

部品組立型業務アプリケーション開発ツール

アップギャラリー “APPGALLERY”

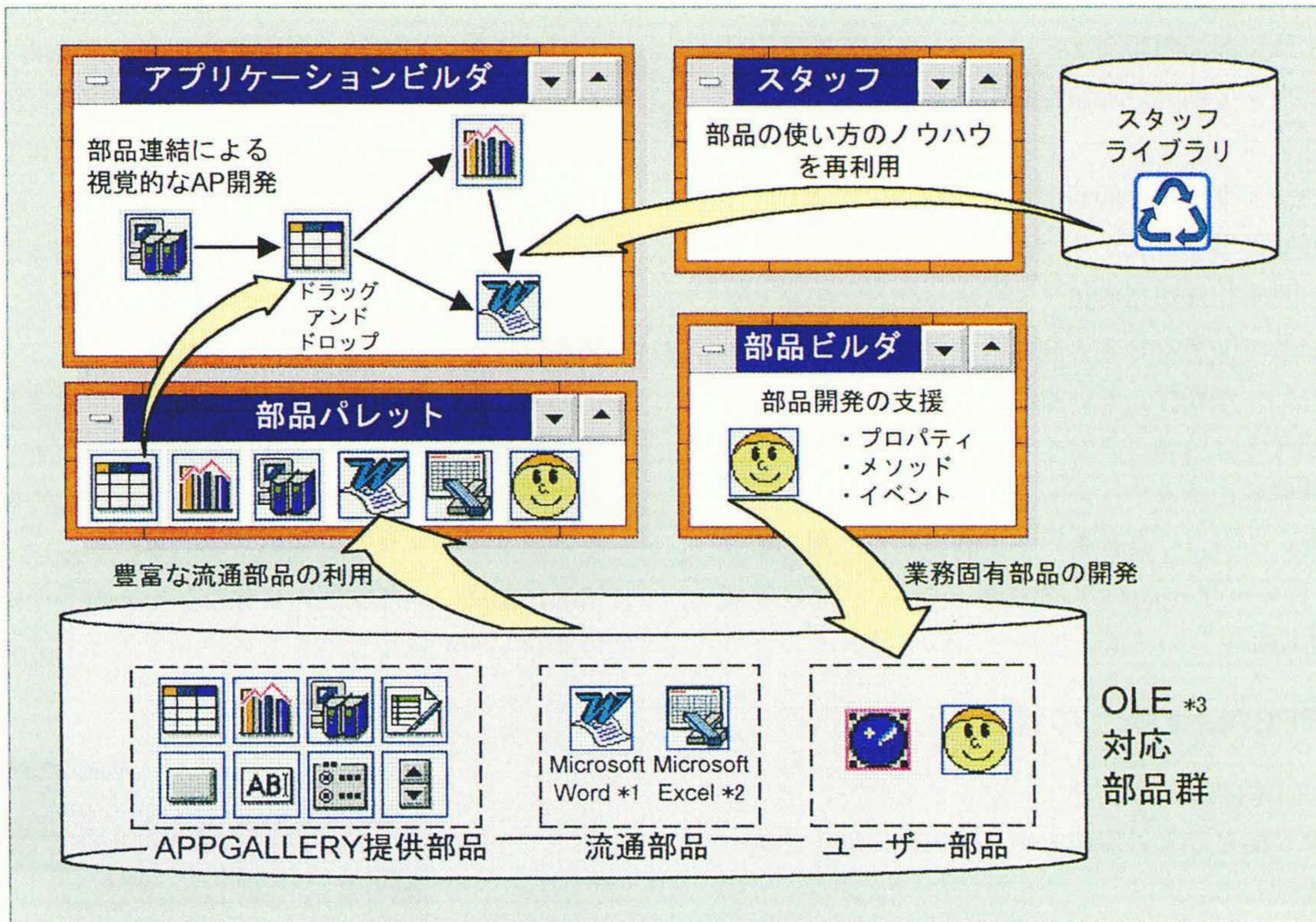
Component-based Visual Programming Environment

富永雅介* Masasuke Tominaga

鶴 秀夫** Hideo Tsuru

原 幸** Miyuki Hara

上村清美*** Kiyomi Kamimura



注：略語説明など AP(Application Program), OLE(Object Linking and Embedding)

*1 Microsoft Wordは、米国Microsoft Corp.の商品名称である。 *2 Microsoft Excelは、米国Microsoft Corp.の商品名称である。 *3 OLEは、米国Microsoft Corp.が開発したソフトウェア名称である。

