

# CPU 置換ソフトウェア開発における マイグレーション統合サービス

池田 祐一 殿岡 直哉

株式会社OKIアイディエス（以下、OIDS）は、主に映像・音声・通信機器や産業用機器などの開発を行う設計会社である。弊社では、過去にも現行製品に対して製造中止となったデバイスを新デバイスに置換える開発を受託してきたが、デバイス置換え後の評価工程において量産品質を確保するために多大な時間とコストを費やすことが多かった。

近年ではデバイス製造中止に伴う置換え対応の需要が更に増えてきており、開発期間短縮と量産品質確保を両立させる開発手法の確立が必要であった。

## デバイス置換え開発とは

デバイス置換え開発とは、現行量産製品の基板上で動作するCPUの製造中止に伴うCPU置換え開発のことである。

デバイス置換え開発では、現行量産製品との機能面・性能面での互換性が求められるが、システムが老朽化していることと同時に当時の設計資料が存在しないケースが多くみられる。その場合、機能・性能要件が不透明な状態で開発をスタートさせることになり、デバイス置換え後の現行量産製品との等価性検証に多大な時間を要するという課題を抱えていた。

## OIDS 独自プロセスの確立

過去のデバイス置換え開発の経験を振り返り、設計資産が残されていない老朽化したシステムを、次世代デバイスに効率的に置換える弊社独自のソフトウェア開発プロセスを構築した。本開発プロセスにより、現行システムと同等の機能・性能を維持したシステムへの再構築が可能となった。この弊社独自の開発プロセス構築については、Webセミナーで社外へアピールし、ソフトウェア案件受託拡大に繋げている。

デバイス置換え開発時に顕在化が予想される課題・リスクの回避/低減方法を適用したソフトウェアにおける「デバイス置換え開発プロセス」について、以下に紹介する。

## リバースエンジニアリングが求められる理由

現代のビジネス環境では、テクノロジーの持続的な進化に沿ってシステムの更新が必須となっている。既存のソフトウェアやシステムは、時にはその機能性や効率性を保つため、また時には新しいビジネスの要求を満たすために、常に進化し続けなければならない。未来を見据えた置換え開発は、ビジネスを次のレベルへと導く鍵となる。ここで、「リバースエンジニアリング」はその前工程における非常に重要な役割を果たす。

弊社では、「リバースエンジニアリング」を通じて既存ソフトウェアやシステムの深い理解と詳細な機能解析を開発の初期段階で実施することとした。このプロセスにより、現存するテクノロジーの持つ価値を完全に引き出し、将来の開発計画の礎となる洞察を提供する。置換え開発の成功には、既存資産のポテンシャルと限界を知り、これを新しいソリューションにどのように統合・活用するかが鍵となる。

市場動向に目を向ければ、ソフトウェアのライフサイクルが短くなり、テクノロジー選択が多様化する中で、組織はより迅速かつ柔軟にシステム更新を行う必要に迫られている。弊社の「リバースエンジニアリングサービス」は、そのような動的な市場環境において、クライアントが競争優位性を維持し、技術的障壁を克服できるよう支援する。

弊社の専門知識は、過去のシステムの役割と価値を明らかにし、次世代へのスムーズな移行の道を開くことにあつた。ソフトウェアアーキテクチャーの深い理解、機能の洗い出し、互換性の評価、そして最適な更新戦略の策定に至るまで、初期段階での「リバースエンジニアリング」は、賢明な投資判断と持続可能なシステム開発のための強固な土台を築くことができる。

## デバイス置換え開発の課題

ここで、デバイス置換え開発における課題を4つ述べる。

### (1) 流用元の設計書がない、現行機的设计者が不在

既存製品の設計書がない、または電子形式で保存されていないケースが多々ある。さらに、現行機的设计者が退

職していたり、プロジェクトから異動していたりする場合、技術的な知識や設計意図が継承できていないことがある。結果として設計意図の調査などに時間を取られてしまい、デバイス置換え作業が大幅に遅延するリスクがある。このような状況では、置換え対策後の主要装置の量産継続や市場投入が困難となる。

(2) 流用元の設計が古く、新デバイスにソースコードをそのまま流用できない

既存のソフトウェア設計が古過ぎることで、新しいデバイスにそのまま流用できないケースが多々ある。

この場合、ソースコードの大規模なリファクタリングや再設計が必要となり、開発コストや開発時間が増加する。

また、互換性の問題が発生し、動作確認やデバッグに多大な労力を要す。その結果、製品の機能・性能が現行機と同等以上であることの保証が困難となる。

(3) 既存システムのドキュメント不足と理解不足

既存システムに関するドキュメントの記載内容が不十分で、システム全体の構造や動作が十分に把握しきれないケースが多々ある。そのため、新しいデバイスへの置換えが困難となり、テスト及び検証プロセスを実施する際に多くの時間が必要となる。

