

プロジェクト事例に学ぶ！ 陥りやすいレガシーオープン化の落とし穴と対策

2015年4月23日

株式会社ワンサークル

代表取締役 星野 亨



アジェンダ

1. 当社のご紹介
2. レガシーモダナイズの全体像
3. レガシーモダナイズの課題事例
4. 当社の提言

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

2015年2月16日設立、4月1日創業

大手システムインテグレータにて29年間、メインフレームを中心とした基幹システムの開発、オープンシステムへのマイグレーション経験を通じて、レガシーシステムユーザー様が抱える課題を技術面からアドバイスすることを目指して設立しました。

主な事業内容

- ・技術顧問サービス:レガシーシステム将来計画策定のアドバイス
- ・技術支援サービス:レガシーオープン化プロジェクトのサポート
- ・情報発信活動:レガシーシステムの最新化を正しく理解していただくための情報発信、啓蒙活動

レガシーシステムのレイヤと特徴

| 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 | <ul style="list-style-type: none">・ユーザーの資産である・移行前後での等価性が求められる・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 | <ul style="list-style-type: none">・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)・レガシー特有の機能が含まれている・機能互換が保証されているオープン製品は無い・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 |
| コンテキスト層 | 業務システムが稼働するために必要な技術領域 | <ul style="list-style-type: none">・主に製品ベンダ提供の基盤機能で実現されている・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・現行システム資産調査では把握できない・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、価値観が要件に反映されている・過去の経緯は散逸しているケースが多い |

レガシーモダナイズの課題要素

| 大分類 | 中分類 | 技術領域 | 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 | 業務データの格納先、出力先媒体 | DB、帳票、画面、テープ |
| | データ | 業務データ実体、符号化方式 | EBCDIC、2バイトコード |
| 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 |
| | 処理方式 | 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 |
| | 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関連、メンテナンス関連 |
| コンテキスト層 | 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 |
| | 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 | |

課題事例のご紹介(ケースA)

| 大分類 | 中分類 | 企画・計画 | 設計・変換 | 照合試験 | システム試験 |
|---------|-----------|-------------------------|---------------------|----------------------|------------|
| ユーザー層 | プログラム | 移行資産確定 | コンパイラの厳密性 テスト網羅性 | 照合試験シナリオ | |
| | 入出力 | | 帳票フォーマット | 画面フォーマット 2バイト文字対応 | |
| | データ | | | データ照合条件・照合単位 | 商用切り替えシナリオ |
| メカニズム層 | API | | 製品品質 | 全帳票出力確認 | |
| | 処理方式 | | オンバッチ連携処理 | | |
| | 外部連携 | | ファイル連携製品 | | 他システム連携試験 |
| | ユーティリティ | | | | |
| コンテキスト層 | 性能・スケーラブル | Open to Openのため、大きな変更なし | | | |
| | 可用性 | | | | |
| | 運用 | | | | 運用部門ツール対応 |
| | 保守 | | | | 保守運用、契約条件 |
| | セキュリティ | | | | |

課題事例のご紹介(ケースB)

| 大分類 | 中分類 | 企画・計画 | 設計・変換 | 照合試験 | システム試験 |
|---------|-----------|--------|---------------------------|-------------------|--------|
| ユーザー層 | プログラム | 移行資産確定 | 4GL・アセンブラ変換設計 | 照合試験シナリオ テスト手順 | 進行中 |
| | 入出力 | | DB非互換、帳票、外字(印字)、 端末管理 | | |
| | データ | | 文字コード変換 移行の文字コード整合性保証 | データ照合条件・ 照合単位 | |
| メカニズム層 | API | | 大量の機能追加 | | |
| | 処理方式 | | サブシステム間連携 バッチジョブスケジューラ | | |
| | 外部連携 | | 接続先毎に異なる通信手段 | | |
| | ユーティリティ | | JCLからの膨大な全数調査 | | |
| コンテキスト層 | 性能・スケーラブル | | サブシステムにまたがるトランザクションのスケール化 | | |
| | 可用性 | | 特定サーバの高可用性 | | |
| | 運用 | | TSS画面を使用した運用 | | |
| | 保守 | | アプリ開発、デプロイツール | | |
| | セキュリティ | | 端末認証、アクセス認可 | | |