“APPGALLERY”の構成

APPGALLERYでは、豊富な流通部品を利用した、視覚的なアプリケーション開発を可能にするとともに、業務分野固有の部品を開発し、その部品の使い方のノウハウを再利用するための支援機能を提供している。

計算機環境のCSS(Client Server System)化に伴い、エンドユーザー自身が業務システムを効率よく開発できるツールが求められている。そのため、エンドユーザーによる業務アプリケーションの開発とカスタマイズを容易にする“APPGALLERY”を製品化した。

APPGALLERYでは、ソフトウェアを部品化し、それらの部品を「視覚的につなぎ合わせる」という

方法で業務システムを開発する。さらに、「部品のつなぎ合わせ方のノウハウ」を再利用する機能も提供している。典型的なアプリケーションはコーディングで開発できる。ソフトウェア部品としては、業界標準のプログラム間連携技術“OLE”¹⁾を採用しており、サードベンダが開発した流通ソフトウェア部品を利用することができる。

* 日立製作所 情報・通信開発本部

** 日立製作所 ソフトウェア開発本部

*** 日立製作所 情報システム事業部

1 はじめに

CSS化が浸透し、計算機システムの適用による業務の効率化がさらに進行するとともに、業務アプリケーション開発の生産性をさらに高めるためのプログラミング環境が要求されるようになってきている。そこで、ハードウェア産業では常識となっている「規格化されたインタフェースを持つ部品の組合せ」によるシステム開発方法を、ソフトウェア開発へも適用して生産性を向上させようとするコンポーネントウェアの考え方が現れてきた。

一方、ハードウェアの低価格化と高性能化により、1人1台の計算機環境が広がりつつある中で、情報システム利用者の要求仕様が高度化・個人依存化し、アプリケーション開発が従来のシステム部門主導型からエンドユーザー主導型へと移行している。さらに、エンドユーザー自身が開発またはカスタマイズ可能なアプリケーション簡易開発環境を要求する声も高まっている。

ここでは、このような背景の下で開発した、部品組立型のエンドユーザー向けアプリケーション開発環境“APPGALLERY”の特徴と機能について述べる。

2 “APPGALLERY”の特徴

APPGALLERYは、エンドユーザーでも容易にアプリケーションが開発できる環境とするために、以下に述べる機能を提供している。

- (1) エンドユーザー部門向けの機能として、「視覚的に部品をつなぎ合わせる」という形態でアプリケーションを開発する環境を提供する。さらに、「部品のつなぎ合わせ」を視覚的に誘導・支援する機能を提供することにより、テキストプログラミングなしのアプリケーション開発を可能にする。
- (2) システム部門向けの機能として、業務分野に応じた汎(はん)用的な部品とその部品の利用をガイドする機能を開発するための環境を提供する。これにより、エンドユーザー部門とシステム部門との分業体制による効率的なアプリケーション開発を実現する。
- (3) パーソナルコンピュータ環境でプログラム間連携のための業界標準技術となっているMicrosoft[®]のOLE 2.0を、ソフトウェア部品の規格としてサポートする。これにより、OLEをサポートした流通ソフトウェアを自在に組み合わせたシステム構築を可能にする。

※) Microsoftは、米国Microsoft Corp.の登録商標である。

3 APPGALLERYが提供するAP開発支援機能

3.1 アプリケーションビルダ

アプリケーションビルダは、業務プログラムを、部品(アイコンで表現)とコネクタ(複数のアイコンをつなぐ線)で視覚的に組み立てるエディタである。アプリケーションビルダは、処理内容を視覚的に記述する作業空間としてのキャンバスと、利用可能な部品を格納する部品パレットの二つのウィンドウによって構成される(図1参照)。アプリケーション開発の流れは、(1)使用する部品のアイコンを、部品パレットからキャンバスへドラッグアンドドロップすることにより、部品の実体を生成し、(2)キャンバス上では、生成された部品の実体をコネクタで関連づけるという基本操作で構成される。

部品は、一般のオブジェクト指向言語で定義するクラスに相当し、部品の内部状態を表すプロパティと、部品の内部状態に対して外部から操作を行うためのメソッドをインタフェースとして持っている。さらに、部品の中には、部品の内部状態の変化などを外部へ通知するためのイベントを持つものもある。一方、コネクタは、それ

