



DOA+

DOA+とは
研究報告書
記事

コンソーシアム御案内

コンソーシアムの目的
会 則
組 織
活動案内
入会のご案内
お問合せ
個人情報保護

会員向けページ

第1分科会
第2分科会
第3分科会
会員情報の確認と変更

● システム開発技術の変遷とDOAの歴史

コンピュータは情報処理の自動化ツールである。そこでまずPOA(Process Oriented Approach)によるプログラム開発が始まった。特定ユーザの要件を属人的プログラム仕様書としてまとめ、プログラミングが行われた。手間のかかる計算が自動化され、インプットデータを与えるだけで正しいアウトプットが得られ業務が合理化された。おおよそ1960年代の話である。

ファイルが使えるようになって、バケツリレー方式でプログラムを連携して一連の処理ができるようになったが、IPOと言って、プログラムを中心にインプット、アウトプットを考えていた。1970年代になって親子関係や、BOMのような複雑なデータ構造を扱える構造型DBMSが登場したが、プログラム中心の考え方はほとんど変わらなかった。

1980年代になってRDBが登場して、データベースを中心にする考え方が登場した。ユーザの欲しいのは画面・帳票すなわち業務遂行のためのデータである。プログラムはこれを提供するための手段にすぎない。素材となるデータをストアするデータベースを中心にシステムを構築すべきだ。1985年に日経コンピュータ誌上に登場した堀内一先生(当時日立製作所)のDOAの主張である。

DOAが認知され、データモデリングが行われたが、多くは個別アプリケーションDBのテーブル設計を目的とするものであった。CASEツールが登場し開発の生産性アップが図られたが、アプリケーション間のデータの共用/重複などは見過ごされ、孤島システムが乱立する結果となった。実装独立のドキュメントは用意されず、ビジネスやITの変化に対応する保守は、ソフトウェア実体に対する差分メンテや温泉旅館別館建て増し方式で行われ、属人化・暗黙知化が進行した。

データ総研は1985年創立し、THデータモデルをベースにしたデータモデリング方法論PLAN-DBIによるコンサルティングを開始した。その主張、「DBMSは意識せず業務データを実装独立に整理する」、「エンティティをリソース・イベント・要約・在庫型・断面などに分類し、リソースDB(マスターDB)は個別アプリケーションDBとは分離して1個作る」をいれて、複数個のアプリケーションに対するデータモデリングを実施した企業では、データインフラの構築に成功し、孤島システム化・暗黙知化を免れることができた。

1990年代に入り、バブルがはじけ、短期にソリューションを求める傾向から、C/S化、ERP導入、Y2K、Web化などの課題の前に、データインフラ構築のためのDOAは忘れられがちとなった。主要な個別アプリケーションは、一応システム化され、現行システムのデータ構造を手直しすれば事足りるとか、パッケージ活用だからというケースが増えたためでもある。しかしその結果、データ構造からシステムを把握できる、ゼロからシステムを設計できる人材が手薄となった。すなわち画面・帳票に表れたユーザの情報要求を分析・調整してこれを満たす共用データを定め、情報システムの設計図としてまとめる人が育っていないのである。

しかし、2000年を過ぎて、DOA復活のニーズがふたつ登場しているようである。

ひとつは広域データ流通である。SCM、CRM、連結会計など、それもアプリケーション、企業、国をまたがったデータ流通が、在庫ミニマム、業務スピードアップ対応などのために要求され、従来の孤島システムからの脱皮が課題となっている。かつてのDOAが個別アプリケーションの中でのDOAであったが、今回はこれをまたがる1レベル上の個別アプリケーション間のDOAである。

もうひとつは、OOからのDOAへの関心である。安価・高性能のPCの普及により、GUI入力が当たり前となって、OOP技術の進化が促進され、OO技術の上流への適用が追及されている。MDA(Model Driven Approach)はその典型と言える。しかし業務システムは画面・帳票すなわちデータからスタートする。したがってどこかでDOAからOOPに繋がなければならない。業務システムを対象とするOO技術者のDOAへの関心が高まりつつある状況と言える。

かつては処理を担当するコンピュータがホスト(主人)、その周りにファイル・DBが配置されたアーキテクチャであった。今はネットワークを従えたDB群が中心にあつて、これを共用するコンピュータが周辺に存在するアーキテクチャとなっている。ハード環境におけるPOAからDOAへのシフトである。個体と個体が直接コミュニケーションすると $n(n-1)/2$ 個のスパゲッティインターフェースができる。これを、ハブを設け個体と場の関係にすれば n 個のインターフェースに収めることができる。この共用資源であるデータを中心に置くデータ中心アプローチこそが、今後の大規模システムの自然なアーキテクチャとなるはずではないか。

