高速開発を実現するソフトウェア開発技術

2014年 9月 2日

株式会社NTTデータ
木谷　強

NTT DATA

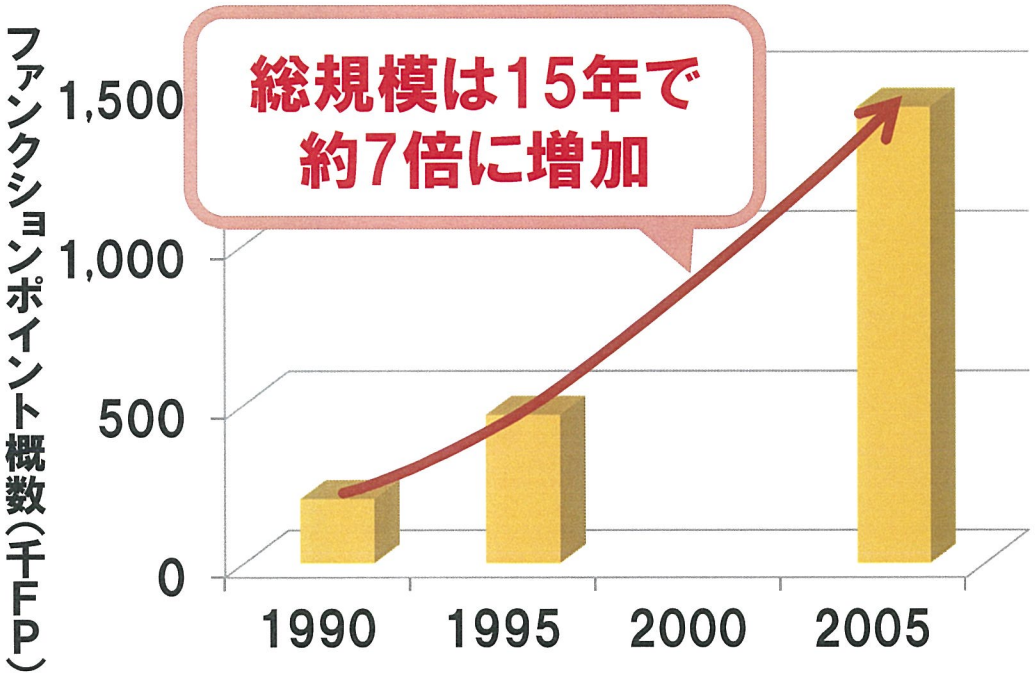

情報サービス産業を取り巻く
環境の変化

ソフトウェア開発の大規模化・短期化

NTT DATA

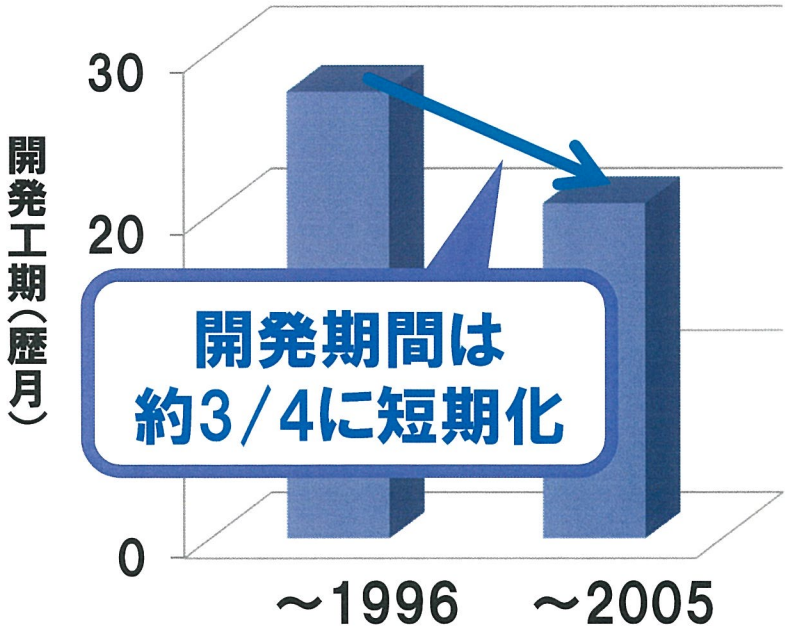
企業が保有するソフトウェアは激増、開発期間は短期化している

企業が保有するソフトウェア総規模の推移（米国）



※全分野ソフトウェアの平均FP概数（国防総省を除く）

ソフトウェア開発工期の推移（米国）



※10000FP規模プロジェクトの平均工期

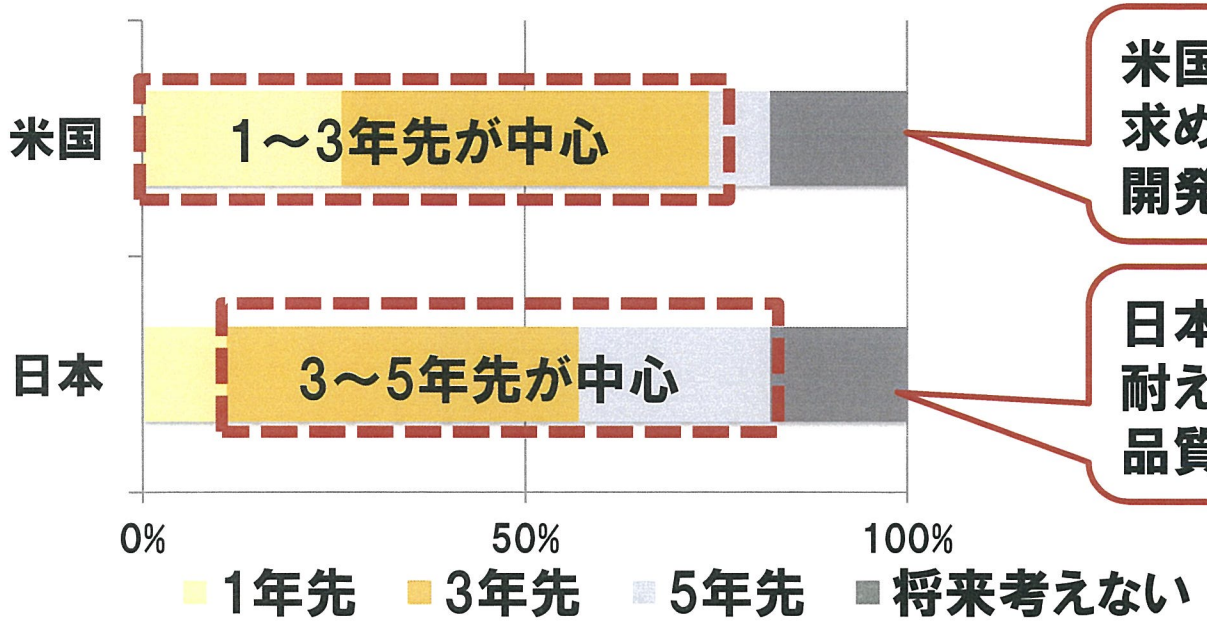
Capers Jones著「ソフトウェア開発の定量化手法第2版／第3版（共立出版）」掲載データよりグラフ作成

海外でコモディティ化している価値：速さ

NTT DATA

海外では短期間での効果を求められるため 開発期間が日本より短い

新規IT投資時に想定する投資効果の有効期間



米国では、短期間での効果を求められるため、開発スピードを重要視。

日本では、長期の運用に耐えられるよう、品質・ドキュメント・証跡が必要。

開発期間の短期化が必須

資料：経済産業省「IT 戦略と企業パフォーマンスに関する日米韓の国際比較」（平成19年）

日本で実現してきた価値：品質

NTT DATA

日本は欧米と比較して10倍～20倍の品質を実現

ソフトウェアの不具合数に関する国際比較

	日本	米国	インド	欧州他
プロジェクト数	27	31	24	22
ソフトウェアの品質 システム導入後1年間に発見された1Kstepあたりの不具合報告（中央値）	0.020	0.400	0.263	0.225

出典：Cusmano.M等「Software Development Worldwide : The State of the Practice」
（IEEE Software Nov./Dec. 2003, pp28-34）

