

CASE 実施例

—大阪ガスのケース—

明神 知

1. はじめに

オージス総研は、1990年よりCASE導入に取り組む、すでに18プロジェクトへの適用実績を持つ。最近では、BPRを目的とし、業務の改革、システム開発プロセスの改革をビジネス・モデリングおよびデータ中心アプローチで実施する例が増えてきている。また、クライアント・サーバーシステムへの対応も急で、ミッションクリチカルなビジネスアプリケーションについてもCASEによる開発が有効である。

本稿では、当社のCASE適用事例を中心に、関連する技術について紹介する。また、当社の今後の取り組みについて概観し、これからCASEを適用される読者の一助としたい。

2. CASE導入のねらい

1) 導入のねらい

オージス総研は大阪ガス㈱の100%子会社で、大阪

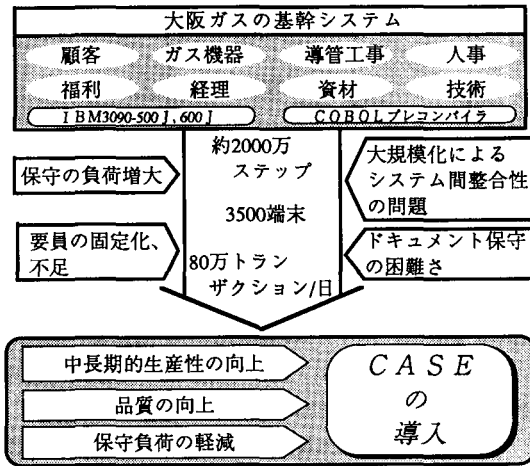


図1 CASE導入のねらい (1989年時点)

ガスの基幹システムの中長期的生産性の向上、品質の向上、保守負荷の軽減を目的として、1990年からCASE導入に取り組んだ。

CASEに期待したのは、保守の負荷増大、要員の固定化、大規模なシステム間の不整合、ドキュメント保守の困難さを解決することであり(図1)、「1サイクルのシステム開発の生産性向上だけでなく、システム保守の生産性向上までを考慮すること」であった。

2) 導入経緯

1989年9月にIBMからAD/Cycleの発表があり、システム開発の問題を解決するソリューションとしてCASEに期待が集まった。

ただ、CASEを実用に供するには推進体制や開発環境、教育、管理など多くの課題があった。これらを解決するために、1990年2月にCASE推進プロジェクトを発足させ、CASEツール導入の前提となるシステム開発方法論とCASEツールの調査、研究と開発環境の整備を行なっていった。(図2)

当社では、システム開発方法論として、次の理由からIE(インフォメーション・エンジニアリング)を

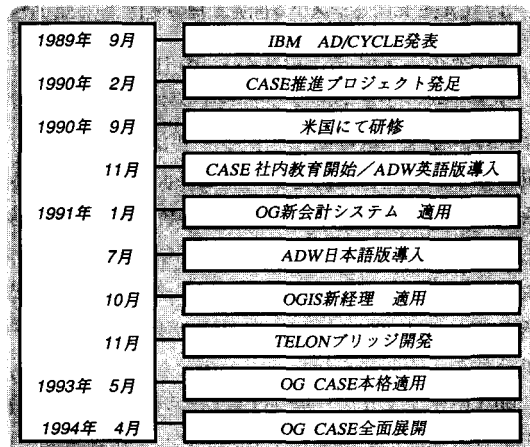


図2 CASE導入経緯

みょうじん さとる (株)オージス総研
〒550 大阪市西区千代崎3-2-95

ベースの方法論として採用した。

- (1) 全開発工程をカバーしている
- (2) データ中心アプローチ (DOA) である
- (3) データ系とプロセス系の連携
- (4) 方法論を具現化したツールがある
- (5) わかりやすい