課題事例のご紹介(ケースC)

| 大分類 | 中分類 | 企画・計画 | 設計・変換 | 照合試験 | システム試験 |
|---------|-----------|---------------------------|---|--------------|--------|
| ユーザー層 | プログラム | 移行資産確定 | アセンブラ解読 コンパイラ厳密性 | 照合試験シナリオ | |
| | 入出力 | 正規化 | COPY句vsエンティティ対応 | | |
| | データ | | マルチレイアウト 繰り返し項目 COBOL独自属性 初期値(LowValue,null) | データ照合条件・照合単位 | 試験網羅度 |
| メカニズム層 | API | | 階層型DBのAPI化 COBOL項目編集機能 | | |
| | 処理方式 | | JCL・バッチジョブスケ ジューラ互換設計 | | |
| | 外部連携 | | | | |
| | ユーティリティ | | JCLからの全数調査 | | |
| コンテキスト層 | 性能・スケーラブル | | 資源排他区間の違い | | |
| | 可用性 | 共通インフラ化の方針により、レガシー実現方式の排除 | | | |
| | 運用 | | | | |
| | 保守 | | | | |
| | セキュリティ | | | | |

課題発生特性マトリックス

| 大分類 | 中分類 | 企画・計画 | 設計・変換 | 照合試験 | システム試験 |
|---------|-----------|--------|----------------------|--------------|--------|
| ユーザー層 | プログラム | 移行資産確定 | オープン化方針に依存して難易度の変化 | 照合試験シナリオ | |
| | 入出力データ | | | データ照合条件・照合単位 | |
| メカニズム層 | API | | システム毎に発生傾向が異なる | | |
| | 処理方式 | | | | |
| | 外部連携 | | | | |
| | ユーティリティ | | | | |
| コンテキスト層 | 性能・スケーラブル | | レガシーの実現手段にこだわると深刻化する | | |
| | 可用性 | | | | |
| | 運用 | | | | |
| | 保守 | | | | |
| | セキュリティ | | | | |

共通の課題

個別の課題

ソリューションカバレッジマップ

| 大分類 | 中分類 | 変換サービス | リHOST製品 | マイグレーションサービス | システムインテグレーター |
|---------|-----------|--------------|-------------------|-------------------|------------------------------|
| ユーザー層 | プログラム | ◎(多彩な変換パターン) | ○ | ◎(独自方法論+経験値) | ユーザーニーズに合わせたソリューションの選択と組み合わせ |
| | 入出力 | △ | ○(製品毎に異なる) | | |
| | データ | △ | ○ | | |
| メカニズム層 | API | △(一部) | ◎(機能不足時エンハンス等で検討) | ○(適材適所でソリューション選択) | |
| | 処理方式 | N/A | ○ | ○(適材適所でソリューション選択) | |
| | 外部連携 | N/A | △(一部製品のみ) | △(一部) | |
| | ユーティリティ | N/A | △(一部製品のみ) | △(一部) | |
| コンテキスト層 | 性能・スケーラブル | N/A | N/A | ユーザーサポート | ユーザー要件に応じたシステムインテグレーション |
| | 可用性 | | | | |
| | 運用 | | | | |
| | 保守 | | | | |
| | セキュリティ | | | | |

中核となるソリューション

当社の提言

| 大分類 | 中分類 | 提言内容 |
|---------|-----------|---|
| ユーザー層 | プログラム | 言語変換率は実現性の評価尺度にならない 企画時点で移行対象資産の確定を行う 照合試験シナリオ、テスト準備に関して業務開発部門との作業合意が必要 |
| | 入出力 | プログラムが処理できる形式への変換の実現性 (言語独自のデータ形式、構造定義は要注意) |
| | データ | 照合試験でのカバレッジ目標、許容レベルを企画時点で決定する |
| メカニズム層 | API | ソース、JCL等の全調査で全ての機能を洗い出す |
| | 処理方式 | 資産調査だけでは全ての把握は不可能 |
| | 外部連携 | 現行有識者へのヒアリング、業務観点からの調査 |
| | ユーティリティ | 利用用途の洗い出しと移行選別 |
| コンテキスト層 | 性能・スケーラブル | レガシーとオープン是实现方式が異なるため、現行システムの実現手段に固執(通称「現行踏襲」)せず、SLAの再定義とオープン製品の代用とする |
| | 可用性 | |
| | 運用 | 現行システムの理解と新システムでの方針を企画時点で確定する。 |
| | 保守 | |
| | セキュリティ | |

経験、実績が有効・業務ナレッジが必須

レガシーナレッジが必須

トップダウンの大胆な割切りが有効

当社の取り組み

「レガシー問題のセカンドオピニオン」としてレガシーシステム特有の課題と将来構想に応じたアドバイスをを行います。

網羅的・中立的なアドバイス

- システム目線による網羅性
- ベンダー中立の立場

業界ソリューションとの協力
関係

- 実力ベンダー様との協力関係
- 最適なソリューションのご紹介

メインフレームへの理解

- ユーザーITシステムへの理解
- コンテキストに至る現状把握

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