米国の20分の1

欧州・インドの
約10分の1

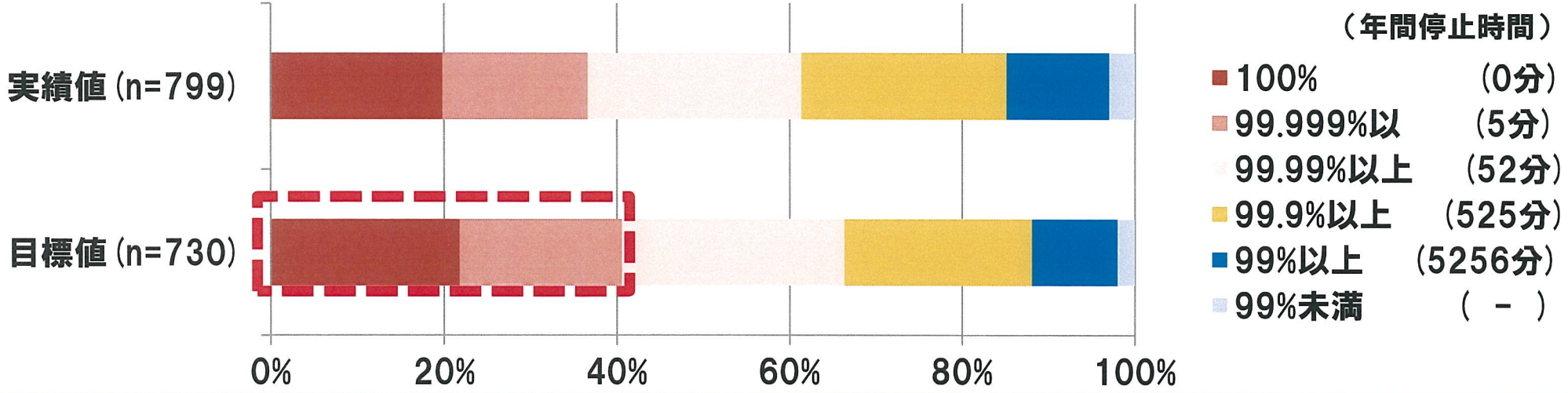
日本で実現してきた価値：品質

NTT DATA

基幹系情報システムの稼働率実績で 約4割の企業が年間停止時間5分以下を達成

基幹系情報システムの稼働率

※　稼働率は、計画停止の時間を計算式から除いたもの



日本の従業員1,000人以上の企業の「基幹となる情報システム」（含・情報系システム）の稼働実績

	回答件数	月間停止時間（時間）合計
合計	230	298.45

1社あたりの月間停止時間　1.7324

出典：JUAS 「第17回企業IT動向調査2011（10年度調査）」

日本のソフトウェア開発の方向性

NTT DATA

日本の価値である「高品質・高信頼」を維持しつつ
「短納期・低コスト」開発を実現できなければ
日本のシステム開発従事者は生き残れない。

生産性の向上

- 人に依存しない開発
- 少ないリソースで大量生産

＋

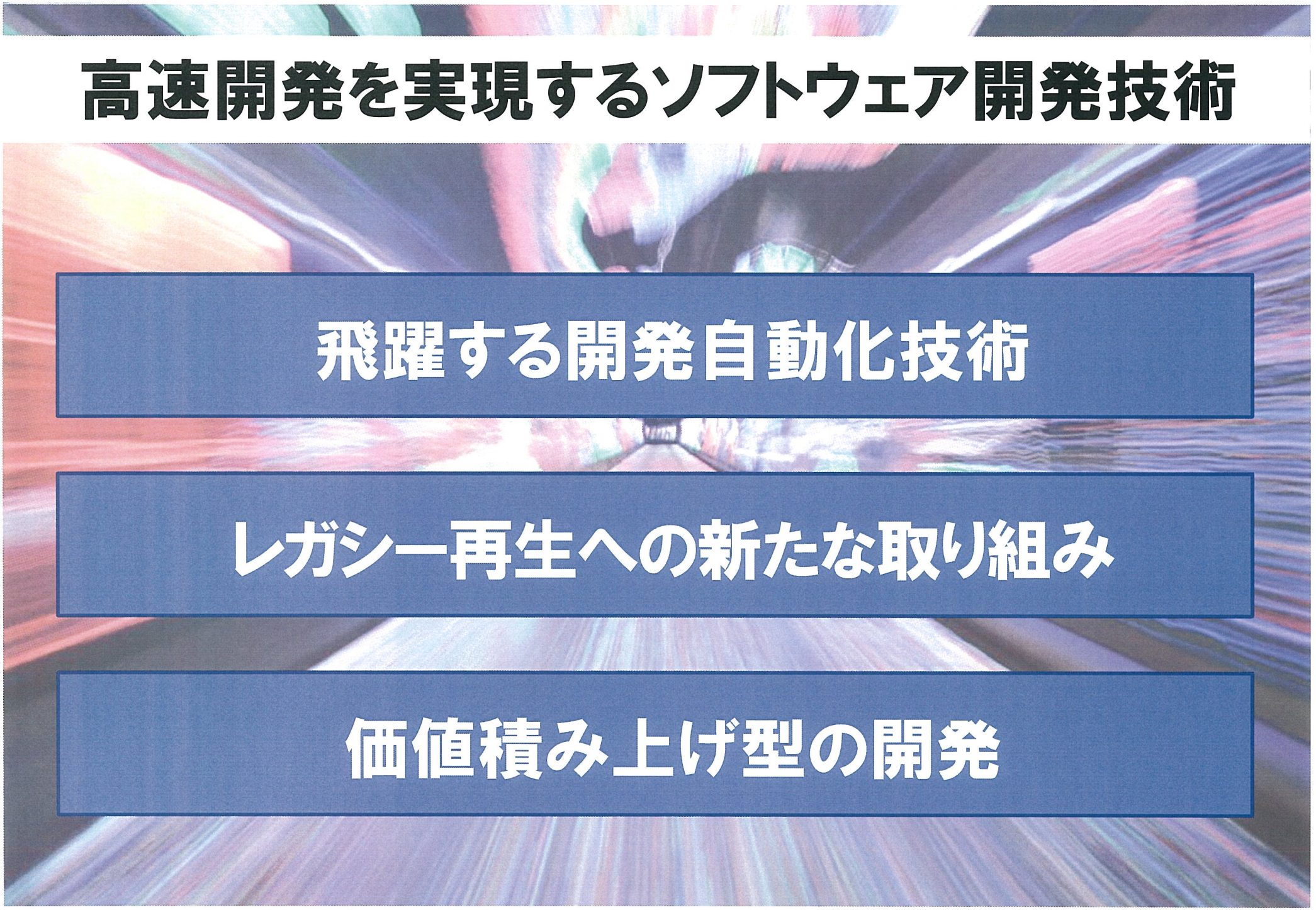
品質の確保

- 人的ミスの排除
- スキルによらない品質の画一化

開発の人的要素を極力排除する技術が有効

ソフトウェア開発自動化

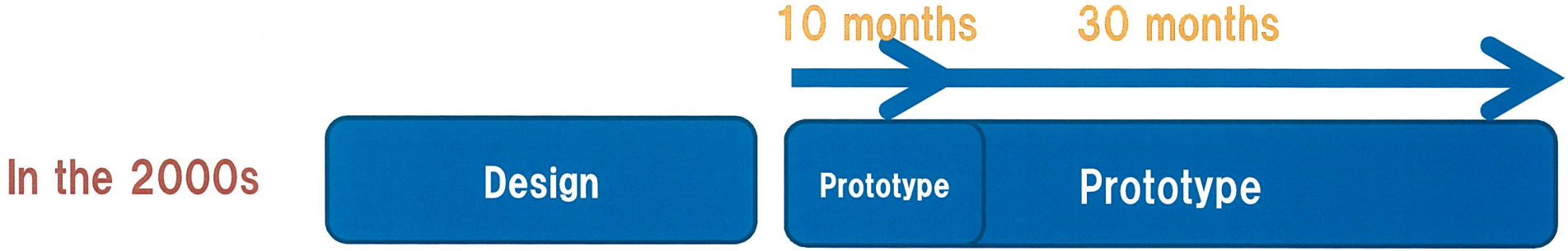
高速開発を実現するソフトウェア開発技術

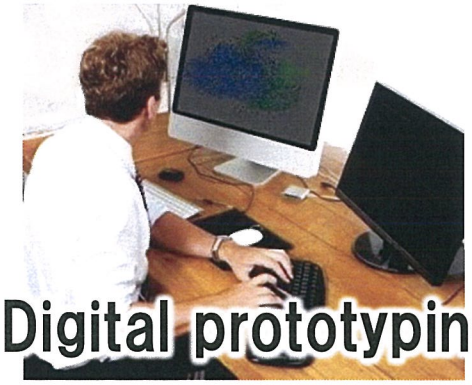


自動車業界に学ぶ

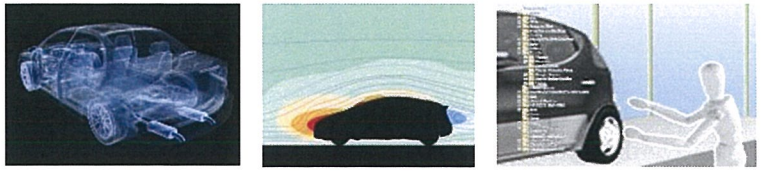
NTT DATA

自動車業界では、計算機能力を最大限に活かす「デジタル生産システム」によって、新車開発の大幅なスピードアップを実現





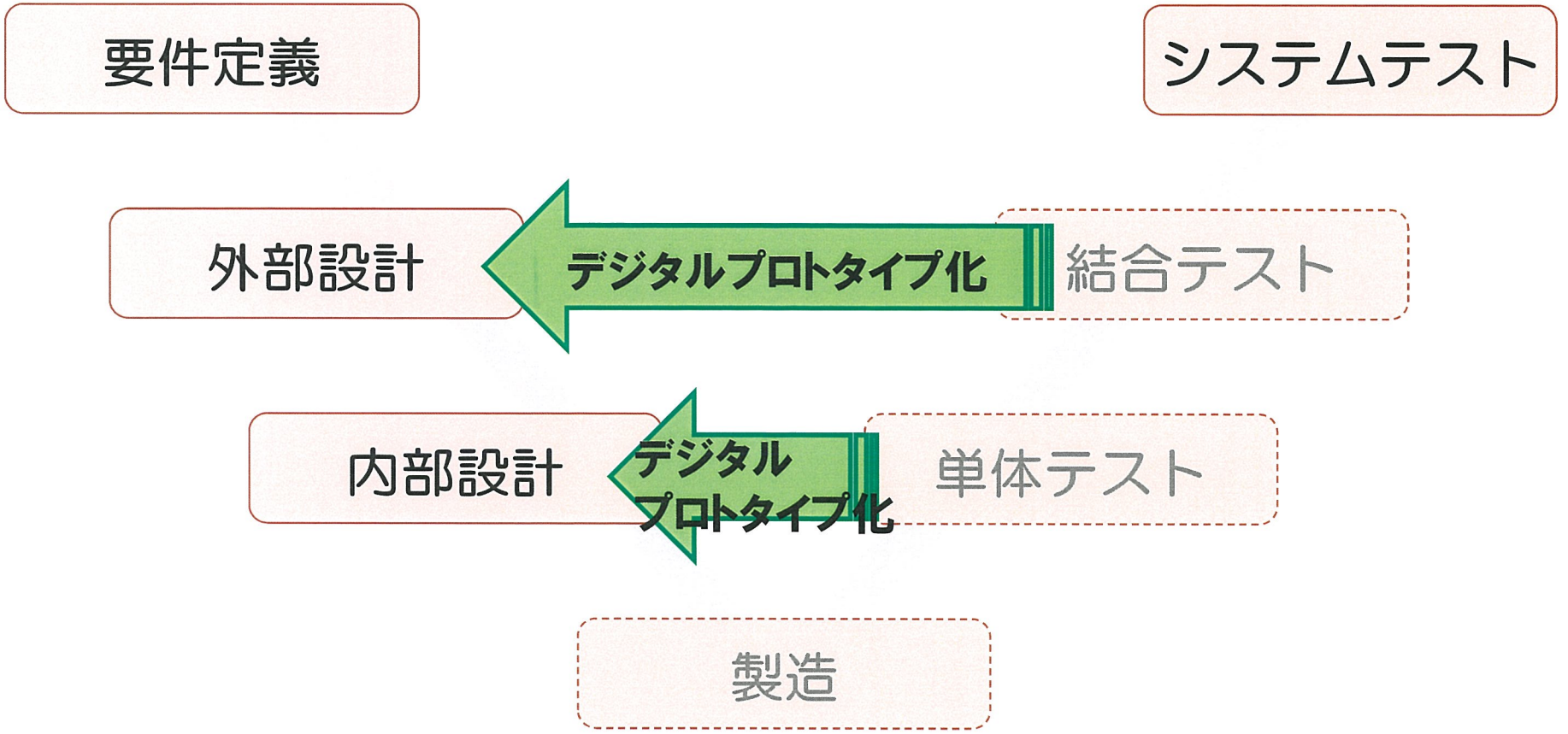
Digital prototyping



ソフトウェア開発へのマッピング

NTT DATA

テストの徹底的な上流シフトを実現



飛躍する開発自動化技術

ソフトウェア開発の自動化領域

NTT DATA

開発・管理・運用の各領域で自動化が取り組まれている。

自動化8領域

現行解析の自動化	設計の自動化	コーディングの自動化	テストの自動化
レガシーコードからの正確な仕様解析	設計段階での整合性自動検証	複雑・多様なロジックの完全自動生成	試験項目の自動生成と自動実行

プロジェクト管理の自動化	ライブラリ管理の自動化	運用の自動化	システム基盤構築の自動化
管理情報の集計・可視化	ビルドやテスト環境へのリリースの自動化	システム運用作業の自動化	システム基盤の自動インストール・設定

NTTデータのソリューション

NTT DATA

TERASOLUNAによる開発プロセスの自動化。

自動化8領域

現行解析の自動化	設計の自動化	コーディングの自動化	テストの自動化
TERASOLUNA Reengineering	TERASOLUNA DS	TERASOLUNA IDE/ViSC	TERASOLUNA RACTES

プロジェクト管理の自動化	ライブラリ管理の自動化	運用の自動化	システム基盤構築の自動化
PMWB	SDWB	ITSM-WB	PRORIZE 基盤構築自動化ツール

