

EUCによる分散システム上のアプリケーション構築に関する検討

6T-5

甲斐 こずえ 中條 祐一
NTT 情報システム本部

1 現状

現状の業務の多くは、メインフレーム集中型のシステム上に構築されている。そのため、以下のような問題が生じている。

- ・利用者の多様な要求すべてに対応できない
- ・システムの変更のためのコストが膨大

このため、業務を行う人が主体となり、利用者の要求にタイムリーに対応できるシステムが必要になっている。

一方で、高性能PCや高機能OS、ビジュアルプログラミングツールの普及により、利用者が比較的容易にアプリケーションを開発できる環境が整いつつある。

しかし、これらの利用者が開発したアプリケーションは、データの分析など、業務システムでは提供されていない情報系業務への適用が多く、基幹系業務への適用は、あまり行われていない。

2 目的

上記の問題を解決するため、基幹系の業務についても、利用者が必要とする機能を利用者自身が実現するという考え方に基いてシステムを構築する。

この実現に当たり、以下の二点の実施が不可欠である。

- (1)集中型システムから分散システムへの移行
- (2)EUC (エンドユーザコンピューティング) の推進

本稿では、特に(2)について検討を行う。

2.1 実現のための条件

EUCによるアプリケーションプログラムの構築に当たり、アーキテクチャ及び開発手法を確立する必要がある。

これらを確立するに当たって、以下の二点を考慮しなくてはならない。

- (1)開発期間の短縮化と開発用コストの削減、及び重複開発の防止
- (2)マルチベンダ性と選択の多様性、及び柔軟性の確保

3 実現方法

本稿での検討では、以下の二つのコンセプトに基づいた、EUC推進の実現方法を検討する。

- (1)3層クライアント/サーバ・アーキテクチャ
- (2)コンポーネントウェア

3.1 3層クライアント/サーバ・アーキテクチャ

アプリケーションをPresentation (HUI部)、Functionality (機能部)、Data (データ提供部)の3つの層に分割し、個々の機能を独立にする考え方 (Fig.1)であり、アプリケーションの柔軟性や変更容易性の向上を図るものである。

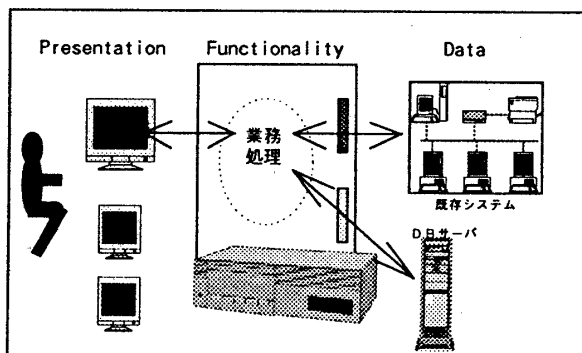


Fig1. 3層クライアント/サーバ・アーキテクチャ

3.1.1 問題点

- (1)3層クライアント/サーバ・アーキテクチャとは、アプリケーションの論理的な構造を示した

Examination of construction on the distributed system which used End User Computing
Kozue Kai, Yuichi Chujo
NTT Information Systems Headquarters

ものであり、実装についての規定はない。

- (2)実装に使用するツールやアプリケーションを動作させる実行環境には、製品に依存した機能やインタフェースの差異が存在する。

3.1.2 対策案

(1)実装の規定

アプリケーションプログラムの論理的な構造から、実際のプログラムの構造への対応付けを規定する。

- ①ひとつのプログラムはひとつの層の機能のみを実装し、異なる複数の層の機能を含んではならない。
- ②Functionality層は以下の2種類に明確に分離する。
 - (a)アプリケーションのフロー制御機能部分
 - (b)他層もしくは機能層内の自身以外の部品へ機能を提供する関数部分
 これは(a)が(b)を呼び出す構成のみを認める。

3.2 コンポーネントウェア

部品を組み合わせることで、アプリケーションプログラムを構築する仕組みであり、以下の三点によって成立する。

- ・機能の部品化
- ・部品の組み合わせによるプログラムの構築
- ・実行時の部品の再利用及び流用

3.2.1 問題点

- (1)アプリケーションプログラム作成の度に、独自の部品を作成し、これを組み合わせただのではメリットが少なく、アプリケーションプログラマへの負担が大きい。
- また、エンドユーザが自身でシステムの変更ができない。
- (2)部品作成時の規約が無いため、使用するツールやアプリケーションの実行環境等に依存した、特有の部品が作成される。

3.2.2 対応策

(1)共通部品の提供

- ・Presentation、Functionality、Dataの各々について、業務に近い部品を、共通に提供す

る。利用者は予め用意された共通部品を、必要に応じてカスタマイズし、それを組み合わせることで独自のアプリケーションプログラムを作成する。

(2)部品開発用テンプレートと、部品を利用するアプリケーションのテンプレート(フレームワーク)の作成

- ・利用者が必要最小限の処理を記述することにより、部品/アプリケーションの形勢が行えるテンプレート(フレームワーク)を提供する。ただし、これは開発ツールや実行環境に依存してはいけない。

Fig2に、本検討による実現方法に基づいたアプリケーション構築イメージを示す。

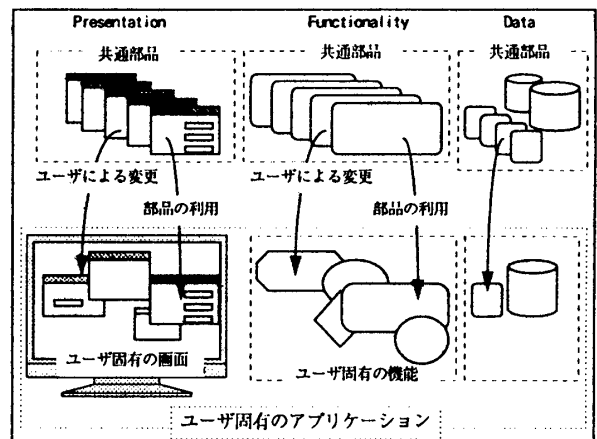


Fig2. 実現イメージ

4 今後の課題

現在、実際のアプリケーションへの試行適用を行っている。今後、評価および結果の反映を行っていく。

参考文献

- (1)甲斐、菅沼 「テンプレートを用いた、使いやすいGUI画面の設計法」 情報処理学会第49回全国大会論文集
- (2)甲斐、菅沼、稗田 「トランザクション環境下でのパネル主導の画面制御方式の実現」 情報処理学会第47回全国大会論文集
- (3)松本、菅沼、黒岩、河合 「3層クライアント/サーバシステムの検討」 情報処理学会第51回全国大会論文集