マイグレーションについて

レガシーシステムの技術要素すべてに整合が取れた移行方針を立てる必要がある。

TCO削減を達成する秘訣:「余計なことをしない」
今あるものを、そのままに
あきらめが肝心

PJを問題化させない秘訣:「最初に決めて途中で変えない」

開始前にすべての分析を完了し方針を立てる
(つもりで臨む)

方針を決めたら絶対に変えない

2007年問題について

メインフレームの単純リプレイスによって
超複雑な基幹システムの設計思想が
エンジニアとともに失われる

オープン化にともなう業務仕様の調査を通じて
暗黙知の形式知化と世代間の引継ぎを行う

慣性力について

IT運営での企画部門と運用部門の二極化

企画部門

- 推進力(変革重視)
- IT計画を策定

運用部門

- 慣性力(現状維持)
- システムを熟知

オープン化は企画部門主導が大半だが、システムを熟知する運用部門の協力が無ければ成功しない。

ITマネジメントに関する両者の協調関係の構築と意思決定の一本化

前半のまとめ

「いま」やるべきことは、リスク対策としてのオープン化

- 脱メインフレーム
「余計なことをしない」
「最初に決めて途中で変えない」
「トップダウン(投資回収ベース)で計画しない」
- 2007年問題
業務システムの形式知化
- 慣性力(IT運営の二極化)の対応
企画部門と運用部門の協調関係



**NEXT
STEP**



3. 「これから」やるべきこと

レガシーからの真の脱却のために

老朽化はハードやプログラムだけではない。
業務システムの構成要素全体がレガシー化

ハード、プログラムをモダナイズしても、問題の本質は変わらない(再レガシー化)