(4) 重複コードの整理やソースコードのスリム化

長年にわたりメンテナンスや改修を繰り返してきたソースコードは、重複や冗長な部分が多くなる傾向がある。

これにより、コードの可読性が低下し、バグやメンテナンス難易度が増加する。また、新しいデバイスへ移行する際にパフォーマンスの最適化も課題となる。

### 解決手法の確立：「N-Process<sup>TM</sup> \*1)」

従来のデバイス置換え開発プロセスでは、前述の課題に対応するにあたり、プロセスの後半で量産品質確保のための改良開発ステージが必要であった。

従来の開発プロセスを図1に示す。従来はソフトウェア開発手法のVモデル<sup>1)</sup>を参考にV字プロセスと定義し、開発

### <従来の開発プロセス：V-Process>

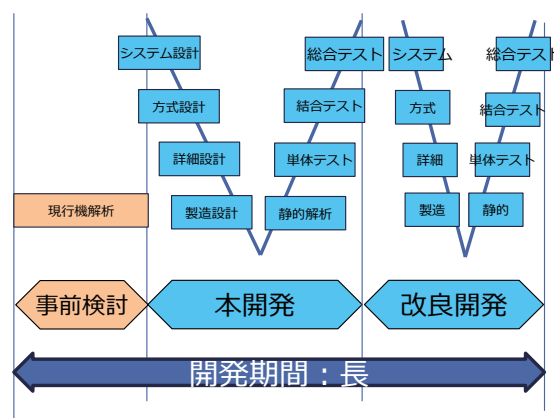


図1 従来の開発プロセス

そこで弊社では独自の新開発プロセス「N-Process」を考案した。「N-Process」では、事前検討段階でリバースエンジニアリングを実施し、開発内容を十分精査する。これにより、従来ではプロセスの後半で実施していた改良開発をせずに、現行機の機能・性能を満足したシステムに置換え開発することが可能となる。

新開発プロセスを図2に示す。

### <新開発プロセス：N-Process>

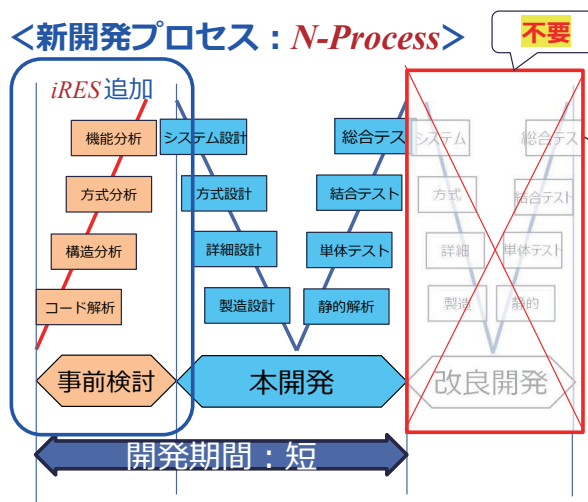


図2 新開発プロセス

図2において、V字プロセス（本開発）の前にリバースエンジニアリング*iRES*<sup>TM</sup> \*2)の工程を追加することにより、デバイス置換え時の開発内容を十分精査することができ、改良開発ステージを省略しても現行機の機能・性能を満足したシステムに置換え開発することが可能となる。