また、CASE ツールとして次の理由から ADW を選択した。

- (1) IE に沿ったツールである
 - (2) 上流から下流まで統合化している
 - (3) データ設計とプロセス設計をサポートし、整合性を保っている
 - (4) リポジトリとの連携機能がある
 - (5) 外部とのデータ交換が可能である (オープン性)
 - (6) バッチ、オンライン両方をサポートしている
 - (7) IMS, CICS, DB2 に対応している
 - (8) IBM の AD/Cycle に含まれている
- 特に、オープン性は最近のクライアント・サーバシステム (CSS) への対応において重要かつ有効な与件であった。

また、情報システム部門の役割の急激な変化の中で、安定した情報基盤としてのデータモデルを整備し、データを核にしてユーザコンピューティングを推進するためにも、DOA でリポジトリベースの開発法を選択したのは正しい決定であったと言える。

3) CASE によるシステム開発工程の変化

データ中心アプローチ (DOA) の IE (インフォメーション・エンジニアリング) によれば、従来の直線的なプロセス中心の開発方法からデータ分析を重視する多面的な開発になる。ADW の特筆すべきは、プロセス分析とデータ分析の間にインフォメーション・ビューという関連がつけられ、プロセスで使われていないデータがないか、未定義のデータはないかなど、有機的な連携を保ちながら整合性のある開発を行なうことができることである。

図3に示すように、開発手順についてもデータ分析とプロセス分析を融合させて各モデルを洗練させていくことになる。

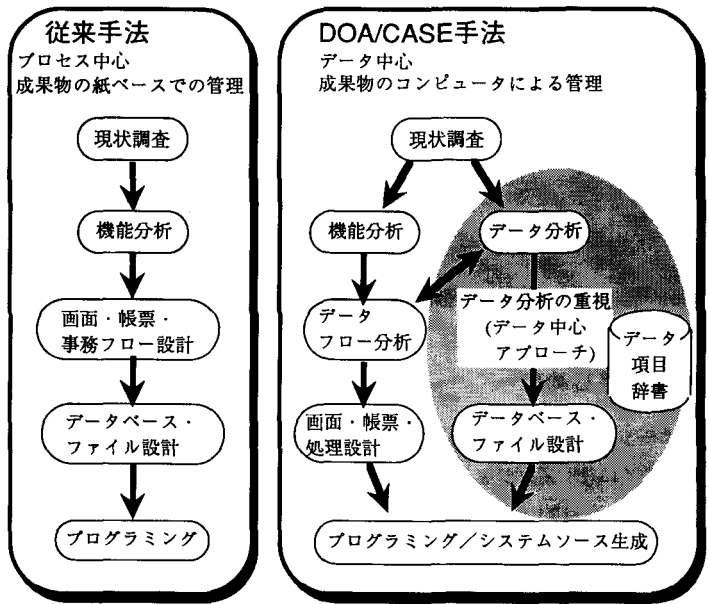


図3 DOA/CASE手法

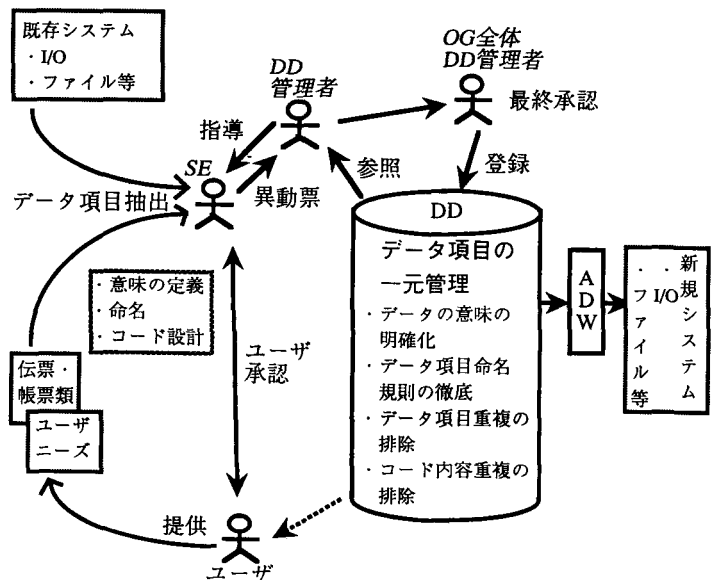


図4 データ項目辞書

また、図4に示すデータ項目辞書を整備し、ユーザー業務の言葉で、システム内部のデータ項目を一元管理している。これにより、要件定義における誤解を防ぎ、保守でのユーザーの間合わせに対して、ユーザー業務とシステム定義データの連携がとれるようになった。管理すべきデータ項目数は、従来と比べて50~80%に激減する。