現行解析の自動化 TERASOLUNA Reengineering

NTT DATA

現行システムのプログラムや運用ログから、処理や機能、仕様を正確にスピーディに低コストで解析

現行システム

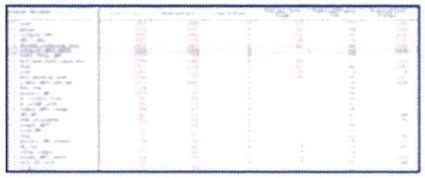
COBOL、JCL、
Java、PL/I 対応

ソースコード

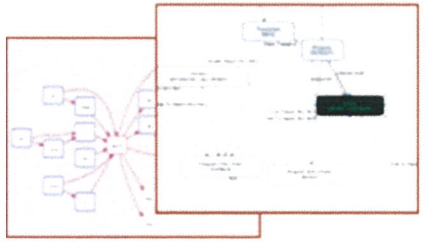
プログラム解析

資産整理

現行分析の範囲、コスト、リスク等の把握のための分析資料作成



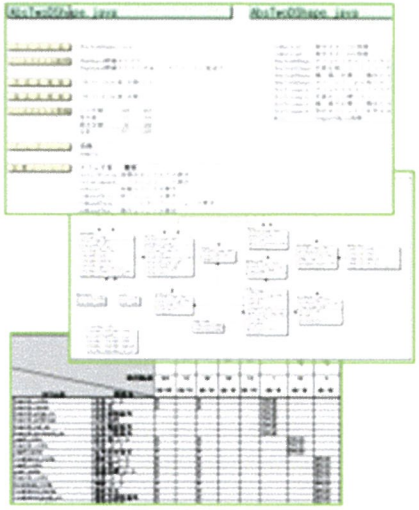
不要・不稼動資産分析



資産構造分析

「処理」の理解

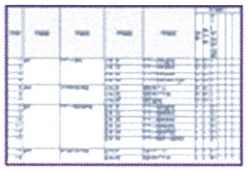
プログラム設計書・内部設計書レベルの成果物生成



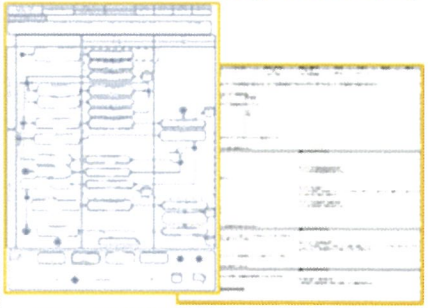
業務処理ロジック抽出

「機能、仕様」の理解

外部設計書・要件定義書レベルの成果物生成（研究中）



データ編集仕様抽出

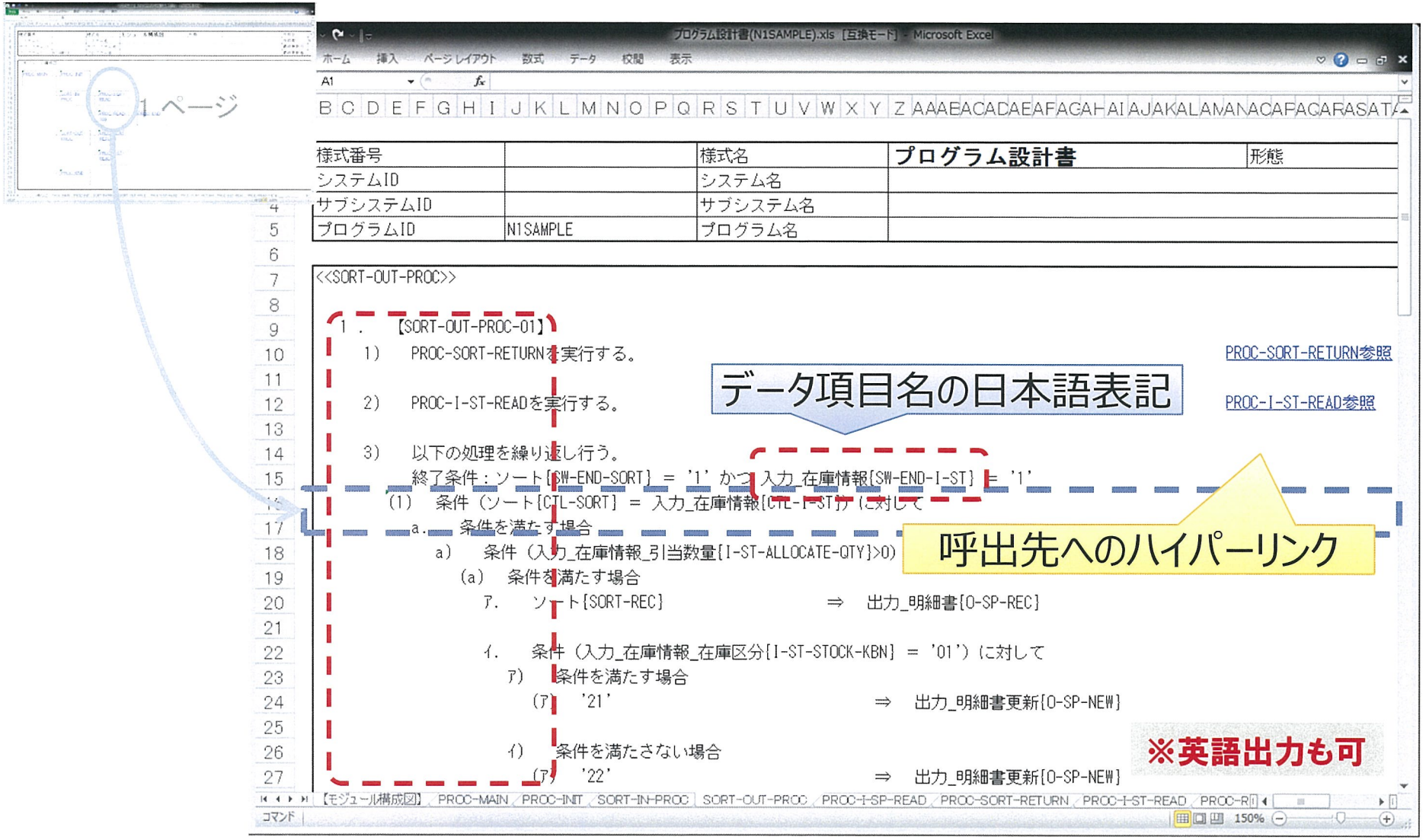


個別業務ルール抽出

現行解析の自動化 TERASOLUNA Reengineering

NTT DATA

プログラムを自動解析し、処理内容を分かり易く表示します。

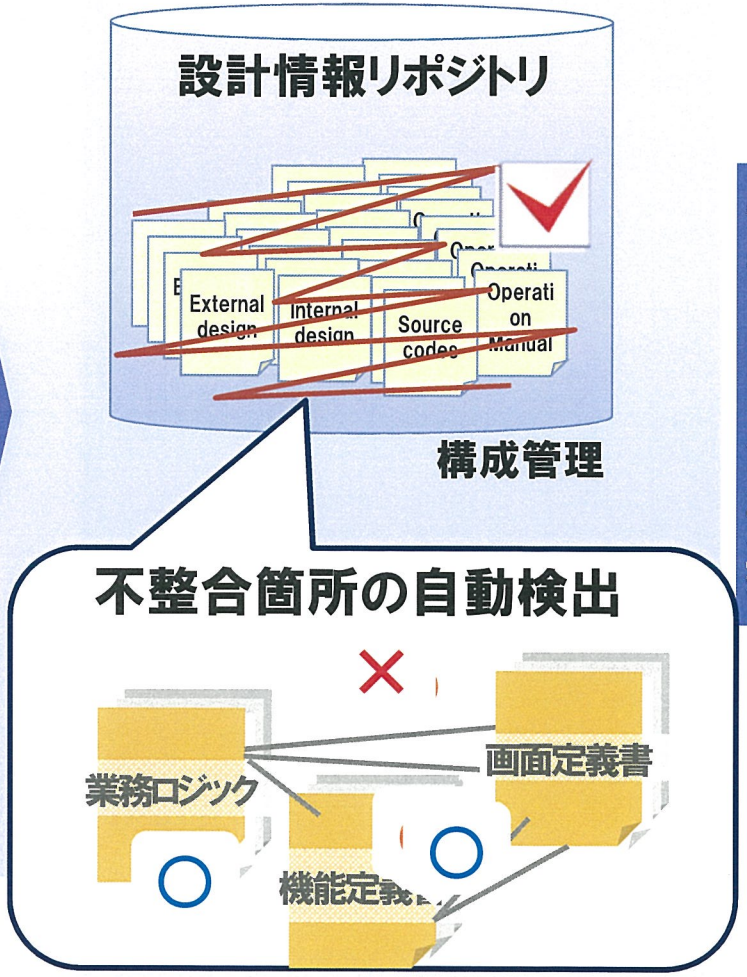


品質チェックの自動化 TERASOLUNA DS

NTT DATA

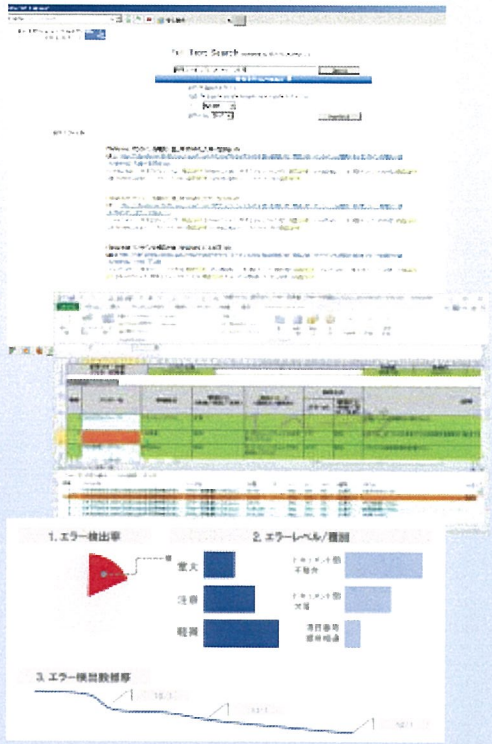
設計情報間の整合性を自動的にチェックし、形式チェックにかかる人的作業を削減、後工程のバグ発生を抑制

設計情報のコミット



情報出力・入力支援

・全文検索結果
・不整合箇所レポート　等

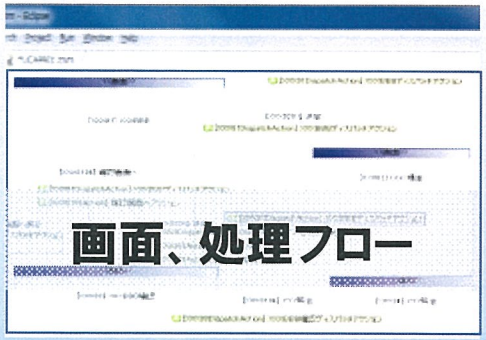


コード生成の自動化 TERASOLUNA IDE/ViSC

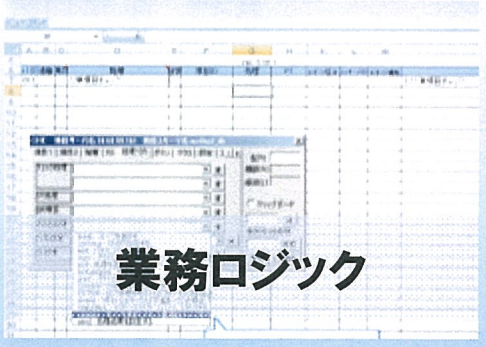
NTT DATA

設計情報からコード、設定ファイル、テストケース等を自動生成し、製造、単体テスト、機能結合テストを大幅短縮

TERASOLUNA IDE

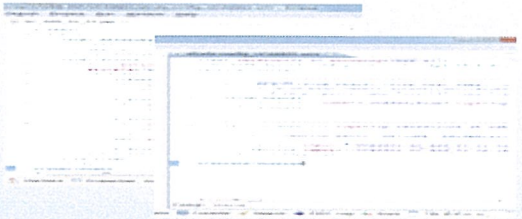


TERASOLUNA ViSC

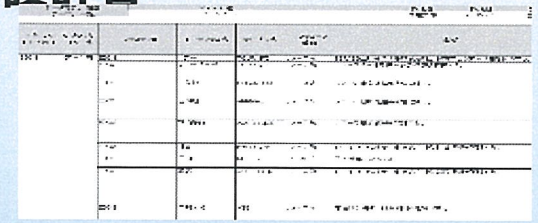


自動生成

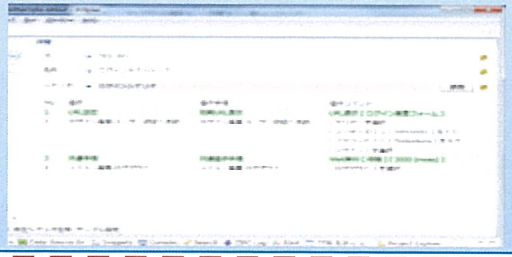
・コード、設定ファイル



・設計書



・テストケース



自動生成

・自動実行可能なテストスクリプト

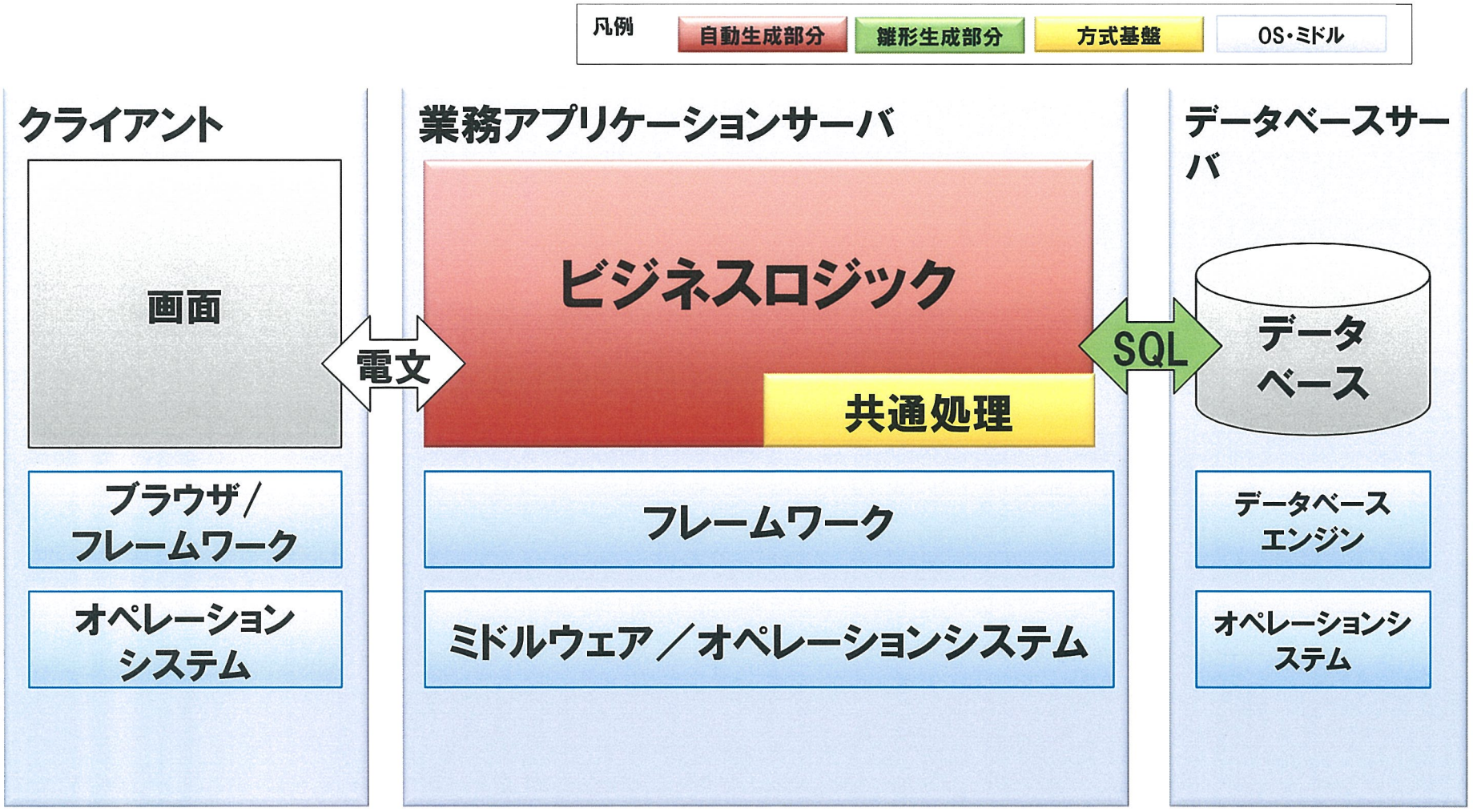
自動化により大幅短縮

設計 | 製造 | テスト

コード生成の自動化 TERASOLUNA IDE/ViSC

NTT DATA

自動生成部分をビジネスロジックに限定することで100%の自動化を実現



テストの自動化 TERASOLUNA RACTES

NTT DATA

プログラムからテスト項目表、テストシナリオの雛形を自動生成、テストシナリオからテストコードを自動生成

```
public class SampleClass {↵
    /**↵
     * 引数a, bともに正常値の場合trueにな
     */↵
    public boolean sampleMethod(int a, i
        if (a > 0 && a < 10) {↵
            if (b > 10 && b < 20) {↵
                return true;↵
            }↵
        }↵
        return false;↵
    }↵
}↵
```