業務システムの見直し

- ・ミッションクリティカルな基幹業務の最小化
- ・他の処理の非ミッションクリティカル化

参考書籍の紹介



題名: ライフサイクルイノベーション

成熟市場+コモディティ化に効く14のイノベーション

作者: ジェフリー・ムーア, 栗原潔

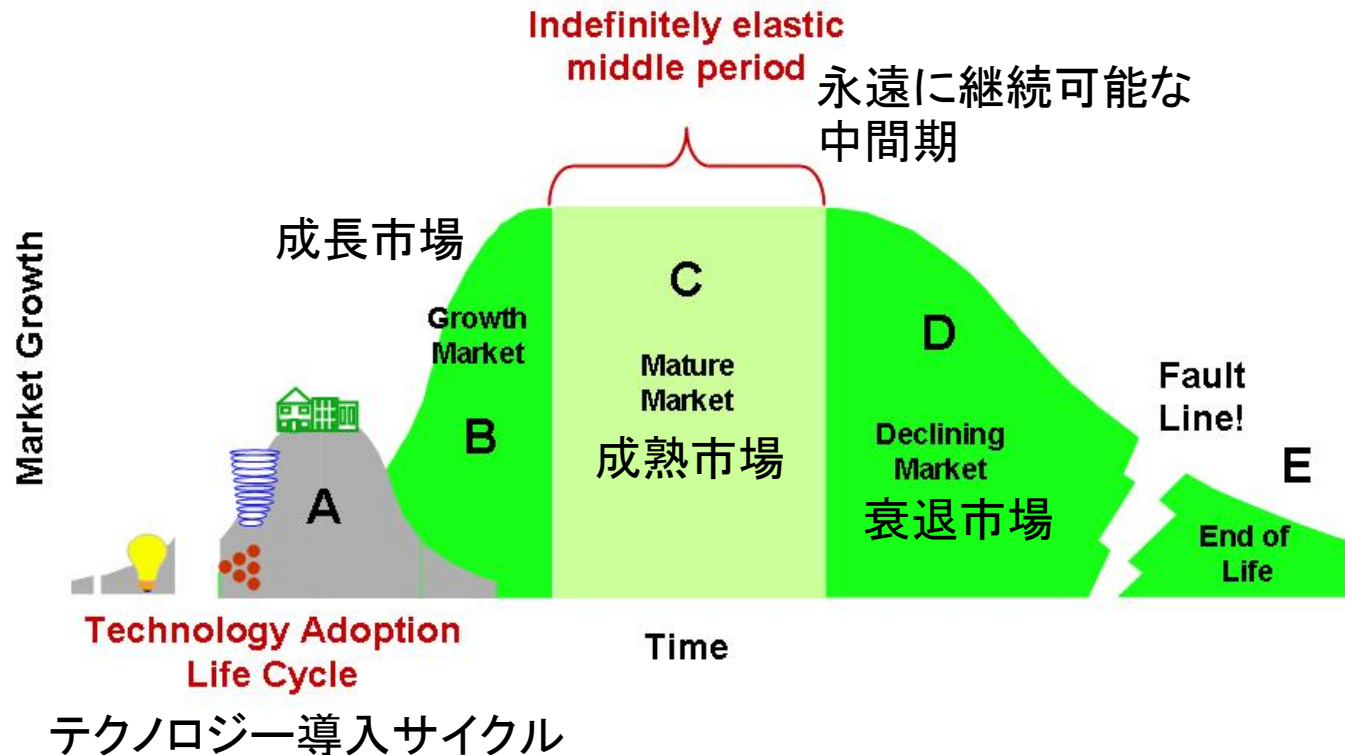
出版社/メーカー: 翔泳社

発売日: 2006/05/16