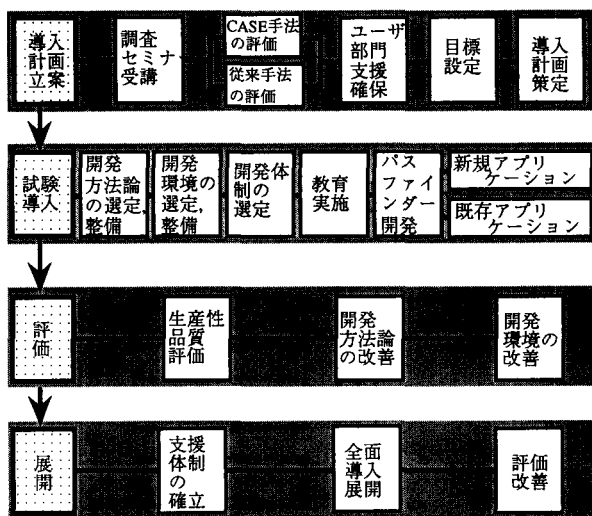


図5 CASE導入プロセス

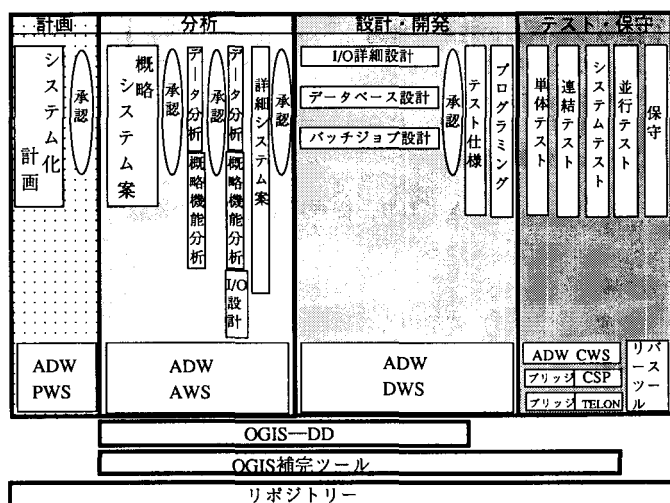


図6 オージス開発方法論

3. CASE 導入アプローチ

1) 導入プロセス

単発的なパイロットでなく、4～10パイロットを継続して定着させていくことにより開発の文化を変革させていった。図5には、各ステップの作業項目を示した。CASEはツールの導入だけでなく、教育、開発環境の整備や方法論の整備など、多くの作業をタイムリーに進めていく必要がある。IEとDOAを融合させ、実績を反映させて最終的には図6に示すオージス開発方法論を整備した。

2) 教育

教育は変化に対するマネジメントの上でも重要で、

一般的にはDOA概要、CASEツール操作、DOA技法などを各レベルの参画者に行なう必要がある。

当社では、特に分析作業についてはマルチメディアCAIを利用して演習形式の擬似体験が行なえるように工夫し、学習効果を上げている。

3) 開発管理

図7のような体制を作り、トップレベルの理解を得て、企業レベルで取り組んだ。

データ項目については、辞書管理委員会を開催して、データ項目追加、修正を審議する。その結果、同音異義や異音同義語が排除された。

4. CASE 適用事例

4.1 初期の問題点

現在までに、社内外18案件、250万ステップ以上のCASE適用実績を持つに到ったが、すべてのプロジェクトが順調にいったわけではない。CASE導入の初期プロジェクトにおいては、以下のような問題を克服する必要があった。