プログラムコード

テストデータ候補値を抽出し
テストの網羅性向上を支援

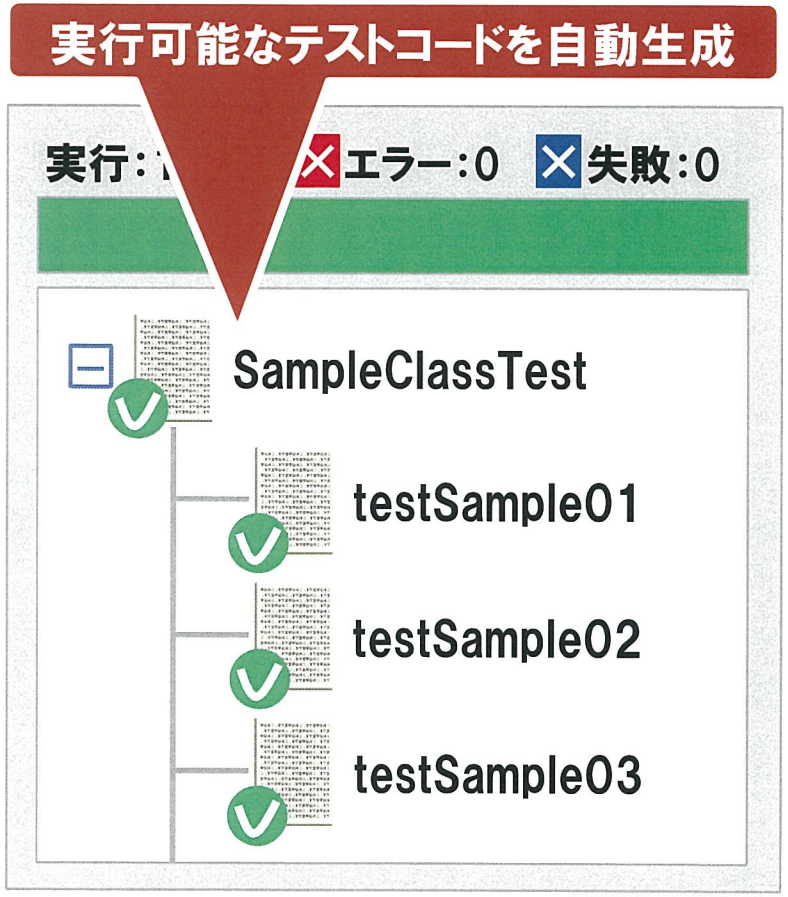
テスト項目表雛形
自動生成

ル	1	2	3	4	...
a=1	Y	Y			
aが正常値でない			Y		
b=11	Y				
bが正常値でない		Y	Y		
○○○					
△△△					
×××					
true	Y				
false		Y	Y		
◇◇◇					

テスト項目表(イメージ)

テストコード
自動生成

実行可能なテストコードを自動生成

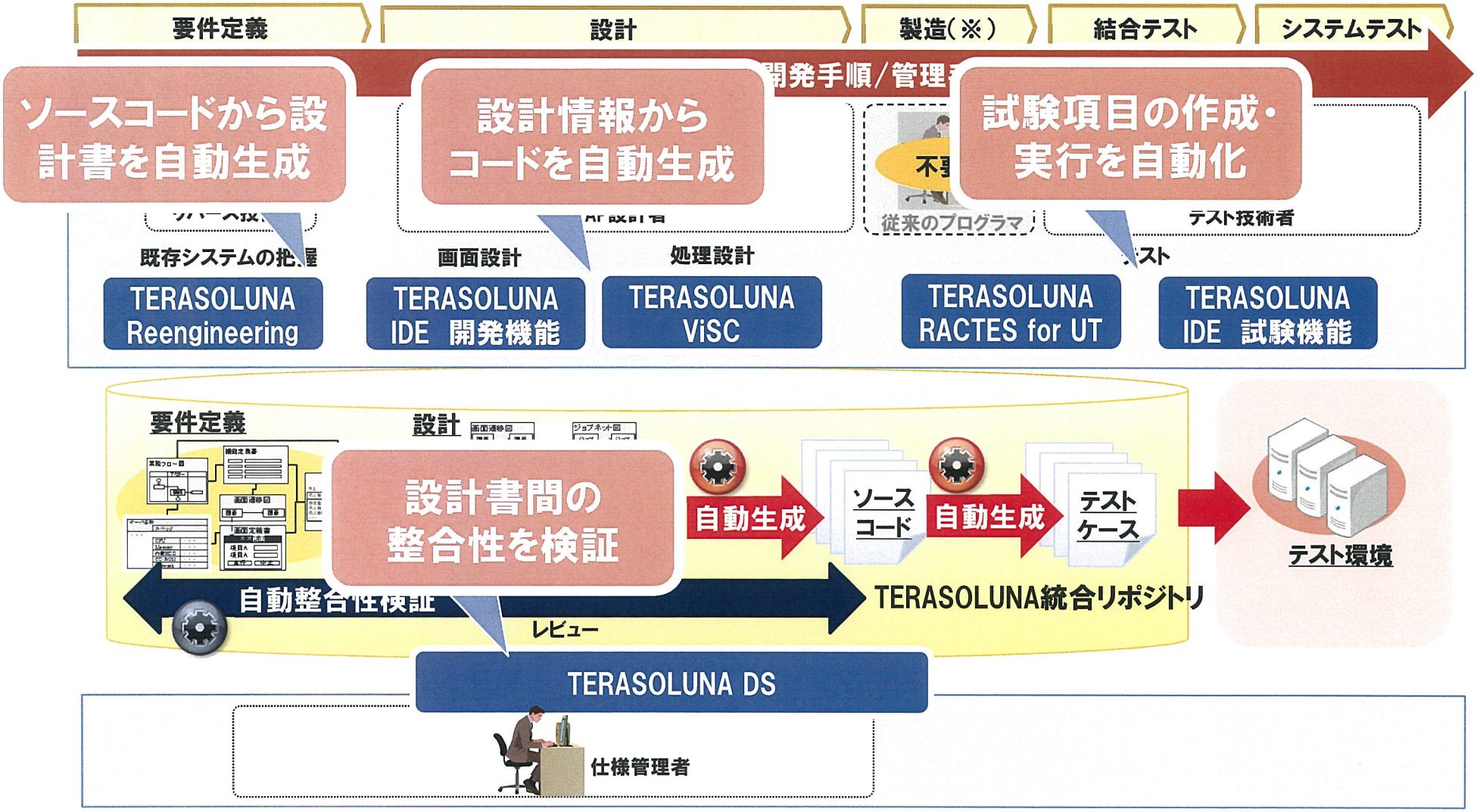


テストコード作成自動化による生産性向上・網羅性向上

自動化の連動　TERASOLUNA Suite

NTT DATA

ツールを連動させることで、システム開発を一層短期化する



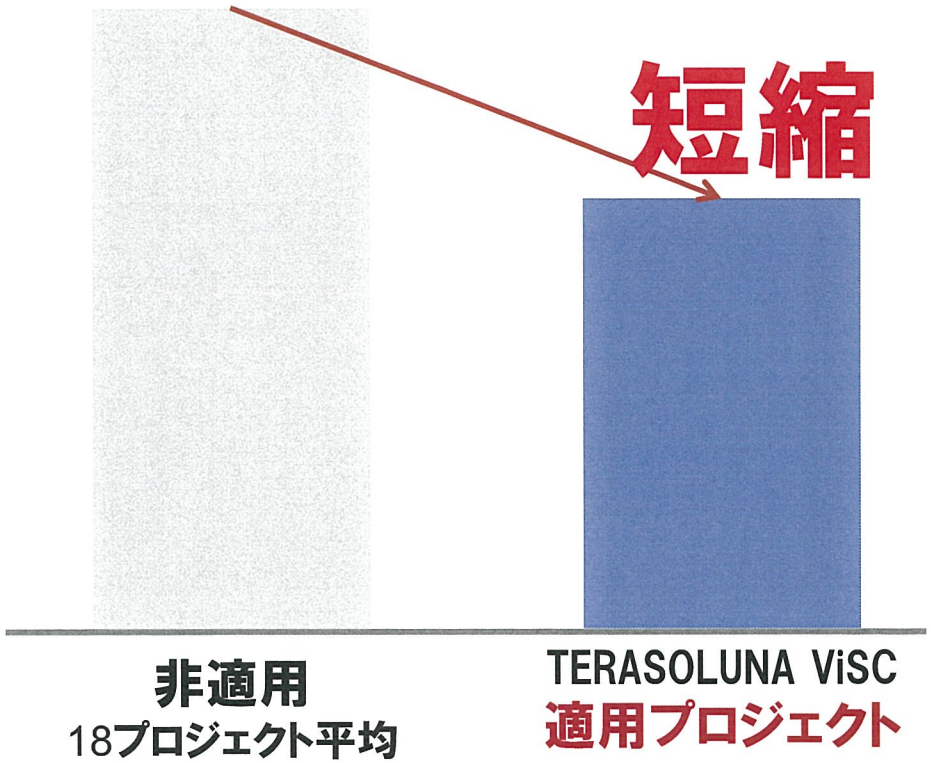
開発自動化の効果

NTT DATA

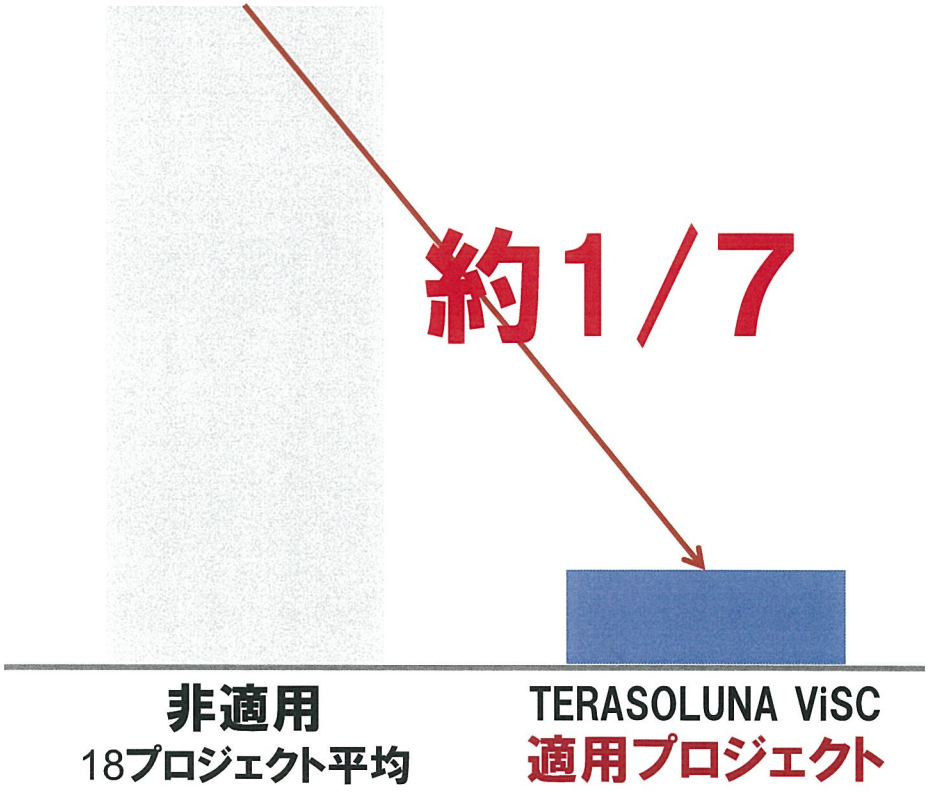
開発自動化ソリューションを用いることにより、「高生産性」と「高品質」を両立したシステム開発を立証