事業ライフサイクルの中で企業が永続的にイノベーションを起こし、成長するための経営戦略論

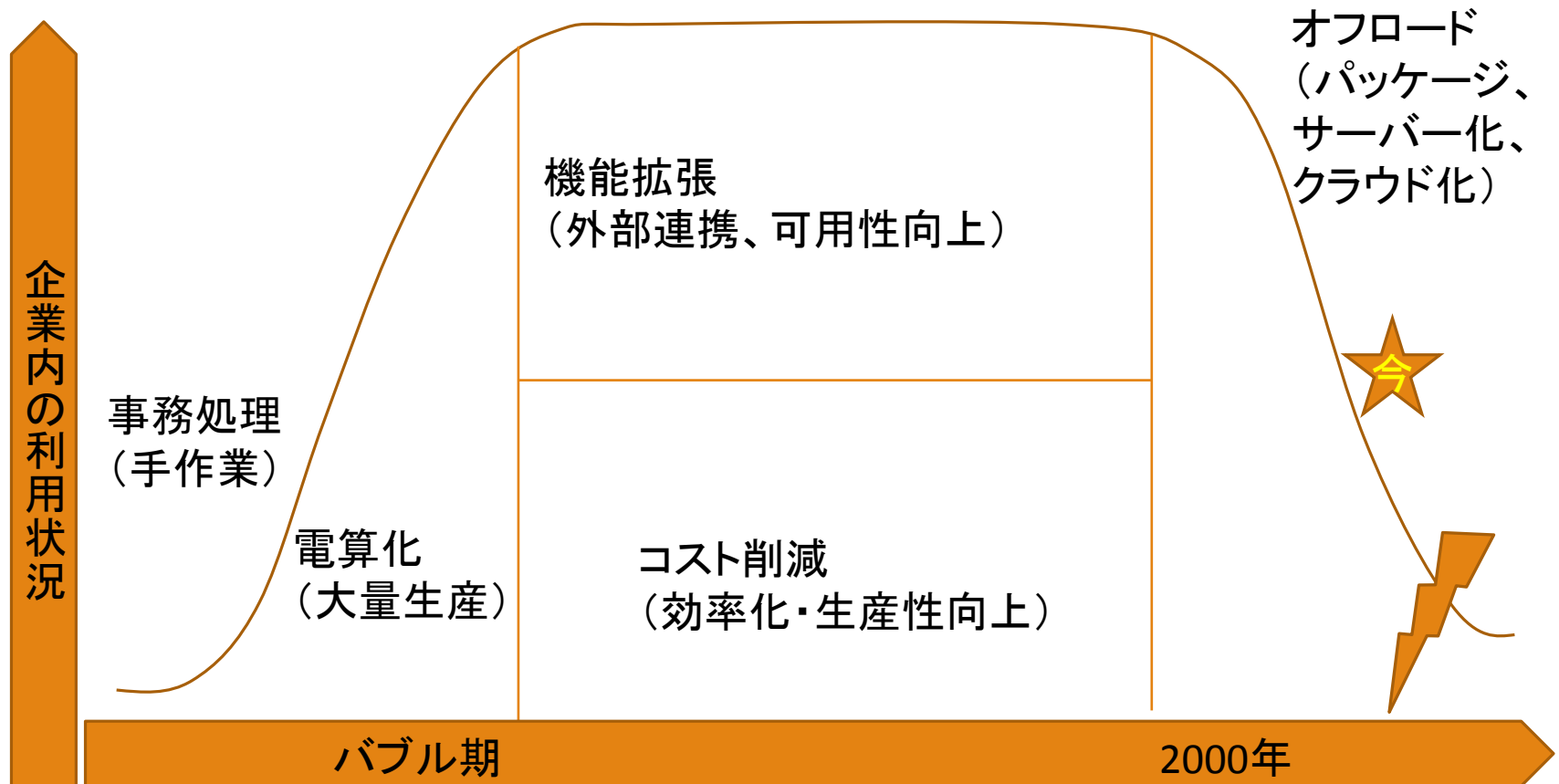
カテゴリー成熟化ライフサイクル

The Category Maturity Life Cycle



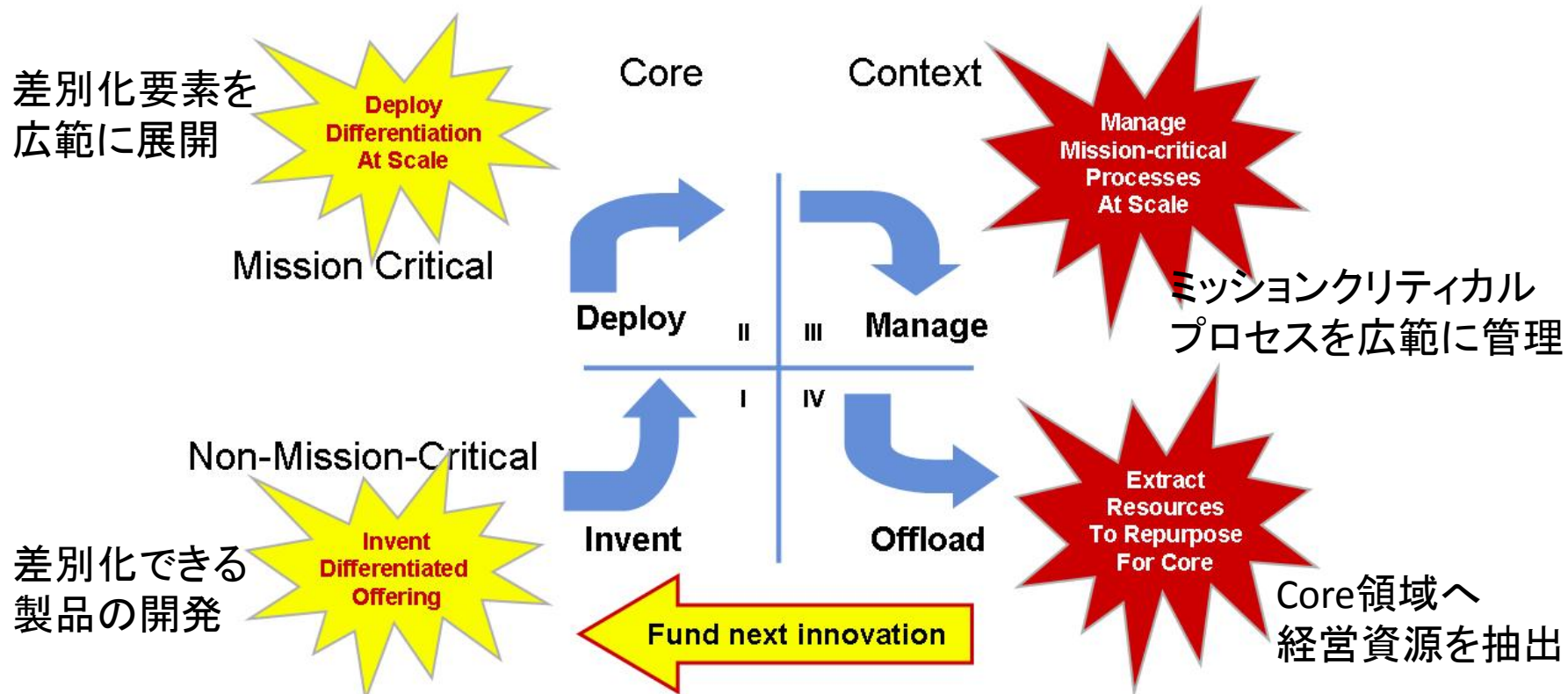
Copyright © Geoffrey A. Moore, 2005, from the book "DEALING WITH DARWIN"