これらの実績の中から、IEを独自にカスタマイズし、オージス総研方法論を整備していった。

1) ユーザおよびマネジメントの

CASEダイアグラムレビューの拒否
ERDやDFDは、それをレビューするにはある程度の訓練が必要である。またその内容は業務そのものであるため、ユーザが

自らその開発に参画する意識が重要である。

当初はユーザやレビューアに負担を強いたこともあったが、現在では、補完のドキュメントを作成し、ユーザ教育も推進し、ユーザ参画型開発を行なっている。

2) 分析工程作業の増加

従来の上流工程においては、データ項目辞書の整備やデータモデリング、プロセスモデリングといった作業は存在しない。

したがって上流工程の期間も長くなり、マネジメントのチェックも従来通りのフェーズ単位では遅くなってしまふ。そこで、分析工程を開発規模に応じて3分割し、3段階承認方式とした。規模が小さい場合は、

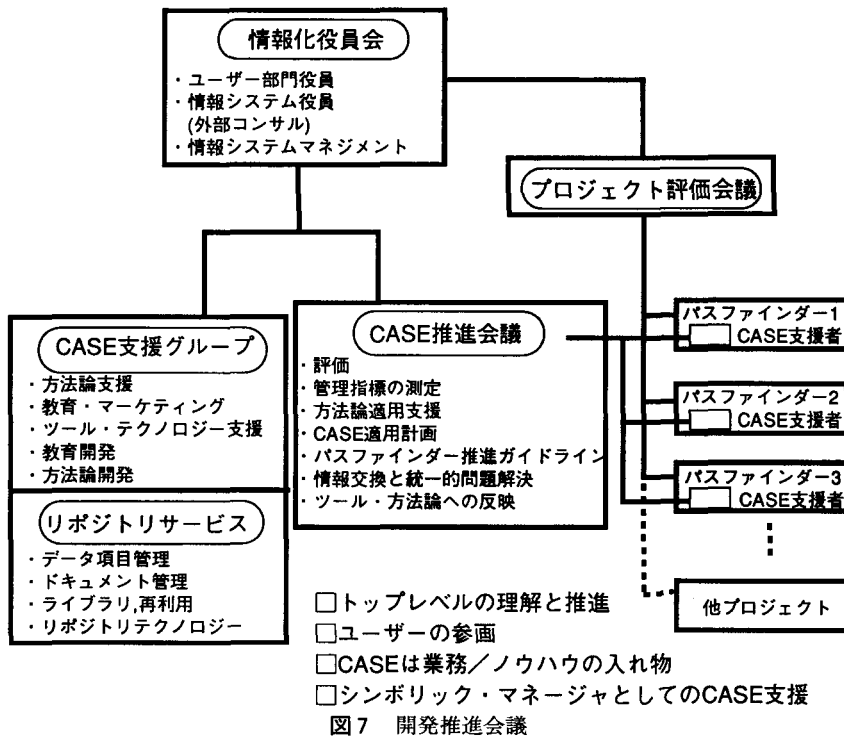


図7 開発推進会議

2 回目の承認は割愛した。(図6 参照)

3) ベテラン SE の反発

短期の生産性で比較するならば、従来手法の方が高生産性である。これは、従来手法ではライブラリや運用環境、さらには教育などが整備されており、利用するツールなども枯れた技術になっているからである。

新技術の場合、それが革新的であればあるほど、100%完全な形では登場してこない。

したがって、保守まで含めた中長期の生産性、品質向上をねらい、短期的な評価で全体を判断せず、継続的な改善が必要である。またモデルベース、リポジトリベースの開発という本質的な部分を落としてお絵書きツール的な利用に走れば、一時的な生産性は上がるが、保守においては従来と同じ混乱に陥ることになる。

4) 進捗管理の精度

新しい手法は工数見積りとその進捗管理において経験値がないために精度の高い管理がどうしても困難になる。

これに初期の段階で対処するためには、細かくフェーズを切って見積もることと、手法によって変動のない機能ベースの見積りであるファンクションポイントの導入などが有効である。