TERASOLUNA ViSC の適用/非適用の比較

工期（設計～結合試験）



受入試験バグ件数



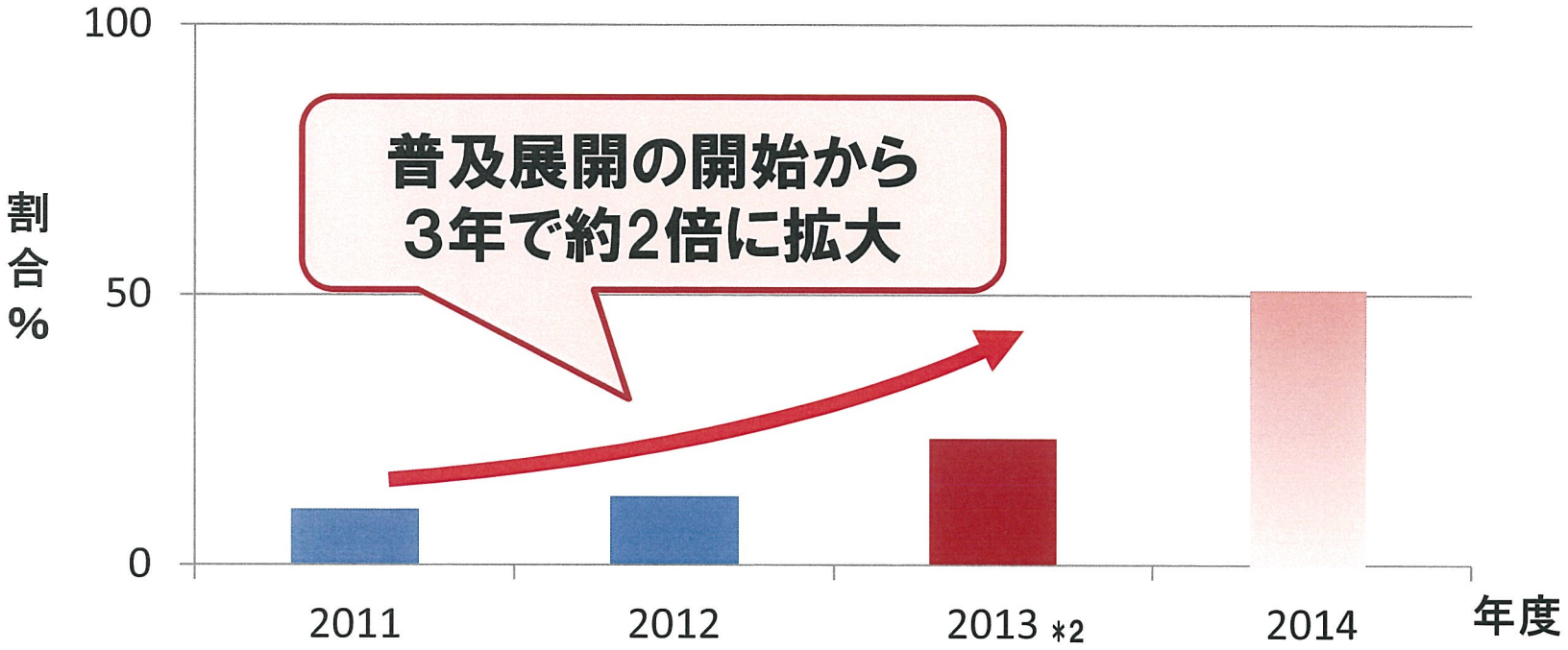
出典）社内調査よりグラフ化

開発自動化の今後の展開

NTT DATA

開発自動化の適用が進む

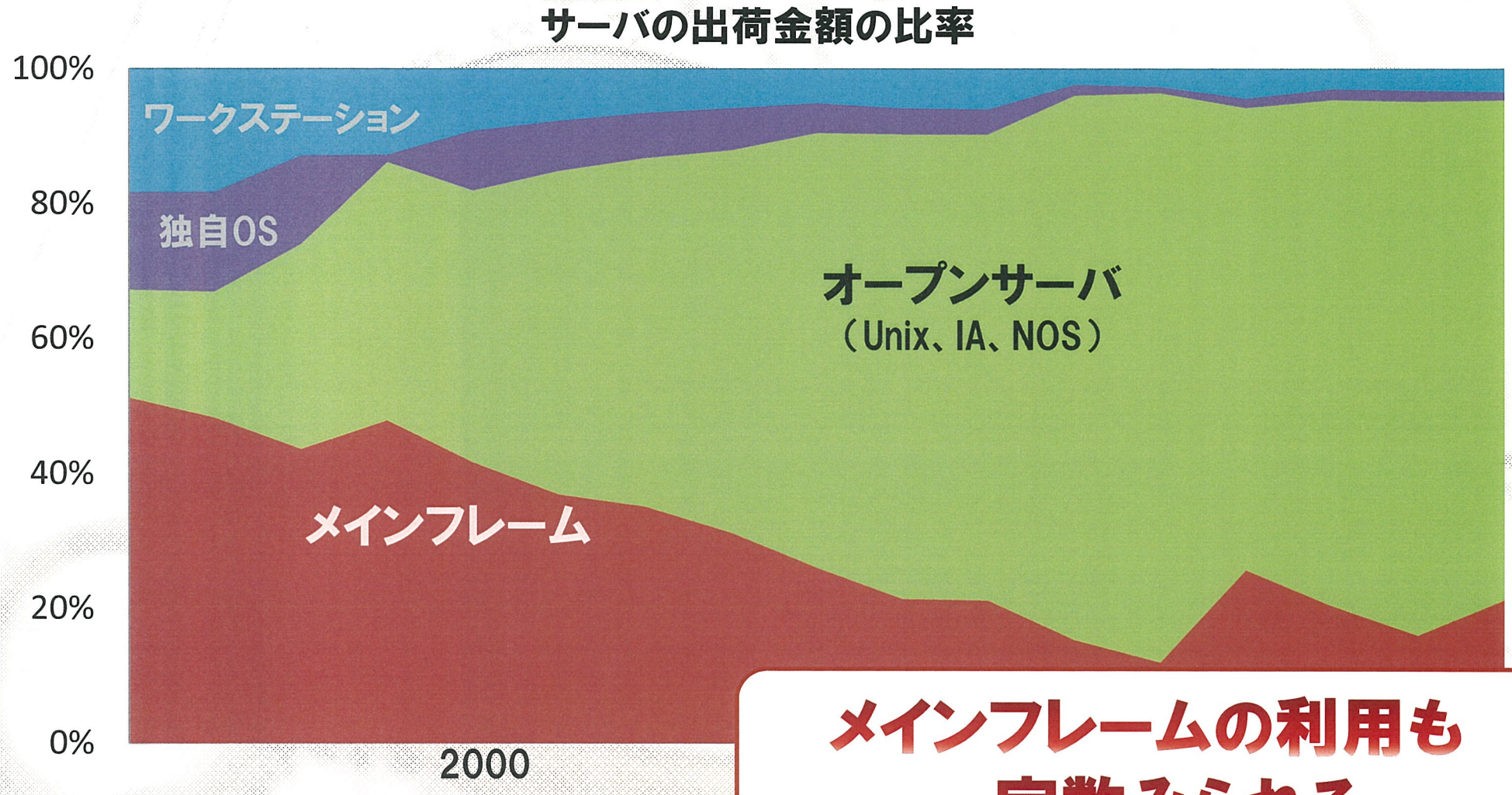
NTTデータの開発案件への開発自動化ツール普及率※1



※1）開発案件のうち、TERASOLUNAが適用可能なJavaの開発案件に対する、TERASOLUNA開発自動化ツールの適用割合を示す
※2）2013年11月末までのデータを元にした、2013年度の予測値
出典）社内調査よりグラフ化


レガシー再生への新たな取り組み

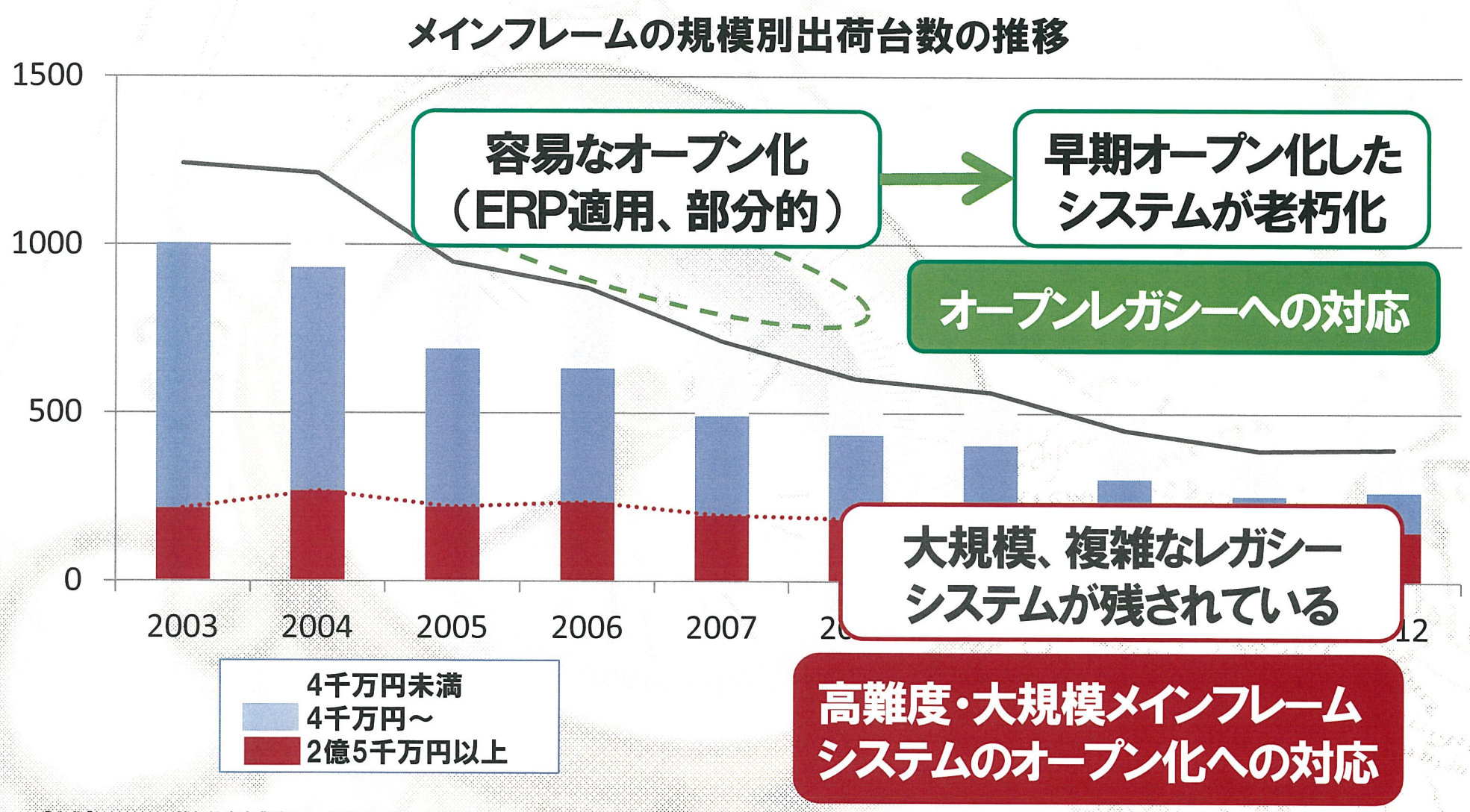
メインフレームからオープンサーバへの市場推移



メインフレームの利用も一定数みられる

【出典】JEITA(電子情報技術産業協会)の集計よりグラフを作成

2つの「レガシーモダナイゼーション」



【出典】JEITA(電子情報技術産業協会)の集計よりグラフを作成

レガシーモダナイゼーションへの期待

NTT DATA

保守が抱える課題から、レガシーモダナイゼーションへの期待が大きい

Before

システムの特徴
- 大規模システム
- 機能追加により仕様が複雑化

問題・課題

- 保守コストがかかる
- 開発要員（COBOL、PL/I）の確保が難しい
- ベンダに依存する
- 俊敏な対応ができない
- 品質が低下している

劇的 レガシーモダナイゼーション の期待

After

保守をしているなら、**仕様を把握**しているはず

仕様は変わっていないから、**短期間**で出来るはず

設計書等の資材を再利用して、**低コスト**で出来るはず

レガシーモダナイゼーションの手法

NTT DATA

レガシーモダナイゼーションは、「リホスト」、「リライト」、「リビルド」の3つに分類される

	アプリケーション	テスト	設計書・有識者	システム最適化	コスト
リホスト	変更無し	現行との比較	不要	×	小
リライト	**単純変換**	**現行との比較**	**要※**	**×**	**小～中**
リビルド	再設計・開発	設計書ベース	要	○	大

