

CASEフレームワークにおける 制御連携／データ統合方式

2M-9

森本孝司 熊谷克夫
(株) 東芝 青梅工場

1. はじめに

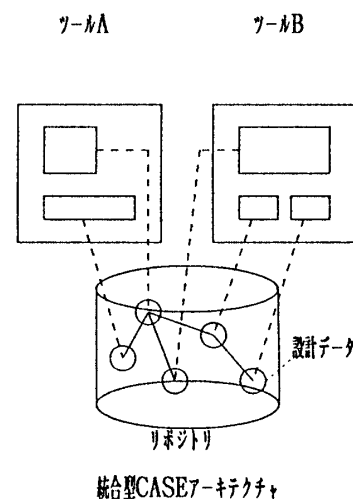
統合化されたCASE環境の実現形態には、全てのCASEツールを自前で用意する統合型および統合化の枠組みを用意するフレームワーク型の2種類がある。統合型CASEは、自前で用意するツール群を利用することにより、データの一元管理およびプログラムの自動生成等の優れた機能を持つ半面、提供されたもの以外のCASEツールを自環境に取り込めない等の短所がある。フレームワーク型CASEは、各種ツールを自環境に取り込みツールの制御の連携は可能だが、設計項目の管理等はできない。ここでは、両者の長所を取り入れた制御連携／データ統合方式を持つCASEフレームワークのメカニズムについて述べる。

2. 統合型CASEとフレームワーク型CASEの比較

それぞれのCASE環境の特徴をデータ統合、制御連携の観点で比較すると次のようになる。

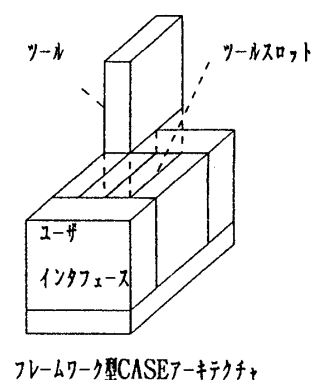
(1) データ統合

データ統合における機能要件は、CASE環境がサポートするソフトウェア開発のライフサイクルにおける各プロセスでのデータの一元管理、バージョン管理およびそれらの共用である。また、データはリポジトリで管理されるが、その管理レベルはそのデータの種類により、設計項目単位（変数名等）あるいはファイル単位で管理されなければならない。統合型CASEツールの場合、設計項目単位でのデータの一元管理、ツール間でのデータの共用が可能となり、これによりシームレスな開発の流れを支援することができる。一方、フレームワーク型CASEの場合は、個々のCASEツールで作成されるデータ（ファイル）はブラックボックスとして扱われるため、ファイル単位での管理は可能となるが、ソフトウェア設計／プログラム設計・・・／保守等がそれぞれ独立した作業となるため、データの一元管理を前提とした支援が十分にできないという欠点がある。



(2) 制御の連携

制御連携に必要な機能は、ソフトウェア開発サイクルの各プロセスでCASEツールが必要に応じて起動できると共にそれらが連動することである。統合型CASEの場合、CASEツールを自前で用意しているため、きめ細かなツールの起動および連動が可能となる。しかし、全てのツールを1つのCASEベンダが用意するのは現実的ではない。また、統合CASEは上記に記述したデータ統合との関連で他のベンダが用意するCASEツールを組み込むことは容易ではない。一方、フレームワ



Control and Data Integration Method of CASE Framework

Takashi Morimoto, Katsuo Kumagai

Ome Works, TOSHIBA Corporation

ーク型CASEは、制御連携を中心に用意するCASEツールの制御連携は容易に実現可能である。

3. CASEフレームワーク

上記の制御連携およびデータ統合に関する比較をもとに両者の長所を取り入れたCASEフレームワークのメカニズムについて記述する。このCASEフレームワークの機能は、統合型CASEの機能を基本とし、フレームワーク型CASEの特徴である柔軟な制御連携の機能を提供する。

本フレームワークの基本としては、項目単位で管理しなければならないデータに関しては、専用の仕様書エディタを利用し、情報をブラックボックスでとらえて良いものに関しては、他が提供する各種CASEツールを利用し、ソフトウェア開発を進めることができる機構を提供することである。

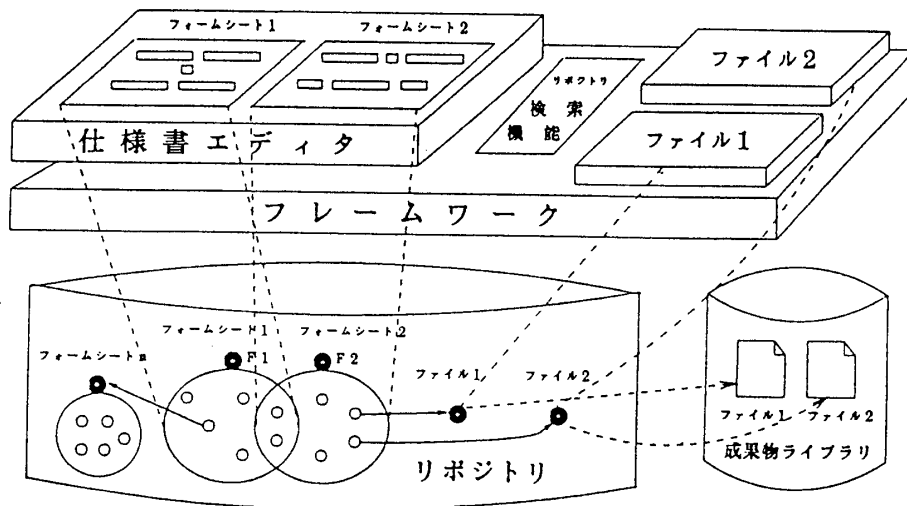
提供される機能は以下ようになる。

(1) 自提供CASEツールと他CASEツールのデータの統合

項目単位で管理されているデータとファイル単位で管理されている情報のデータ関係管理を行うことにより、自前のCASEツールで作成していない情報をリポジトリで一元管理を行う。具体的には、ソフトウェア構成要素（ソース、各種リソース等）の概要／詳細／項目の定義を仕様書エディタで作成し、実際の構成要素は、それぞれの専用ツールを利用して行う。自前で用意したCASEツールと他ベンダが提供するCASEツールの情報を一元的に管理することにより、チェックイン／アウト、構成の整合性の検証等が全ての構成要素に対して可能となる。

(2) 柔軟な制御連携

上記のデータ統合を基本に、全てのCASEツールに対して同一のユーザインタフェースのもとで各種の制御連携が可能となる。連携機構としては、GUIによるオブジェクト指向型ツール起動、リポジトリに格納されている項目から関連するツールの起動、ハイパーテキスト的なツール起動／リンク等から構成されている。なお、フレームワークへの取り込みについては、ツール側の変更無しに統合化をすることが可能である。



4. おわりに

今回、ここで述べたデータ統合／制御については開発を完了し、現在、評価を行っている。これらの機能はCASEフレームワークの基本機能であり、今後はワークフロー／グループウェアとの統合化、分散開発への対応を行っていく。