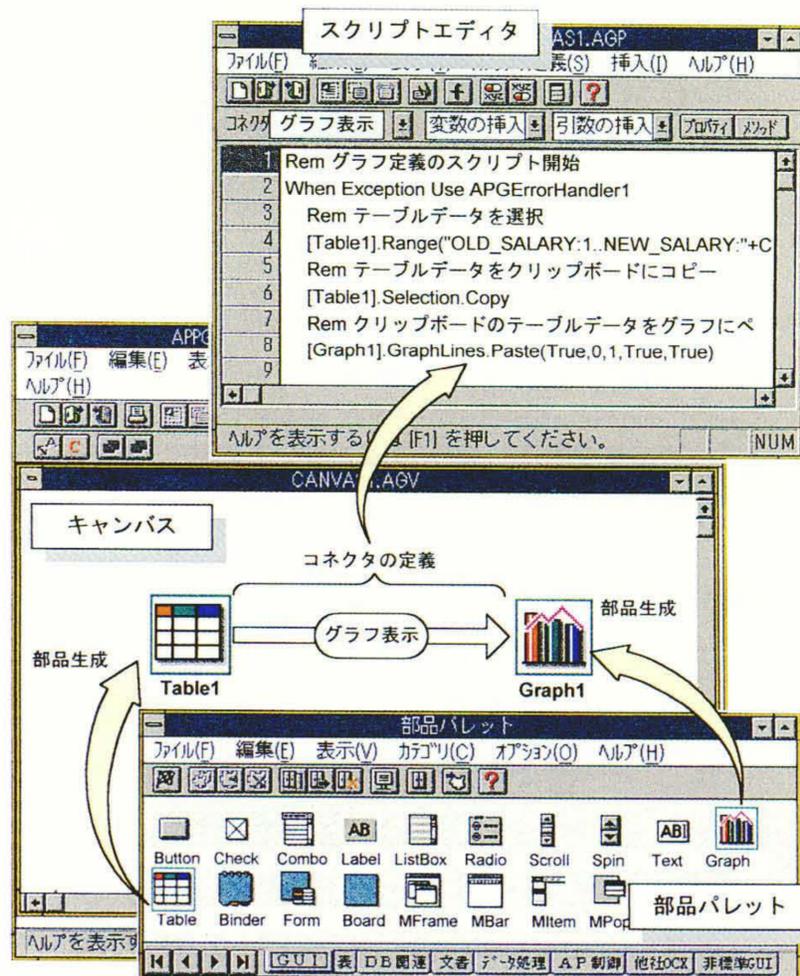


図1 アプリケーションビルダによるアプリケーションの作成
部品パレットから使用する部品をキャンバスにドラッグアンドドロップする。キャンバス上では、複数の部品をコネクタで関連づける。コネクタの処理内容は、スクリプトエディタでテキストコーディングする。

につなげられた部品をパラメータとして起動するプログラムである。コネクタプログラムは、図1に示すスクリプトエディタを用いてコーディングする。言語としては、BASIC(Beginner's All-Purpose Symbolic Instruction Code)を拡張したものを提供している。コネクタの実行制御方式は「イベント駆動型」であり、キャンバス上の部品が発生するイベントによってコネクタを起動する。

以上のようにAPPGALLERYでは、アプリケーションを定義するビジュアル言語として、アプリケーションの画面遷移やマクロな処理の流れが記述できるようなレベルのものを採用し、細かい処理ロジックはテキスト型言語でコーディングを行うという方法を採用した。これは、「おおまかな業務の流れを記述するユーザー(エンドユーザー)と、細かいロジックを記述するユーザー(システム部門ユーザー)は異なる」、「異なるユーザーには異なるツール・言語を提供すべきである」という考え方に基づいている。この結果、キャンバス上に視覚的に定義されたプログラムを見ると、エンドユーザーでもその処理概要を読み取ることができるという大きな利点を得ている。

3.2 スタッフ機能

スタッフとは、部品を接続するコネクタの作成を支援するための対話型プログラムである。APPGALLERYでは、ユーザーが記述する頻度の高いコネクタを生成するためのスタッフ群をスタッフライブラリという形態で提供している。

キャンバスでは、部品のアイコンを別の部品のアイコンの上にドラッグ アンド ドロップで重ね合わせた際に、その部品間に関連づけるコネクタを生成するためのスタッフがライブラリから検索される。ユーザーは、検索されたスタッフの中から、部品間に設定したい処理に適したスタッフを選択・起動し、さらに、スタッフが提示するダイアログにコネクタを起動するイベントや処理のパラメータなどを設定するだけで、目的の処理を行うコネクタを自動生成することができる(図2参照)。

このスタッフ機能を用いることにより、典型的な処理に関しては、コーディングレスでアプリケーションを作成することが可能になる。

なお、スタッフは、APPGALLERYアプリケーションの一種であり、適用業務分野固有のアプリケーション開発手順や部品の使い方のノウハウをユーザー個別のスタッフとして開発し、再利用することが可能である。