現在では、経験値も蓄積されてきたので、精度も上

がってきており、従来手法と同等の進捗管理が可能となっている。

5) 若手中心の体制による業務知識不足

CASEの神話として、システム開発経験の乏しい若手でもツールと方法論によって開発が可能というのがあった。これは一部正しいが、やはり上流の論理から下流の物理設計に至るまでを見通して、自信を持って進捗管理を行なうためには、従来方式におけるシステム開発の経験や業務の知識が必要である。したがって、従来方式を経験した新技術の吸収に積極的なリーダーは必須である。ただし、確実に平均年齢を下げるができるのは事実である。

業務知識については、最低限度の知識は必要であるが、ERD、DFDはユーザとSEの共通言語になるので、SEは業務に精通していなくても、業務の無駄や問題を客観的に指摘することが可能となった。

6) 論理モデルと物理モデルのギャップ

ERD、DFDともにIEの教科書では論理と物理は区別して書くように指導されているが、現実には、スモールステップで順次論理から物理へ移行する必要がある。この問題は、上流から下流までほとんどギャップのないオブジェクト指向でも同じである。

4.2 適用事例

表1に示すように、現在までに社内外18案件、250

表1 CASE適用実績例

アプリケーション	規模	適用フェーズ				リリース
		計画	分析	設計	構築	
新会計システム	100万ステップ DB2, CICS			従来手法 TELON	PCOB TELON	93.4 完了
運用管理システム	20万ステップ DB2, CICS		AWS OGIS-DD			93.10 完了
機器管理システム	12万ステップ IMS				CWS	93.4 完了
弊社 開発情報 管理システム	5千ステップ DB2, CICS					92.10 完了
業績評価システム	14万ステップ DB2, CICS		従来手法 OGIS-DD		PCOB CSP	93.4 完了
弊社経理システム	40万ステップ DB2, CICS			DWS	CSP2 AD	94.4 完了
弊社 固定資産 管理システム	4万ステップ DB2, CICS		PWS		CWS	92.9 完了
設備点検情報 管理システム	3.5万ステップ IMS/DB, DB2				CSP	94.3 完了
セキュリティ・ サービス・システム	1万ステップ DBMS				Visual BASIC	94.4 完了
部品管理システム	15万ステップ		AWS OGIS-DD		C/S	95年度
健康保険 管理システム	5万ステップ(予定) IMS/DB, DB2, CICS(予定)			DWS	CWS	95.3
A社 財務会計再構築	26万ステップ (予定)				SAP	96.3
B社 原価管理システム	10万ステップ (予定)				Visual BASIC	95.3
C社 支払基金請求 処理システム	20万ステップ (予定)			DWS	CWS	95.3

万ステップ以上の適用実績を持つ。

最近ではBPRを目的に、業務をモデル化し、パッケージの活用を前提として、現業務の無駄を排除し、新規案件を確実に定義したうえで、下流の構築部分は限定して、可能な限りパッケージを利用し、維持管理を含めたシステムのユーザ移管を推進している。

これまでに得られた成果を図8に示す。以下代表的な事例を紹介する。

4.2.1 設備点検情報管理システム('94)

1) 概要

分析から構築までの一貫CASE適用、IMS/DB, DB2, 3.5万ステップ(COBOL換算)。

2) 効果

図9に示す生産性が確認された。COBOLによる従来手法(当社比)に比べて14.5%の生産性向上である。この数字自体はあまり大きくないが、要件定義の確実な作り込みにより、手戻りや修正が減少することや、データ項目の整合性と統一性により、保守フェーズでの生産性向上が大きい。