※内容によっては、不要のケースもある

リライトは、アプリケーションを単純変換し、メインフレームをオープンに移行する。リビルドに比べて、低コストで実現できると期待される。

レガシーモダナイゼーションの問題

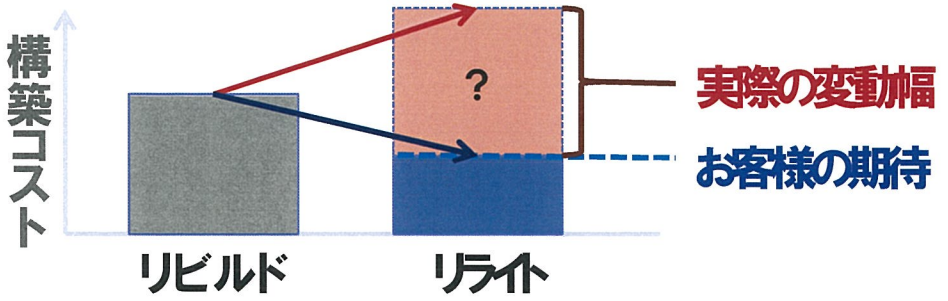
NTT DATA

レガシーモダナイゼーションの状況とお客様の感覚に乖離がある

お客様の感覚

仕様を把握しているはず　　**短期間**で出来るはず　　**低コスト**で出来るはず

大きな乖離



レガシーモダナイゼーション開発の現実

①手順が未確立

再利用と再設計が平行する複雑な開発手順

②アプリケーションだけではない

運用、DB、インテグレーション等

③現状把握が不十分

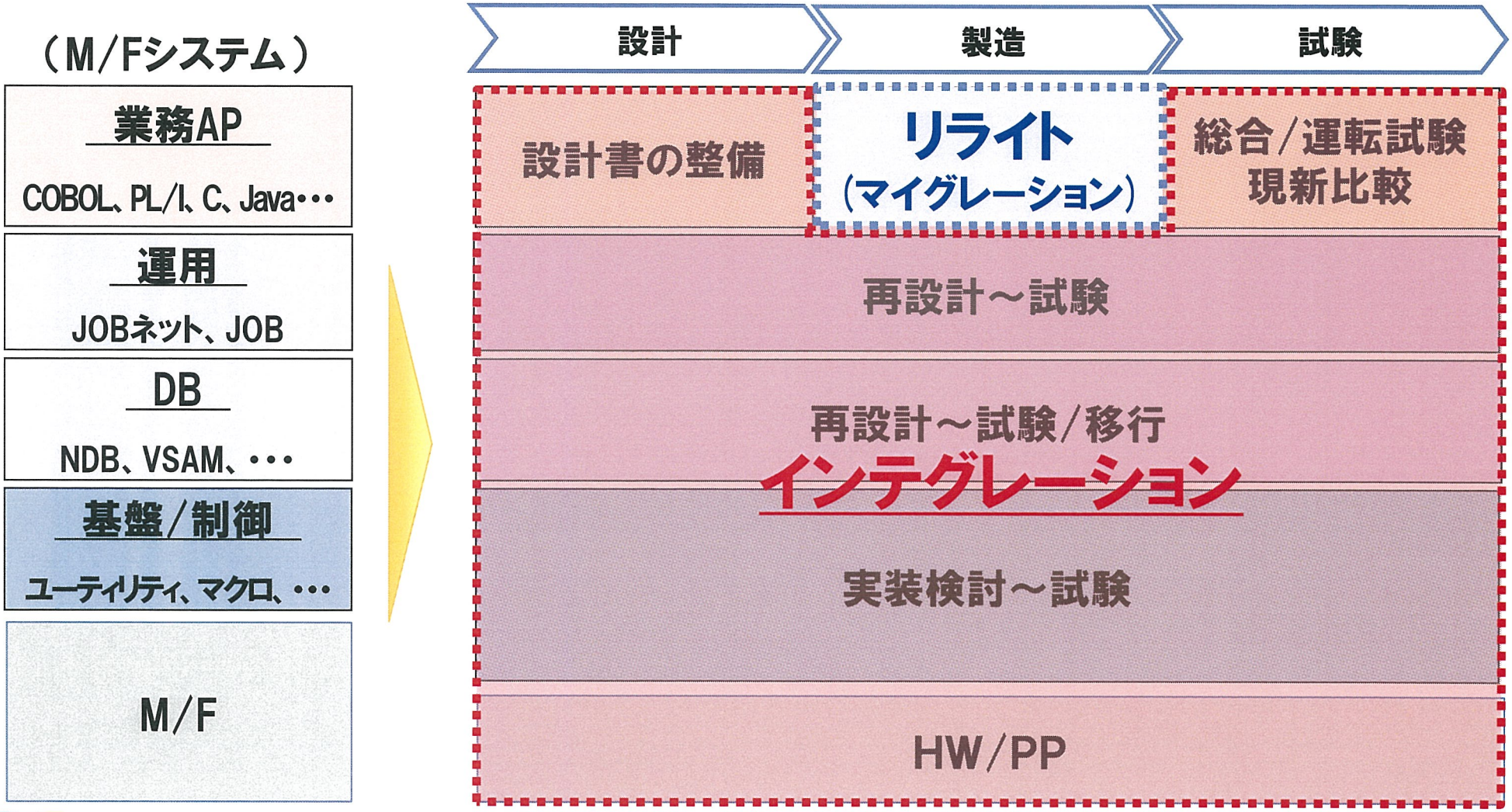
アプリケーションは複雑化
設計書もメンテ不十分

ブラックボックス化し、少数の有識者に依存

対象はアプリケーションだけではない

NTT DATA

比較的簡単に思われる、リライトであっても、多くの領域で再設計～実装～テストの作業が必要



①設計書リカバリー

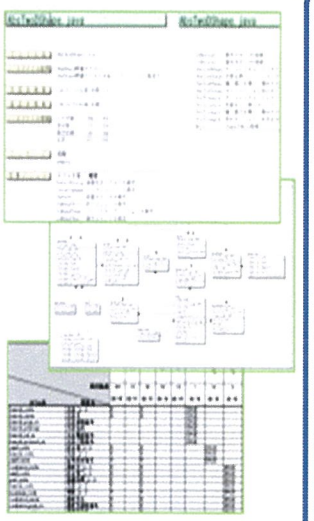
NTT DATA

実働するシステム情報を元にした、開発着手を実現
さらに、ツールの高度化により状況把握の高度化

確実に信頼できる資産

- コード
- 実働システムの
  画面、帳票、
  動作ログ、
  電文キャプチャ
  実データ
- システム設定

TERASOLUNA Reengineering



業務処理ロジック抽出

データ編集仕様抽出

個別業務ルール抽出

「処理」

「機能、仕様」

人手による把握手順

今後の技術開発

ツールの高度化により、
要件レベルまでを自動分析

システムの正しい把握

- 改善要求から要件を定義できる
- テストの設計ができる

「業務」

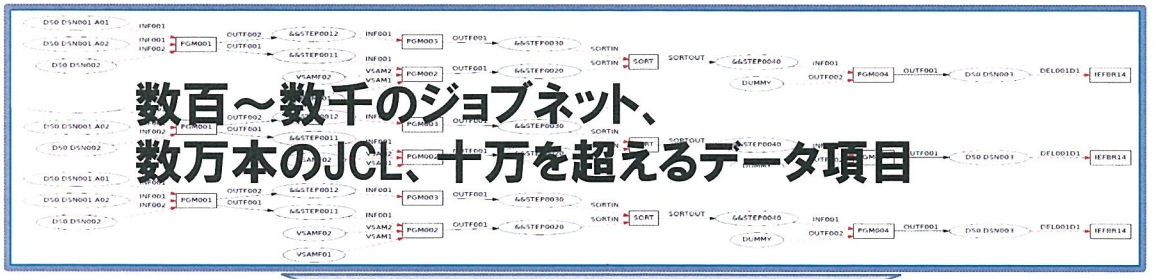
「要件」

②最適化設計

NTT DATA

大規模システムに見られる、バッチ処理、DB、帳票等の、最適化を省力化する仕組みの検証と開発

大規模システムのバッチ処理の例



大規模バッチの改善検証

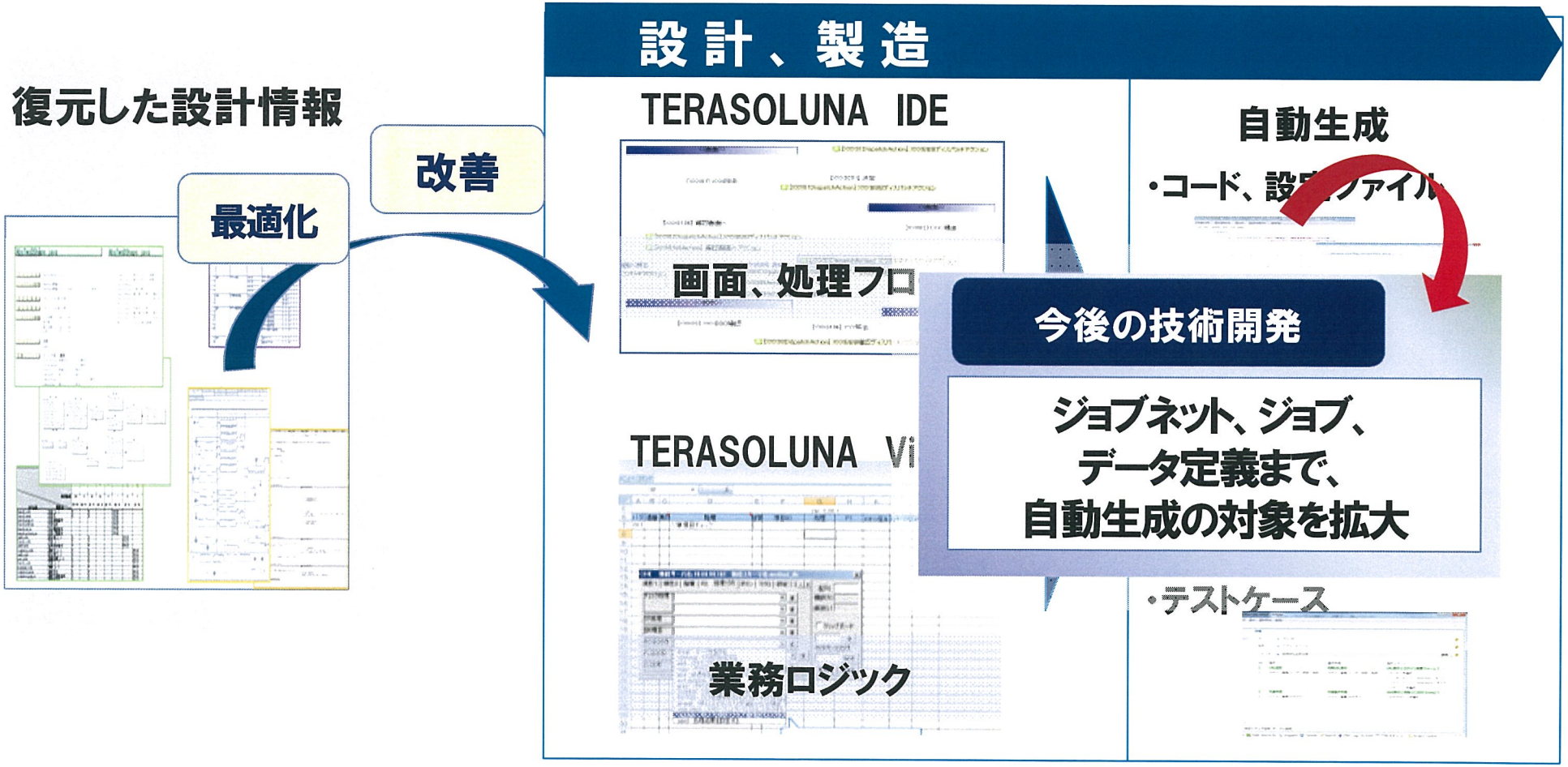
- ファイルベース(バケツリレー)処理からRDB処理への改善
- 重複データや重複処理をしている点の改善
- 並列処理が実現可能な点を改善
- 処理の重いバッチ処理の抽出と改善
  →HadoopやオンメモリDBなどの適用　　等