基幹システムのライフサイクル



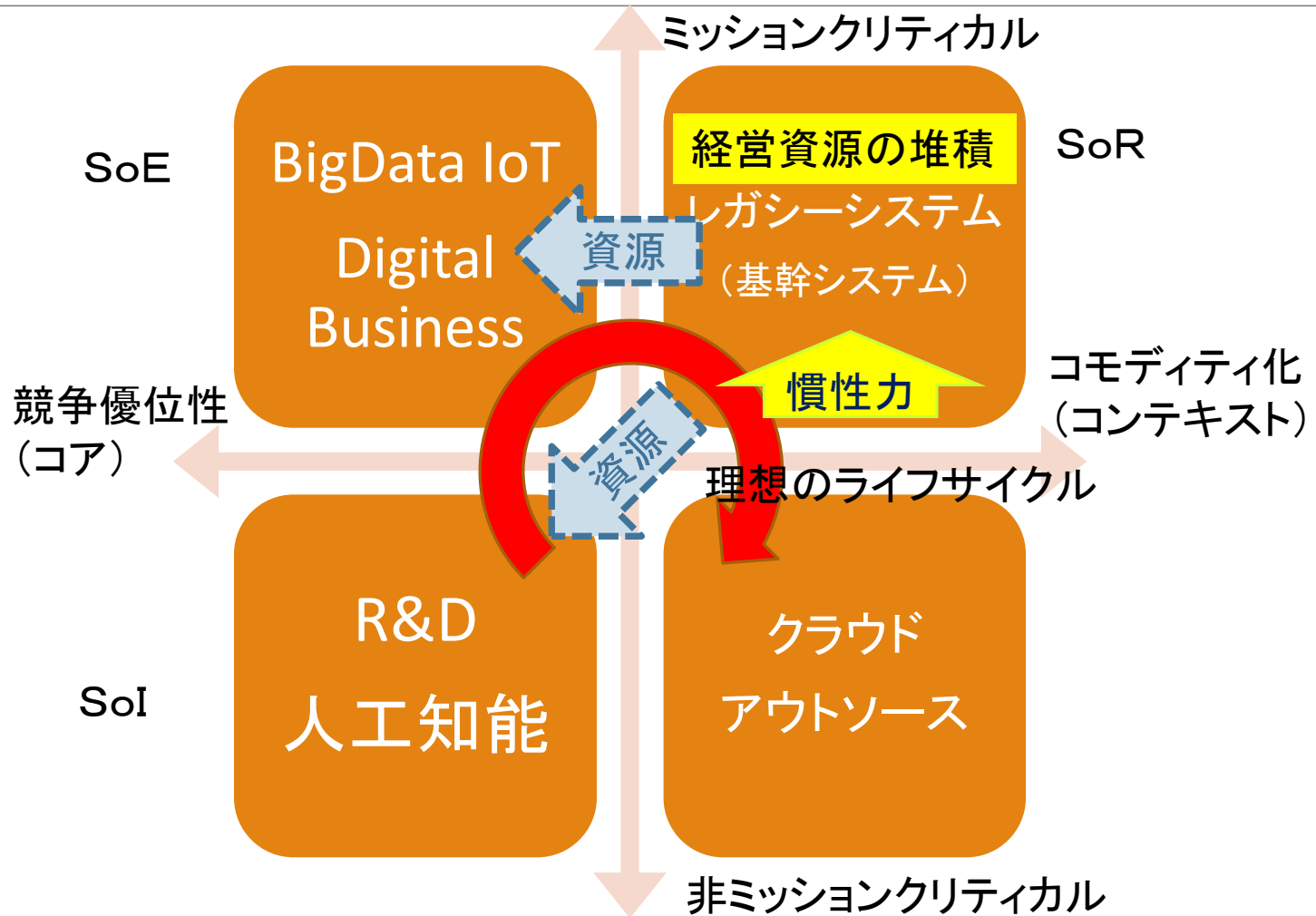
コア・コンテキスト分析

The Cycle of Innovation



