

データ抽象化による既存ソフトウェアのリバース手法

7N-1

秋庭 真一* 戸田 淳子* 山川 敦夫** 堀内 一***
 * (株) 日立製作所 ビジネスシステム開発センタ
 ** (株) オージス総研 *** 東京国際大学

1. はじめに

リエンジニアリングは、深刻なソフトウェア保守問題の解決あるいはダウンサイジングなどの新しい情報システムへ対応させるためのシステム再構築方法論として注目されている技術である。その中でも既存プログラム群からデータ要素やプロセス要素を抽出/分析して、再利用可能なものに整理する技術としてリバースエンジニアリングが注目されている。

本稿では、データ中心アプローチやリポジトリなどの要素技術を取り入れたリエンジニアリング(DORE: Data Oriented Re-Engineering)の方法論と実際にその方法論に従って実施したリバースエンジニアリングの適用結果について述べる。

2. リエンジニアリングの体系

リエンジニアリングの概念は、図1に示すように大きく3つの階層に分けられる。

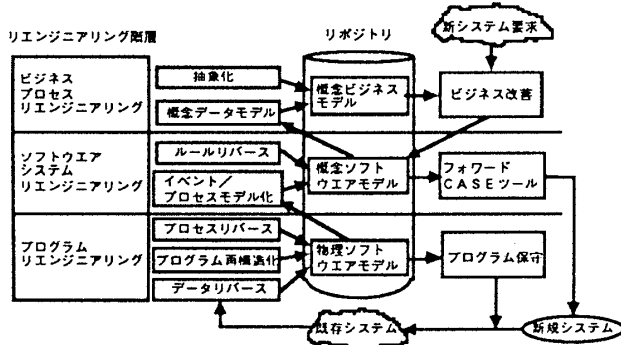


図1 リエンジニアリングの概念

(1) プログラムリエンジニアリング

サブゲッティ状態の個々のプログラムを解析し、その中に存在するデータ要素やプロセス要素を整理し、簡潔で重複コードを持たないプログラムに作り変える技術。

(2) ソフトウェアシステムリエンジニアリング

複数プログラムから構成されるソフトウェアシステムを対象に、プログラム間に埋め込まれたプロセスの重複を排除しながら、抽象化したデータに対する基本的な操作を整理する技術。

(3) ビジネスプロセスリエンジニアリング

単なる業務改善やシステムの見直しの技術ではなく、それを支えるソフトウェアシステムの実態を踏まえた改善や変更を可能とする技術。

3. DOREの概要

DOREでは、図2に示すように既存のイベント対応プログラム群から業務データを抽出し、プログラムの処理論理を業務データ毎にカプセル化し、オブジェクトを構成する。プログラムの論理は、このオブジェクトの相互関連に置き換えられ、抽出されたオブジェクトは、リポジトリに登録され、新規プログラム開発等に利用される。

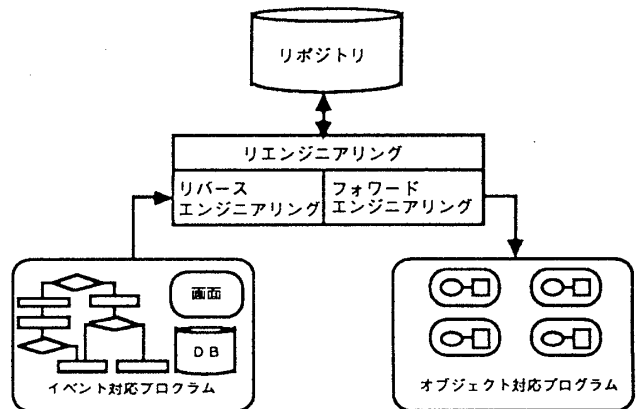


図2 DOREの概念

4. DOREにおけるリバースエンジニアリング

DOREにおけるリバースエンジニアリングの過程は、以下に示す2つで構成される。

(1) データリバース

データリバースは、オブジェクトの核となるデータを抽出する過程である。大きくは、リバース対象のソフトウェアシステム内に存在する「物理データ」と実世界を表現する「概念データ」との関係を明らかにするデータモデリング過程と、プログラマによって任意に設定されているプログラム内の異音同義データの洗い出しやその重複状況、依存状況を明らかにし、オブジェクトを把握するプログラム内データ分析過程に分けられる。

(2) プロセス/ルールリバース

プロセス/ルールリバースは、プログラムを解析して、それぞれのデータの生成、変更、消滅に関する処理(LCP: Life Cycle Process)を抽出し、カプセル化するプロセスリバース過程と、データや業務処理に固有な制約を抽出し、カプセル化するルールリバース過程に分けられる。

5. リバースエンジニアリング適用結果

ここでは、実システムより選び出した5本のCOBOLプログラムの例について、データリバース、プロセス/ルールリバースを行った作業の概要と結果について述べる。表1は、解析対象プログラムの概要を示す。

表1 試用用COBOLプログラム概要

機能概要	処理形態	言語	DB/DC	ステップ数
顧客情報全般の管理	オンライン	OS/VS COBOL	IMS	7536
	オンライン			6793
	オンライン			5693
	バッチ			916
	バッチ			4262
合計				25200