項目	成果物	効果
リポジトリ	・リポジトリモデル開発 ・DDの開発 ・エンサイクロペディア管理ツール	生産性 (予測) 新規 ・分析 約20%向上 ・設計 約10%向上 ・製作 約20%向上 全体で14.5%向上
ツール部・品	・ADW補完ツール 各種ジェネレータ ・プログラム部品 ・会計モデルテンプレート	品質 ・分析での成果物の均質化 及び漏れの減少 ・ユーザとSE間でのコミュニケーション円滑化 ・データ項目の整合性と統一化
教育	・分析技法解説書 ・分析演習課題集(基礎編) ・分析演習課題集(応用編) ・設計課題集 ・指導要領書 ・CAI	効率 ・手順の標準化 ・分析成果物の設計での直接利用 ・設計書が直接システム資源定義に ・1人1台のW/S利用
方法論	・開発基準書(手順編) ・開発基準書(保守編) ・開発基準書(技法編)	要員育成 ・約100名のCASE要員育成
環境整備	・CASE開発用W/S150台 ・LAN設備 ・ADW104セット (現在400セット)	保守の生産性 ・保守すべき成果物は増えている(上流の成果物)にも関わらず工数は変化がない。
リエンジニアリング	・DBRAD/Eを用いた保守支援技術の開発 ・DOAの考え方に基づくリバース技術の開発	既存システム ・既存システム資源の間の関連情報取得の自動化 ・FMP維持管理の一部自動化

図8 CASE整備の成果

従来手法(COBOL)

サ ー ベ イ	業 務 分 析	シ ス テ ム 化 設 計	設 計	製 作	テ ス ト	100%
13	7	10	14	40	16	

CASE

サ ー ベ イ	業 務 分 析	シ ス テ ム 化 設 計	設 計	製 作	テ ス ト	85.5%
	80%	80%	90%	80%	100%	

図9 開発生産性

従来手法

サ ー ベ イ	基 本 設 計 詳 細 設 計	プ ロ グ ラ ミ ン グ	テ ス ト	100
5	47	29	19	

CASE

分 析	設 計	製 作	テ ス ト	105 (80)
38 (30)	29 (19)	38 (31)		

図10 開発生産性

()内はオーバーヘッドを除いた工数%比較

4.2.2 大阪ガス経理システム ('93)

1) 概 要

分析のみの適用 (下流ツールは日本語化されていなかった)。100万ステップ (COBOL 換算), 300画面, DB 2, CICS.

2) 効 果

- ・分析工程の工数が7%削減できた: 初期オーバーヘッド (教育, 開発標準作成, 開発環境の整備) を除く
- ・品質の向上 (分析漏れ減少, 理解の深まり, 矛盾/影響範囲の特定が容易)
- ・保守作業の軽減 (変更が容易)

4.2.3 固定資産システム ('91)

1) 概 要

CASE適用のパイロットとして取り組んだ。4万ステップ (COBOL 換算), DB 2, CICS.

2) 効 果

初期オーバーヘッドを除いて26%の工数を削減した。

ユーザニーズの理解の容易性や属人性の排除, 一元管理による品質の向上, 保守作業の軽減, 要員育成。

4.2.4 当社経理システム

1) 概 要

CASEを上流から下流まで初めて本格的に取り組んだシステム。40万ステップ, DB 2, CICS.

2) 効 果

CSPパターン部品利用率30~40%で初期オーバーヘッドを除いて20%の生産性向上が達成された。(図10) 手戻り作業が, 理解の深まり, 分析漏れの減少, 影響範囲の特定が容易になった。

4.3 CASE導入の成功要因

これまでの適用実績の中から, CASE技術導入における成功要因をまとめると以下ようになる。

1) トップマネジメントによる推進

従来手法でも可能であるのに, あえてCASEでやる目的は, ユーザ業務のモデリングによるBPRであり, システム開発の改革であり, 情報資源管理のしくみを作ることに伴う保守の生産性向上や, ユーザコンピューティングの推進などである。これらは, 強烈な問題意識を持ったトップによる推進によらねば実行は困難である。

2) 変化に対するマネジメント

既存の技術に習熟したベテランが見せる抵抗は, 既存のスキルやノウハウの否定, 伝統への安住などが原因である。技術論よりも変化に対する不安の方が大きいので, これを管理するために, PRや教育, 事例紹介や, 自分にとってのメリットを示すなどの変化に対するマネジメントが必要である。

3) 継続する実適用

1回のパイロットプロジェクトで劇的な成果が出ないからといって, それで中止しては何も残らない。

生産性や品質を定量的に計測して, 何が原因で目標に達成しなかったかを追求し, 改善を繰り返さなければならぬ。

4) パイロットよりもパスファインダー

最初のプロジェクトにあまりにも小規模で重要な案件を選出すると, それが成功してもインパクトが少なく, 体制や計画も十分されずに実施されることが多く, 失敗につながりやすい。