「可視化」「問題把握」「最適化・チューニング」「検証」の仕組み

③コードの自動生成

NTT DATA

TERASOLUNA IDE、TERASOLUNA ViSC を利用して、新規コードを自動生成することで、製造、テストを短縮



④テスト計画

NTT DATA

リホスト・リライトでの現新比較試験には手順・データの準備が必要だが、リビルドの場合は通常開発におけるテストと同様

リホスト、リライトのテスト

現行システムの成果物と新システムの成果物を比較し同じであることの確認が必要

通常のテストに加え、**現新比較試験**が追加

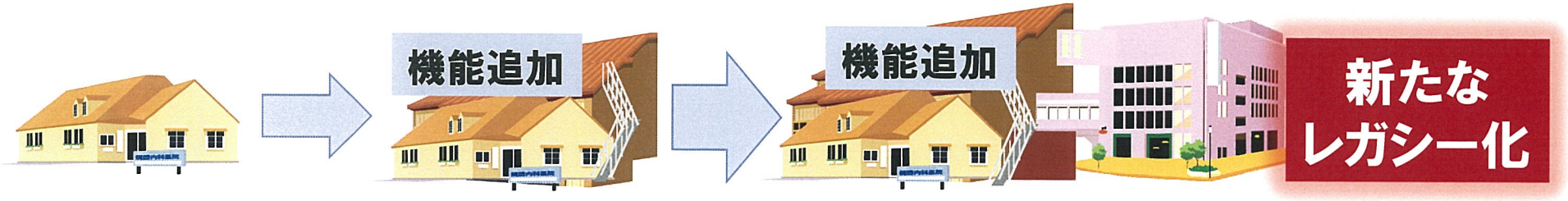
- **データ準備が大変**
  実際の業務データを使用するため、お客様の負担が大きい
- **手戻りの要因**
  問題が生じた際、業務を把握していないと、問題の切り分けができずに手戻りとなる