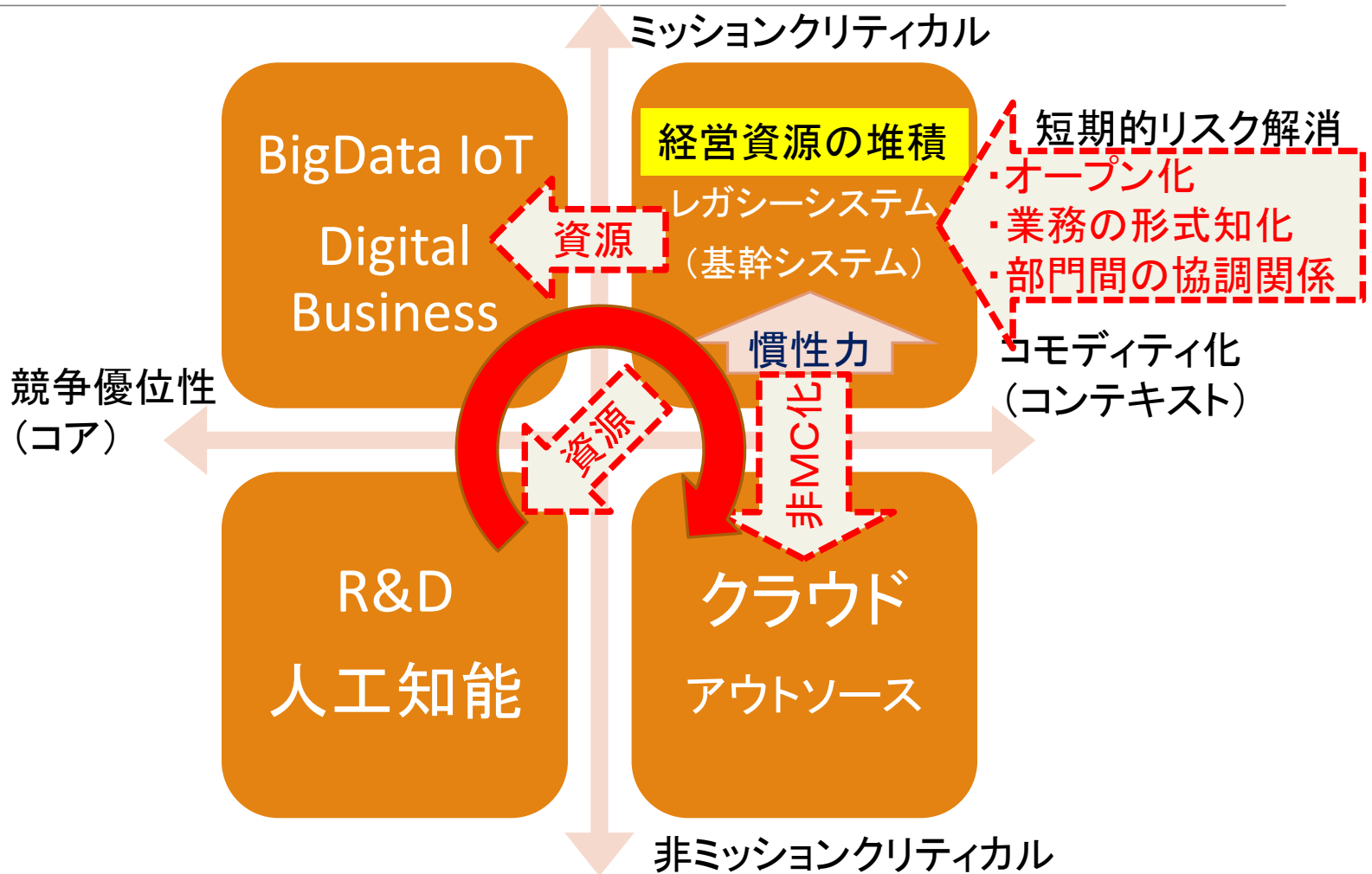
Copyright © Geoffrey A. Moore, 2005, from the book "DEALING WITH DARWIN"