\*1) N-Processは株式会社OKIアイディエスの商標です。 \*2) iRESは株式会社OKIアイディエスの商標です。

## 新開発プロセス「N-Process」詳細

次に、新開発プロセス「N-Process」の詳細を述べる。

### (1) 設計意図を正確に再現:

リバースエンジニアリングによる詳細な解析

「N-Process」では、事前にリバースエンジニアリングを実施し、現行製品の設計意図を詳細に解析する。

リバースエンジニアリング手法のイメージを図3に示す。まず初めに流元元のソフトウェアの素性・特性・評価基準を把握し、開発のベースラインを固める。

これにより、設計書が作成されていない場合や既存設計者が不在の場合でも、元の設計を正確に再現し、開発の遅延を防ぐ。

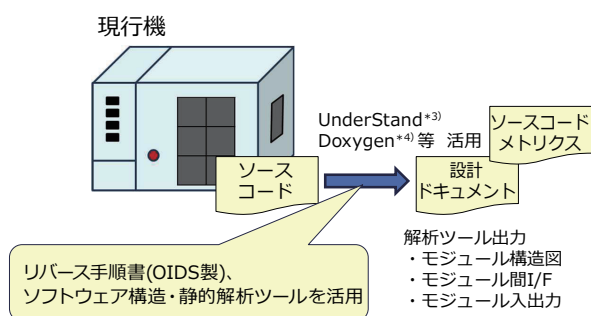


図3 リバースエンジニアリング手法

### (2) スリム化と最適化:

重複コードの整理と互換性調査

リバースエンジニアリングを通じて得られた情報を基に、ソースコード内の重複部分を整理し、スリム化する。

このフェーズの目標は、現行機能を損なうことなく、コードベースの効率と可読性を向上させることである。

等価保証を徹底し、置換え後も機能・性能が現状と同等であることを確認する。

次に、最適な代替デバイスを選定し、新デバイスとの互換性を精密に調査する。現行機と同等の機能・性能を維持することに注力し、等価置換えの保証を行う(図4)。

なお、機能及び性能の向上を目指したい場合は、この等価保証が完了した後の別フェーズで、冗長なコードを削減しつつ、現行機以上の機能・性能を実現するための開発を行う。

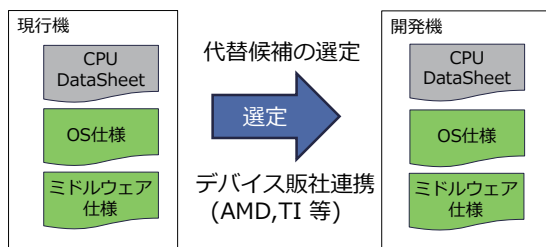


図4 代替デバイスの選定

### (3) 等価証明と評価環境の構築:

現行機と開発機の機能等価検証

実機コードでシミュレーションが可能な評価環境を構築し、現行機と開発機で同じ評価データを使用する(図5)。

この評価環境により、現行機と開発機の機能等価性を徹底して検証し、置換え後も同等の機能・性能を保証する。デバイスの変更に伴う制御方法の調整も行い、現行機と同等の機能・性能を確保する。

このプロセスにより、等価保証の確立を行う。

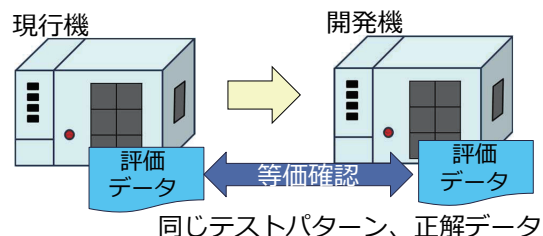


図5 評価データの構築と等価証明

### (4) V字プロセスでの開発推進:

リバースエンジニアリングデータを有効活用

「N-Process」のV字プロセスでは、リバースエンジニアリングで作成した設計資料や評価データを徹底的に分析・精査し、潜在的な課題を事前に洗い出すことで、問題発生を最小限に抑える。また、ここで作成した設計資料や評価データは本開発時にも再利用が可能である。再利用のイメージを図6に示す。

これにより、本開発後の改良開発ステージが不要となり、無駄な工程を排除することで開発期間を大幅に短縮し、お客様の装置を迅速に市場投入できる開発計画を構築する。

### <新開発プロセス : N-Process>

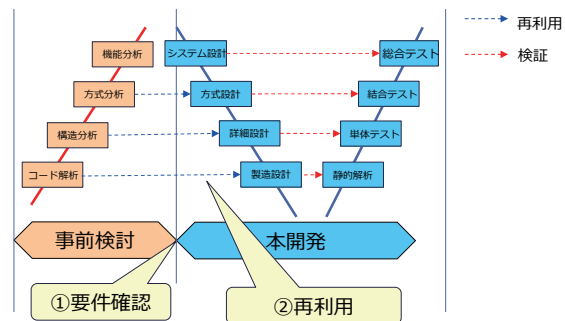


図6 設計資料や評価データの再利用

\*3) Understand: TecMatrix社のソースコード解析ツール <https://www.techmatrix.co.jp/product/understand/index.html>  
 \*4) Doxygen: フリーソフトウェアのソースコード・ドキュメンテーション・ツール <https://sourceforge.net/projects/doxygen/>

## 今後の予定

本開発プロセスは2024年度に構築したものであり、現時点で開発実績を蓄積している段階である。本プロセスを適用した案件をさらに増やして、その結果を評価して本開発プロセスにフィードバックしていくことにより、継続的により良いプロセスに改良していく予定である。◆◆

## 参考文献

1)「Vモデル」『フリー百科事典 ウィキペディア日本語版』2024年8月1日(木) 20:06 UTC

## ●筆者紹介

池田祐一:Yuichi Ikeda. 株式会社OKIアイディエス 事業統括部開発部

殿岡直哉:Naoya Tonooka. 株式会社OKIアイディエス 事業統括部開発部

## TiPo【基本用語解説】

### N-Process

OKIアイディエス独自のデバイス置換開発に特化したソフトウェア開発プロセスの名称を示す。

### iRES (intelligent Reverse Engineering Service)

上記N-Processの事前検討を示す。

### V-Process

ソフトウェア開発手法であるV字モデルを示す。