⑤モニタリング

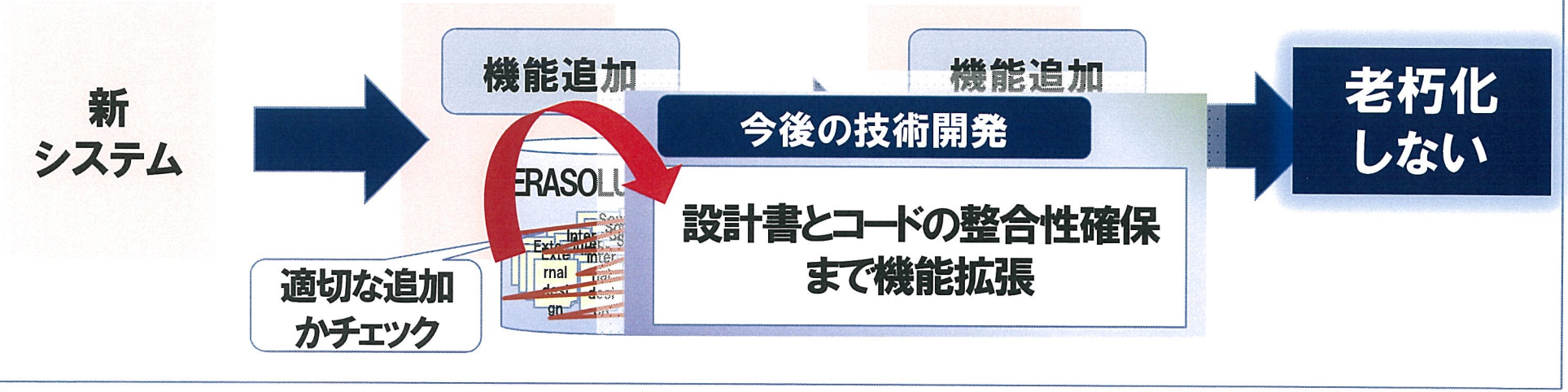
NTT DATA

開発後の保守開発時への TERASOLUNA DS の適用で、適切な進化をチェックし、新たなレガシー化を防止

開発後に何もせず機能追加をくりかえすと・・・



TERASOLUNA DS によりモニタリング

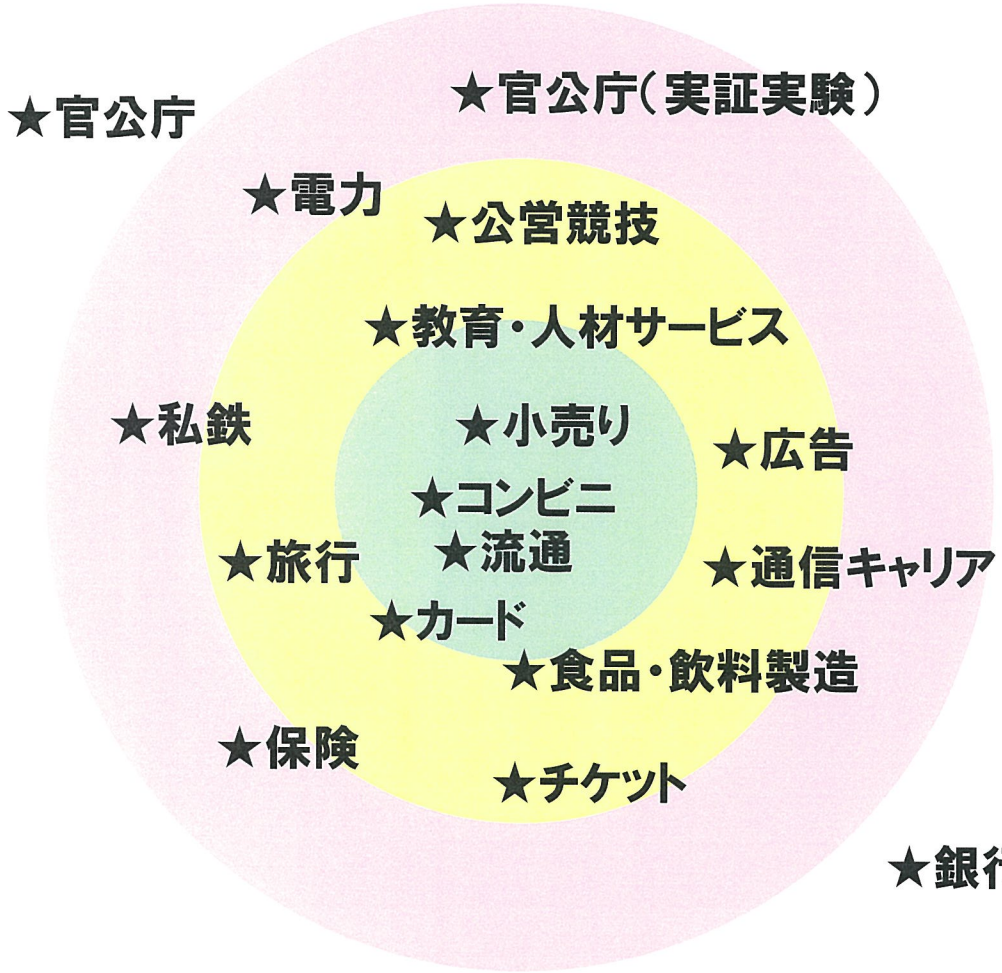



価値積み上げ型の開発

アジャイル開発に注目している業界

NTT DATA

コンシューマ／インターネット事業を中心に、経験価値を重視するサービスで適用が急速に広がっている



：既に適用中の領域

：今後適用拡大する領域

：適用可能性がある領域

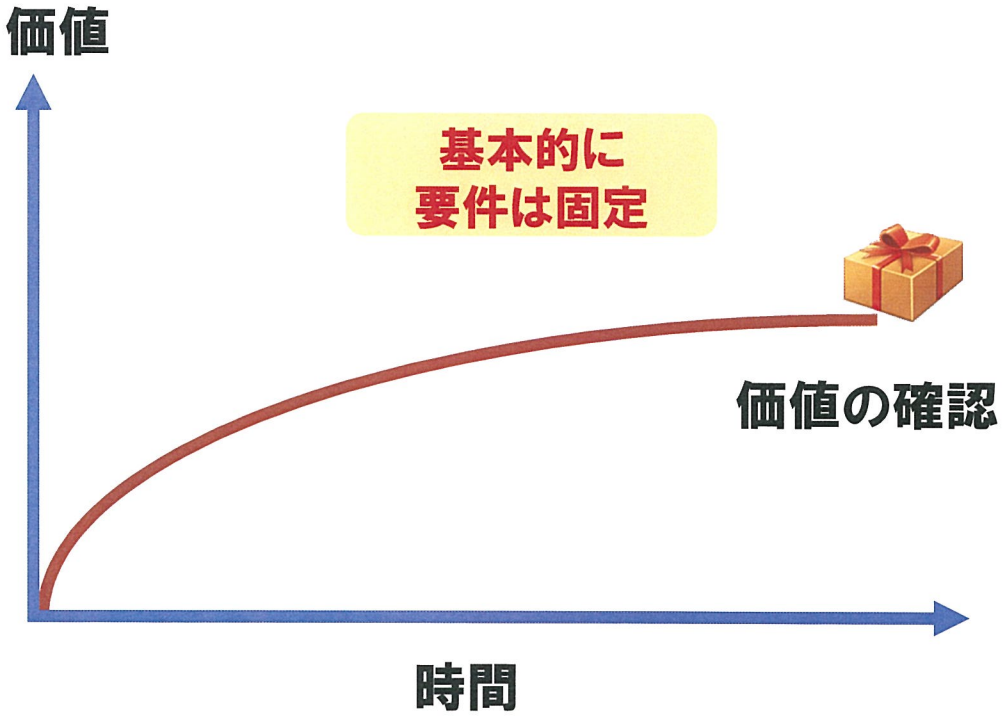
：適用可能性が低い領域

ビジネス価値の考え方のシフト

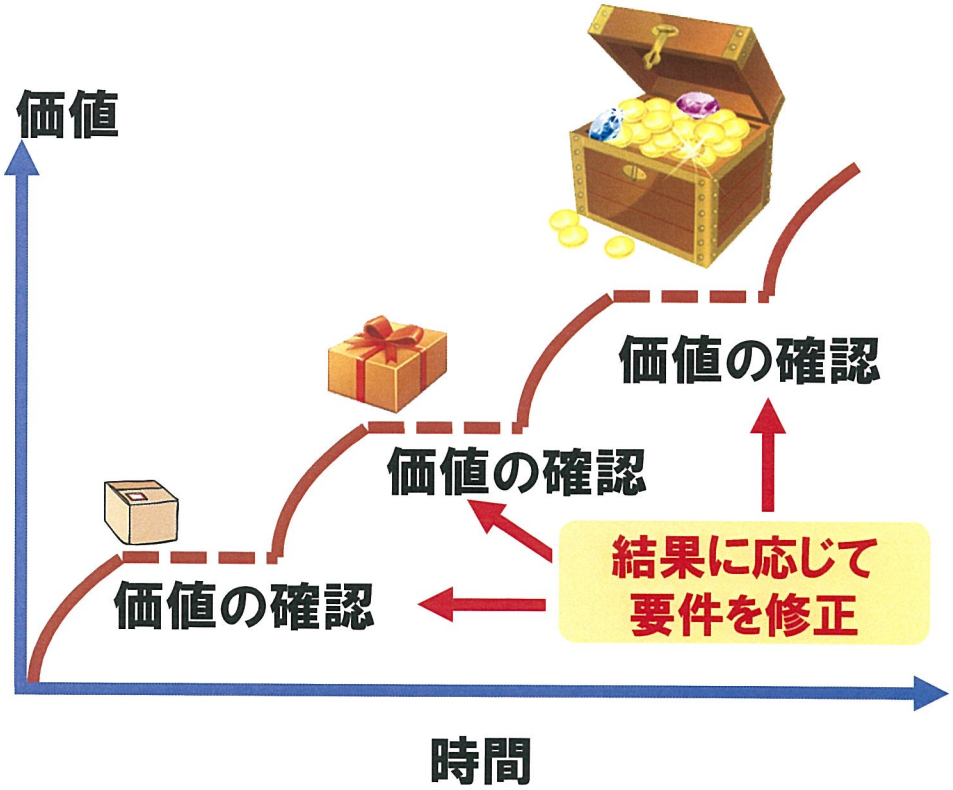
NTT DATA

「**最小限の開発**」と「**価値の確認・方向修正**」を反復することで、
リスクを最小化しつつ、**価値の最大化**を図る手法

ウォーターフォール開発



アジャイル開発

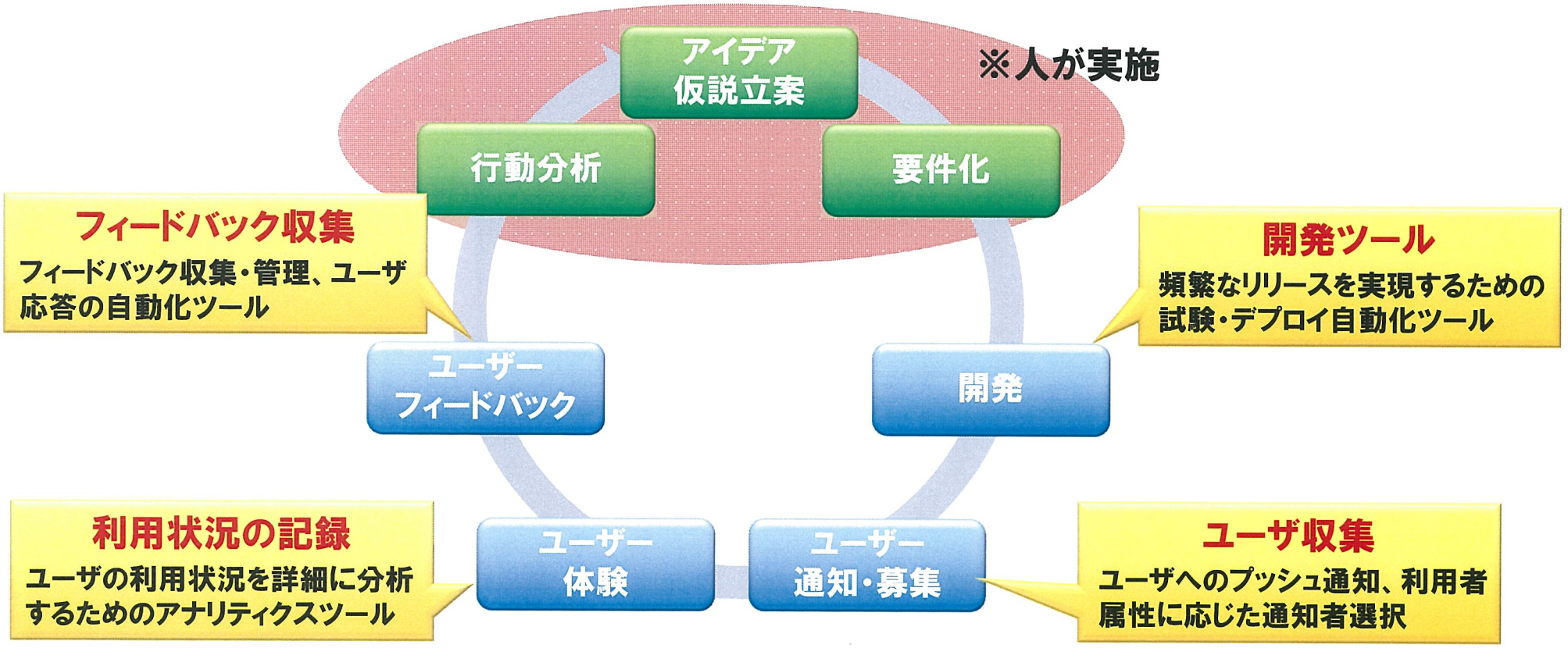


開発サイクルの効率的実行

NTT DATA

アジャイル開発のサイクルを効率的に回すために様々な開発を支援する仕組みが開発されている。

フィードバックライフサイクル



開発環境 Agile Professional Center Cloud

NTT DATA

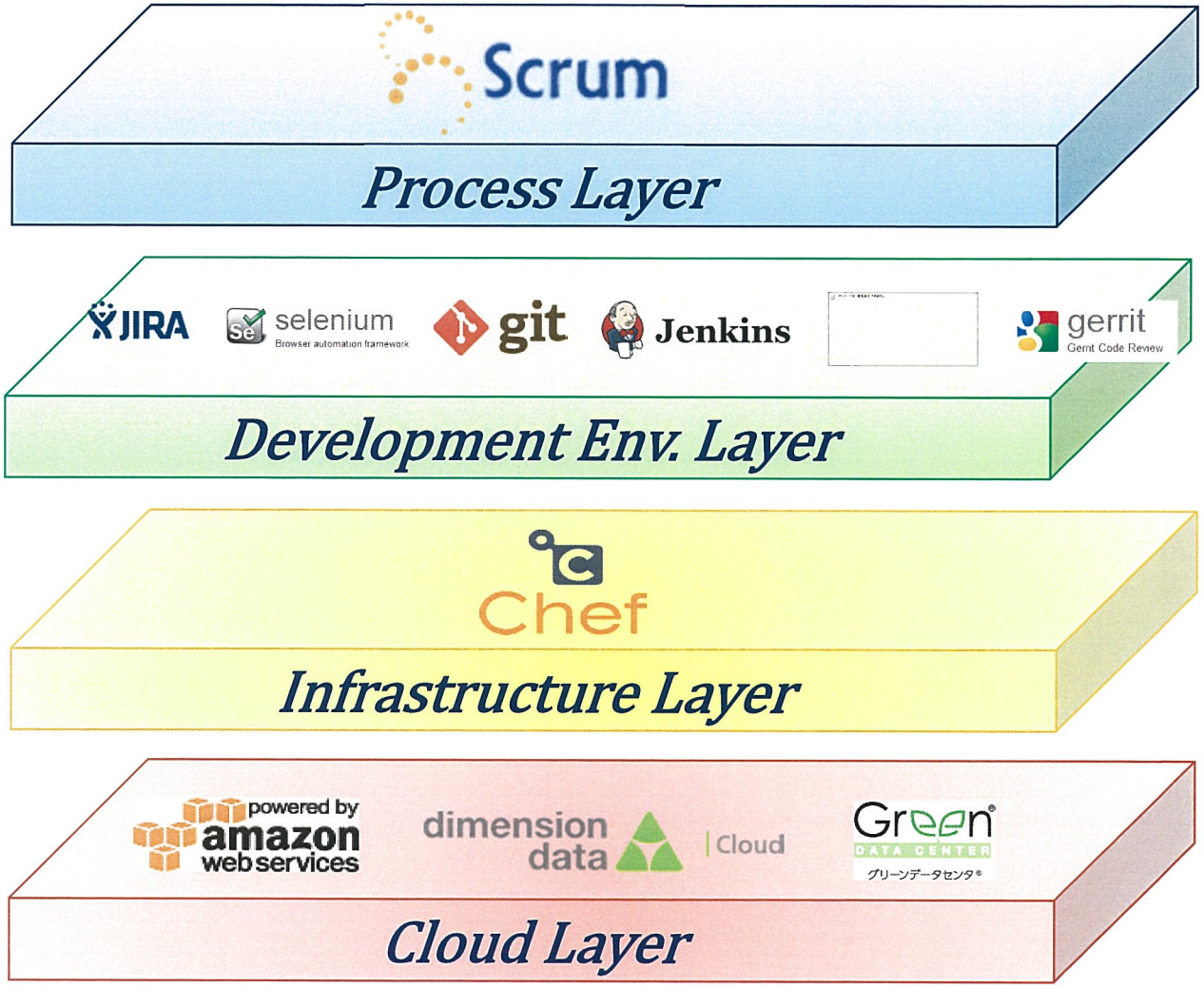
アジャイル開発向け高度開発環境 APC Cloud※ を整備。

Lean Startup
ユーザからのフィードバック
ライフサイクルを構築

高速Scrum開発
開発作業を自動化

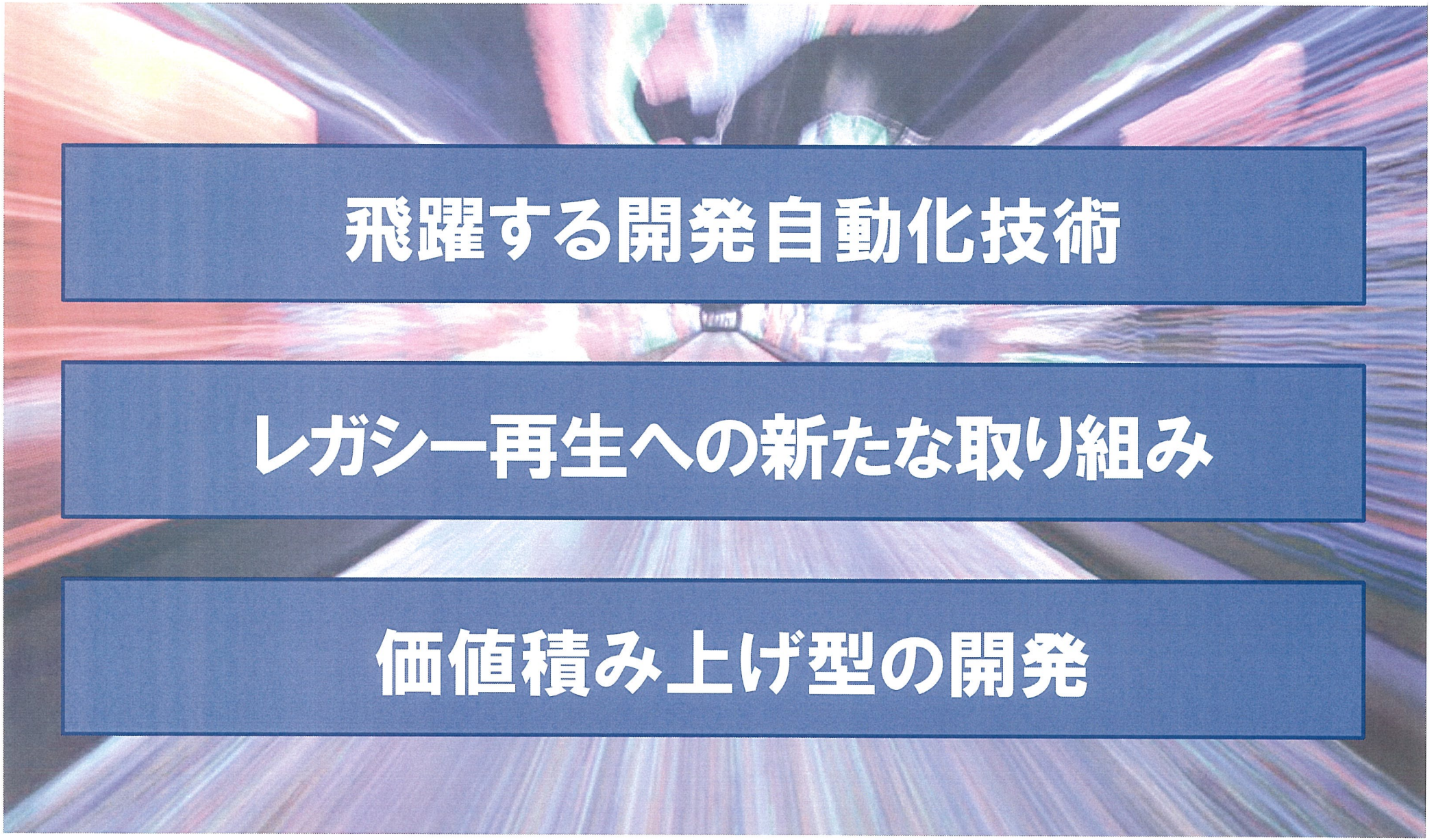
ワンクリックデプロイ
環境構築や
アプリのリリースを自動化

オンデマンドシステム基盤
パブリッククラウドによる
柔軟で拡張性の高基盤



※ APC Cloud : Agile Professional Center Cloud / アジャイル開発向け開発環境

高速開発を実現するソフトウェア開発技術



学会とソフトウェア業界との連携

NTT DATA

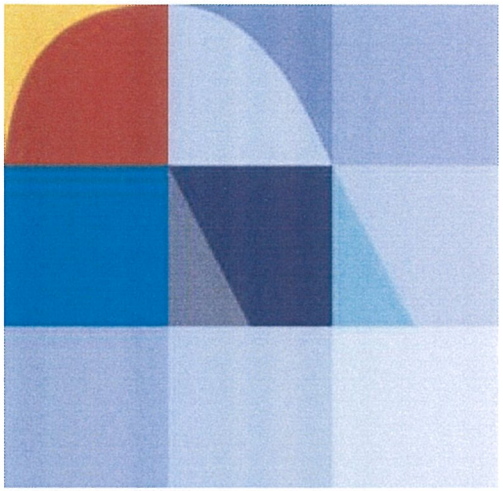
技術研究と実証研究にもとづき、ソフトウェア開発にエンジニアリング要素を導入し、人海戦術体質を改善する

技術研究

人に依存する部分が多く、研究テーマを設定しにくい分野であるが、リバースエンジニアリング、モデルベース開発、形式手法の適用など、大学・研究機関に期待するところは大きい。

実証研究

開発データはお客様関係から公開は困難であるが、ベンチマークデータや事例公開などで、企業はEmpirical Software Engineeringの面で貢献できる。



NTT DATA

Global IT Innovator