ITシステムのコア・コンテキスト分析



競争優位性を確保するために必要な象限にIT資源が配分されない

レガシーシステムへの見直し



慣性力の管理と非MC化による、MC・コンテキストからのIT資源の抽出と再配分

MC・コンテキストへのアプローチ

“MC・コンテキストから経営資源を抽出する5階層モデル

1. 集中化: 業務を一つの権限下に集約し、間接費を削減すると共に、リスク管理の意思決定の権限を一元化する
2. 標準化: 業務の多様性を縮小し、コストの削減とリスクの低減を行う
3. モジュール化: システムをサブシステムに分割し、将来のコスト削減のためにインターフェースの標準化を行う
4. 最適化: 冗長なステップの排除、標準的な手続きの自動化、業務の単純化、低コスト資源での置き換え、プロセスの定量的測定によるモニターと統制を行う
5. アウトソース: プロセスを企業外に委託し、間接費、変動費を削減し、将来必要となる投資額を最小化する。モニタリングを含むサービスレベル合意を締結する“

短期

中長期

ジェフリー・ムーア著「ライフサイクルイノベーション 成熟市場＋コモディティ化に効く14のイノベーション」
、栗原潔訳 翔泳社 2006年 P286

後半のまとめ

「これから」やるべきことは、業務システムの見直しと非ミッションクリティカル化

ミッションクリティカル領域（基幹システム）と非ミッションクリティカル領域の分離

限界まで絞り込まれた基幹システムを永続化させるための仕組みづくり

非ミッションクリティカル領域はオフロード化によりIT資源の流動性を高める。

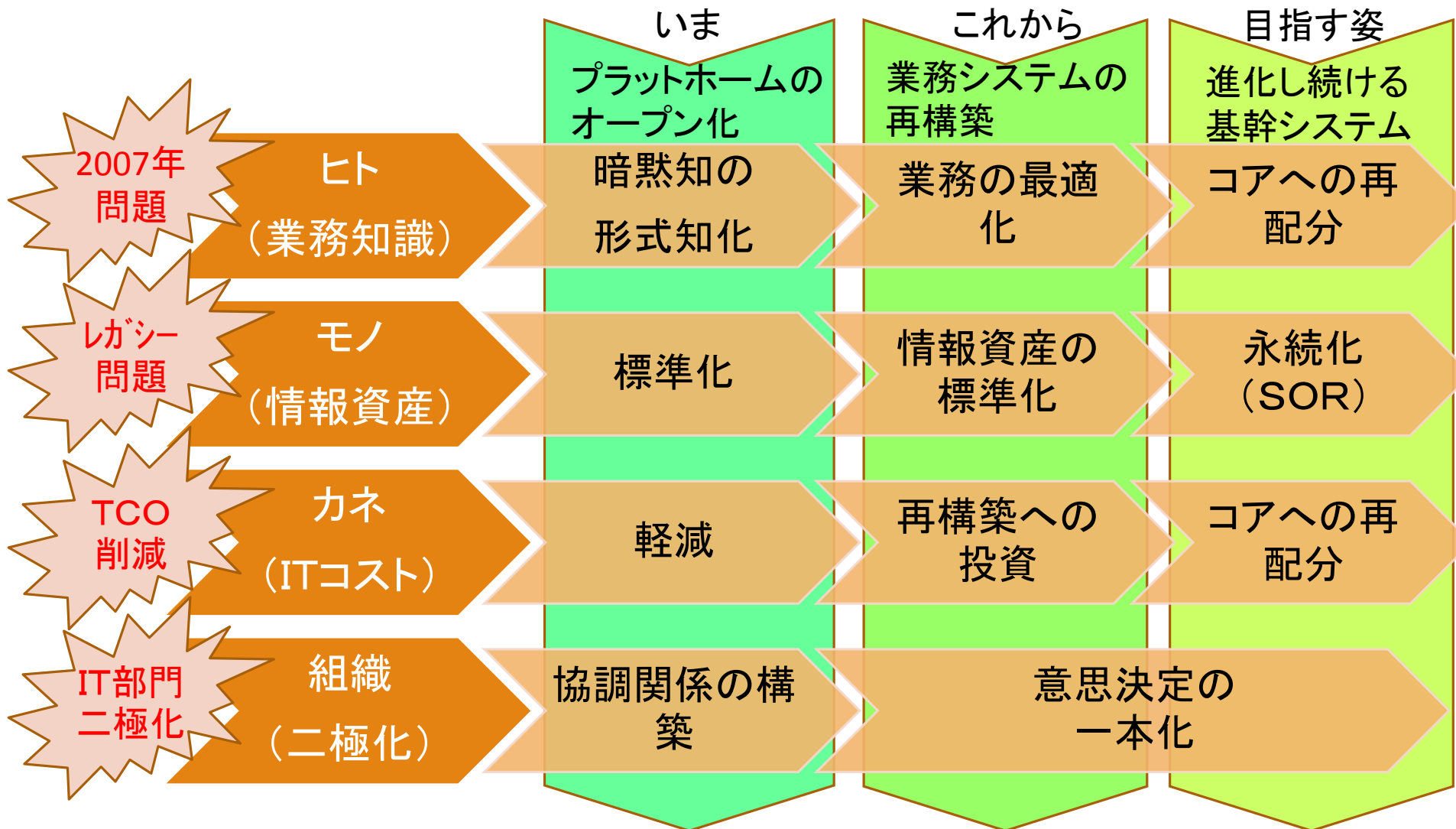


GOAL



4. まとめ

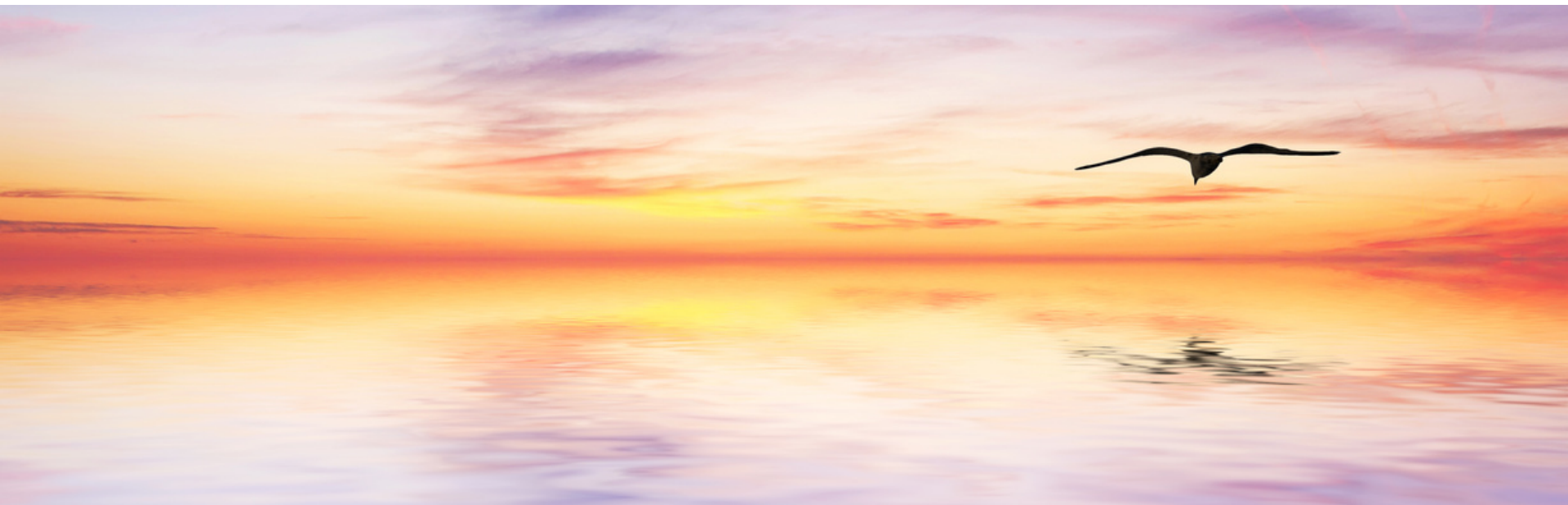
レガシーシステムの将来に向けて



参考文献

「ライフサイクルイノベーション
成熟市場＋コモディティ化に効く14のイノベーション」
ジェフリー・ムーア著 栗原潔訳
翔泳社 2006年初版

ご清聴ありがとうございました



お気軽にご連絡ください

株式会社ワンサークル

ホームページ: www.1circle.co.jp

TEL: 045-550-3495

FAX: 045-550-3496

E-mail: info@1circle.co.jp