(1) データリバース作業概要及び結果

データリバースでは、既存プログラム内で使用されているデータのタイプ、桁数等の属性抽出の他に異なる同義データの分析、業務データの抽出、ドメイン/標準データ名称の付与等を行う。表2は、解析対象プログラムで使用しているデータ項目数と異なる同義データ整理後のデータ項目数を示している。

表2 データリバース後のデータ項目数

プログラム名	使用データ項目数	プログラム内同義データ項目数	プログラム間同義データ項目
PGM1	428	293	
PGM2	470	356	
PGM3	386	258	798
PGM4	77	69	
PGM5	355	219	
合計	1716	1195	798

ドメイン 128個

(2) プロセス/ルールリバース作業概要及び結果

プロセス/ルールリバースでは、ソースコードを基本的なデータ操作を表現する2項オブジェクトと2項オブジェクトとして表現されたプロセスに対する制御を表現した制御オブジェクトに変換/整理する。ここで言う2項オブジェクトとは、データ操作を1対の(2つの)データに対するものとして捉え、その操作と1対のデータから構成される集約オブジェクトと見なしたものである。例えば

MOVE A TO B.

として表現されるプロセスは、オブジェクト識別子をCP01とすると

CP01 [(A, B) MOVE]

と表現される。実際のソースコードをオブジェクトに変換した例を図3に示す。

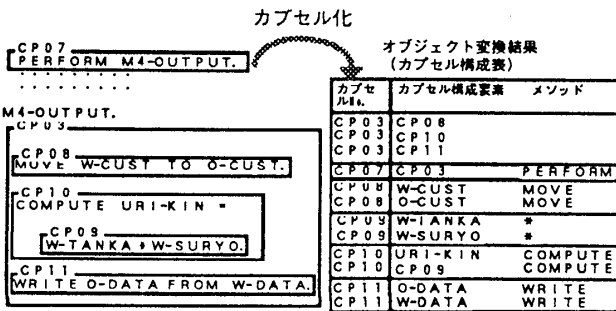


図3 オブジェクト変換例

このような考えに従い、プロセス/ルールリバースを行った結果を表3に示す。

表3 プロセス/ルールリバース結果

プログラム名	プロセス/ルールステップ数	重複排除前カプセル数	カプセル数/ステップ数	重複排除後カプセル数	重複率 (%)
PGM1	1122	1210	1.08	762	37
PGM2	1692	1865	1.10	1072	43
PGM3	1085	1162	1.07	781	33
PGM4	187	199	1.06	138	31
PGM5	406	433	1.07	382	12
合計	4492	4869	1.08	3135	36

(3) 業務ルール抽出

業務ルール抽出では、データリバースで抽出/整理した業務データに対し、プログラムの中でシステム環境やエラーフラグ等の処理を排除した、本来業務として行われるべき操作、制約のみを抽出する。図4は、業務ルールとして抽出された情報をブラウザを通して参照した例である。

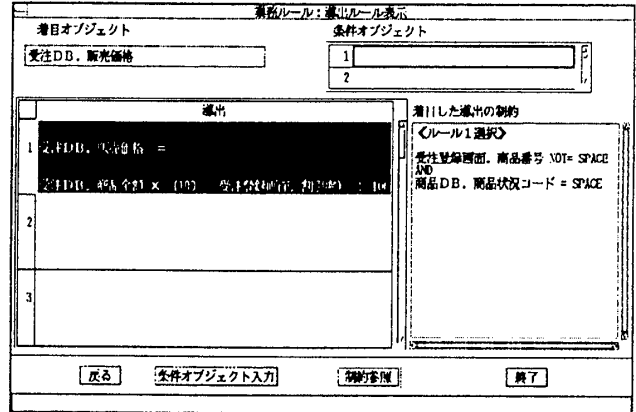


図4 業務ルール抽出結果

6. まとめ

今回行ったデータ中心アプローチによるリバースエンジニアリング作業によって、プログラム毎に作成されたデータ名称の標準化や処理の重複を抽出/整理することができた。また、業務データに着目したプロセスやルールを分析することで、既存システムの理解や再利用可能要素の抽出まで行えることが確認できた。

しかしリエンジニアリング技術を確立させるためには、リバース結果の再利用可能要素からCASEツール等のフォワード過程で使用される再利用部品を作成しなければならない。今後は、本テーマについての検討を行っていく。

参考文献

- 堀内 一: データ中心システム設計、オーム社 (1989)
- 山川敦夫 秋庭真一 他: データ中心によるプログラム論理抽出、情報処理学会情報システム研究会 研究報告93年1月号
- 堀内 一、飯田啓三 他: データ中心によるリエンジニアリングの方法、情報処理学会情報システム研究会 研究報告93年1月号
- 堀内 一、飯田啓三 他: データ中心アプローチによるリエンジニアリング支援技術の適用実験 日立評論93年11月号
- 堀内 一: ソフトウェア再利用を可能とする方法論とツール、情報処理学会ビジネスプロセスリエンジニアリングとソフトウェア技術チュートリアル 94年2月