3.3 部品ビルダ

APPGALLERYでは、適用業務分野に応じた部品を開

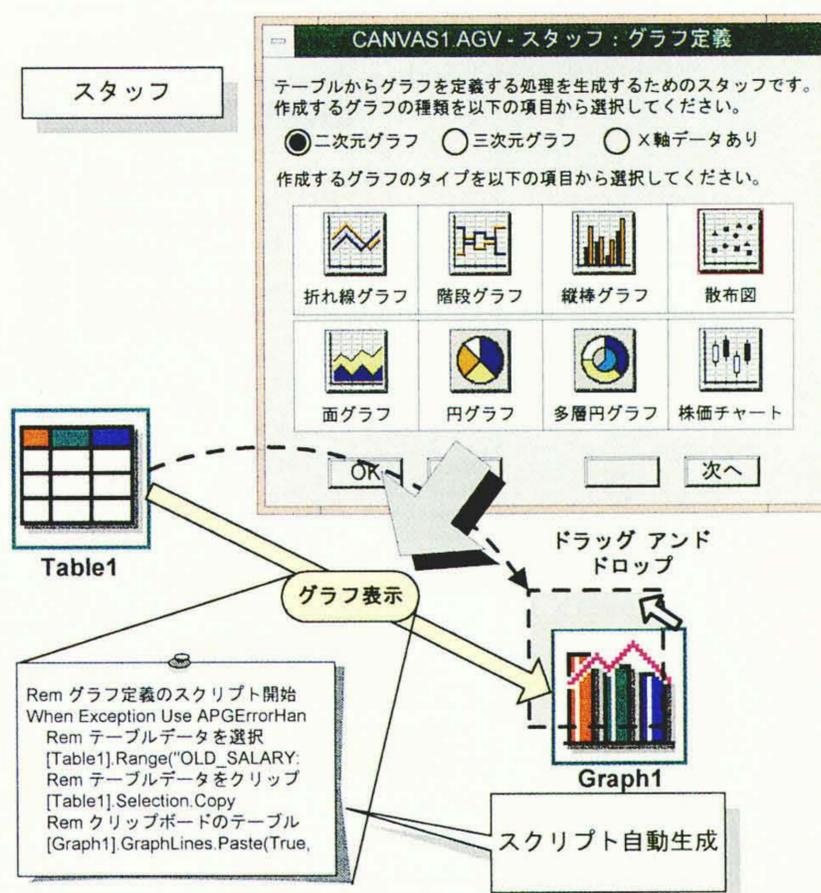


図2 スタッフを用いたコネクタの生成

キャンバス上で部品のアイコンに他の部品のアイコンをドラッグ アンド ドロップで重ね合わせると、利用可能なスタッフが検索される。選択したスタッフの表示するダイアログにこたえるだけで、部品間のコネクタが自動生成される。図の例では、グラフ部品アイコンにテーブル部品アイコンを重ね、テーブルのデータを基にグラフを作成・表示するためのコネクタを生成するスタッフを起動している。

発するのためのツールとして部品ビルダを提供している。部品ビルダでは、部品のインタフェースとして、プロパティ、メソッド、イベントを定義する。メソッドの処理内容は、スクリプトエディタで定義する(図3参照)。

部品ビルダでは、アプリケーションビルダで作成したキャンバスそのものを部品化することも可能である。キャンバスを部品化したキャンバス部品は、アプリケーションビルダでプログラムしたキャンバスに、「部品」としてのインタフェースを定義するだけで作成できる。このキャンバスの部品化機能により、アプリケーションの再利用が促進される。さらに、キャンバスを階層的に定義することにより、複雑な業務プログラムを視覚的に見やすく定義することを可能にしている。

4 APPGALLERY組込み部品

APPGALLERYでは、アプリケーションのGUI(Graphical User Interface)を構築するための基本部品群とともに、ビジネスアプリケーション向けの部品群を提供している(表1参照)。これらの部品はすべて、OLEオートメーション機能をサポートした部品であり、コネ