中程度の規模で必ず成功させねばならない案件を選択すべきである。

5. 関連技術

5.1 リポジトリ

統合化 CASE はリポジトリに開発成果物を格納して、互いに連携されている。保守フェーズにおける生産性の向上は、このリポジトリによって実現される。

一般に CASE ツールが持つリポジトリはプロジェクト単位の開発時の機能である場合が多い。

ADW のリポジトリ (エンサイクロペディア) も例外でなく、WGC (ワークグループコーディネータ) や WGM (ワークグループマネージャ) でエンサイクロペディア管理の効率化や共有化、統合化が順次可能となってきたが、運用環境や既存システムも含めた全社的情報資源管理の実現まで至っていない。

最近、メインフレームの他にワークステーションや PC 上で稼動するオープンリポジトリ製品が出てきており、ADW の全社運用、保守時のリポジトリとして、R&O 社の Rochade が推奨されている。これによれば、企業レベルの情報資源管理が実現される構成管理、変更管理、工程管理、変更影響調査は、このリポジトリを通して行なうことになる。また、ビジネス用語で定義されたデータ辞書を通して、ユーザが業務の用語で既存システムやデータベースをアクセスできるので、ユーザコンピューティングのプラットフォームになる。

今後、クライアント・サーバシステムの構築が増加して、情報資源が分散すればするほど、プロジェクト単位、部門単位、企業単位での管理ニーズがますます高まり、このようなりポジトリの導入例が増えていくと予想される。

5.2 リバースエンジニアリング

新規に開発する場合、CASE によりリポジトリベースの開発を行なえば保守の生産性が向上する。一方、既存システムを解析して設計情報や成果物関連情報を復元、変換してリポジトリに戻すのがリバースエンジニアリングである。

当社では、日立製作所と共同研究で開発したデータ中心リエンジニアリング (DORE) を利用して、既存システムのプログラム、ファイルマスタ、ファイルレイアウト、コード集、JCL などを解析して、データ項目辞書 (DD)、ディレクトリシステム (DS) に登録し、関連や一覧を表示する保守支援システムを構築している。

5.3 CASE による BPR

BPR について、アメリカの場合の問題は業務設計図と現実の乖離であるが、日本では業務設計図そのものがなく、ベテランからの知恵の継承が途絶えていることである。今、まさにビジネス側からも、システム側からも客観的に評価できる共通言語としてのビジネス・モデルを構築して、改革に取り組む必要がある。

5.4 クライアント・サーバ (C/S) 対応

大規模な基幹業務を C/S システムで開発する例が増えている。特殊な業務については、オブジェクト指向や GUI ベースの下流 CASE ツールが多数商品化されているので、上流 CASE からのデータ系、プロセス系の分析情報をブリッジして開発することになる。データ系は分散すればするほど上流 CASE によるデータモデリングによって全体を管理する必要性が増えている。

一方、標準的な業務については、SAP などの基幹業務の C/S 型パッケージの導入により、パッケージの機能を前提として現行業務のビジネスモデルを作成し、新規開発部分を最小化するならば、まさに BPR を実現したことになり、ユーザコンピューティングも一挙に進むことになる。

6. おわりに

1990 年にスタートした当社における CASE の導入は、IT (情報技術) - メインフレーム中心からダウンサイジングへ - とビジネス環境 - システム開発の改革から業務の改革へ - の激的な変化に会いながらも、データ中心アプローチとリポジトリベースの開発という本質は変わらず、定着してきた。

オブジェクト指向への移行もデータ中心アプローチからスムーズに行なえるので、DOA によってシステム開発の改革に取り組んでいきたい。

参考文献

- [1] Making CASE Work, John Parkinson, 1991 NCC Blackwell
- [2] Inside ADW and IEF, McGraw-Hill, Inc., 1993
- [3] 山川, 他: データ中心アプローチによるプログラム論理の抽出, 情報処理学会, 研究報告, 93-IS-42 (1993)