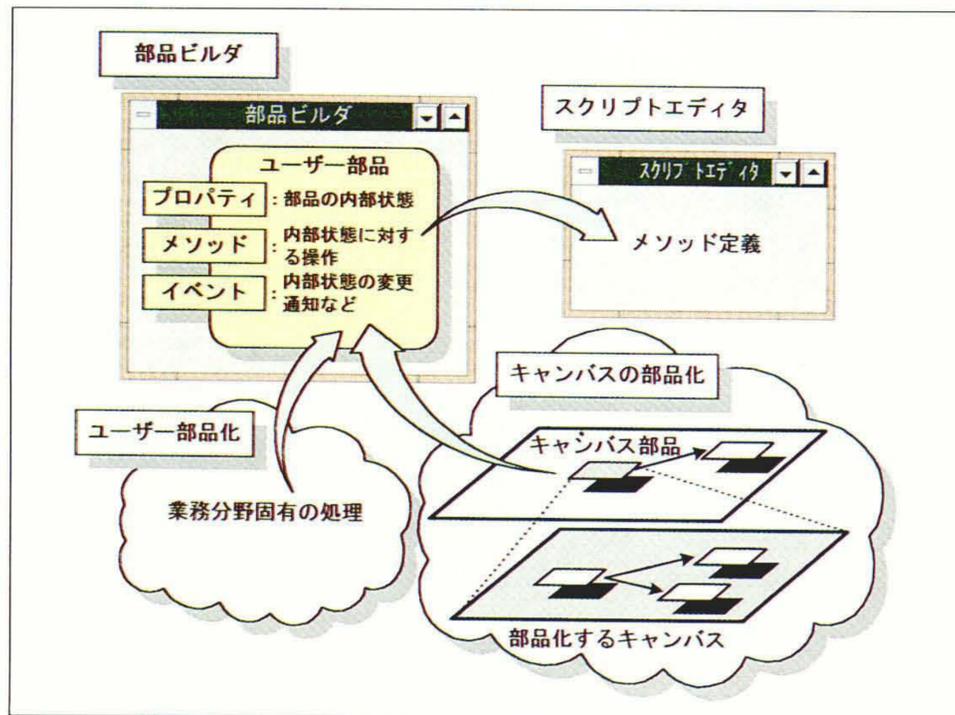


図3 部品ビルダを用いたユーザー部品の作成

部品ビルダでは、部品の内部状態をデータとして格納するプロパティ、部品の内部状態に対して外部からの操作を可能にするメソッド、さらに、部品内部状態の変化などを外部に通知するためのイベントを定義する。メソッドの処理内容は、スクリプトエディタを用いて定義する。また、部品ビルダでは、アプリケーションビルダで作成したキャンバスを部品化することも可能である。

表1 APPGALLERY組込み部品一覧

APPGALLERYでは、開発環境のほかに典型的なビジネスアプリケーションの開発に必要とされる部品群を提供している。

カテゴリー	部品名	主な機能	OLE コンテナ	OLE オートメーション	OLE カスタムコントロール
基本部品	フォーム	GUIの台紙, DB連携機能	○	○	—
	コントロール	ボタンやラベルなどGUIの基本構成要素(約10種類)	—	○	○
ビジネス 向け部品	テーブル	テーブル型データの加工	—	○	○
	グラフ	12タイプのグラフ作成	—	○	○
	レポート	帳票作成, 集計機能	○	○	—
	DBアクセス	QBEなどを用いたDBアクセス	—	○	—
	マルチメディア	動画, 静止画, 音声の表示・再生	—	○	○

注：略語説明

DB(Database), QBE(Query by Example)

クタプログラムの中から操作可能なメソッドとプロパティを公開している。また、OLEカスタムコントロールの規格を満たす部品は、GUIの構成要素としてOLEコンテナであるフォームに埋め込んだ形で用いる。OLEカスタムコントロール部品は、GUIを通したユーザーの操作などに連動してイベントを発行する機能を持ち、コネクタの実行を制御することができる。

5 おわりに

ここでは、視覚的なソフトウェア部品の組立によってアプリケーション開発を容易にする“APPGALLERY”

の基本機能について述べた。

APPGALLERYでは、業界標準のソフトウェア部品アーキテクチャOLEを採用することにより、豊富な流通部品を利用したアプリケーション開発を可能にする。さらに、システム部門や独立の部品ベンダが、業務分野ごとの部品や部品の使い方を支援するスタッフを開発し、エンドユーザー部門やSI(System Integration)ベンダが、その部品とスタッフを駆使して視覚的に業務プログラムを開発する、分業体制による効率的なエンドユーザーコンピューティングを実現するためのツール群を提供している。

参考文献

1) K. Brockschmidt: INSIDE OLE 2, Microsoft Press (1994)
 [邦訳 長尾訳: INSIDE OLE 2, アスキー出版(1995)]

2) J. Udell: オブジェクト指向の新展開コンポーネントウェア, 日経バイト, pp.277~289(1994-6)