

ビジュアル開発環境「Visual Tools」における
データベース定義と画面定義の共用化

7J-8

小野孝志 富田典幸 石橋昌幸 築山昇 長沢喜美男 工藤隆良
三洋電機(株) 情報通信システム研究所

1. はじめに

携帯型端末は、パソコンと比較して、画面サイズ、メモリ、演算処理能力に制限が多く、パソコン上で実現されているような強力なビジュアル開発環境は利用できない。そのため、C言語やアセンブラ言語でプログラム開発を行っており、非常に効率が悪かった。そこで、我々は、上記のような制限のある携帯型端末上でも動作可能なインタプリタとインタプリタ用アプリケーションの作成を容易にするビジュアル作成ツールの構築を行っている。インタプリタとビジュアル作成ツールを合わせて、ビジュアル開発環境「Visual Tools」という。

「Visual Tools」は、アプリケーションの構成要素を「部品」として定義しており、「部品」を組み合わせることで、携帯型端末用アプリケーションを作成することができる。

今回、ターゲットを業務用途のデータ入出力分野に絞り、「Visual Tools」を実用的開発環境とするための開発を行った。具体的には、実行時の部品構造のバケット化[1]、データベース定義と画面定義の共用化、処理速度アップのための平面構造化、および、バケット・インタプリタの開発を行った。本稿では、「Visual Tools」の概要、および、データベース定義と画面定義の共用化について述べる。

2. Visual Tools

システム構成を図1に示す。

「Visual Tools」は、大きく開発側と実行側に分けることができる。

開発側の内部構造は、画面定義部(データベース定義部と共用)、処理定義部、バケット生成部、シミュレーション部から成り立っている。開発側で作成したプログラムをバケット構造に変換し、実行側に渡す。

バケットはアプリケーションを端末で実行するための定義データの集合である。生成されるバケット構造はツリー構造となっており、基本的に、複数の「画面」と「画面」に付随する「部品」から成り立っている。

実行側は携帯型端末であり、バケット構造を解釈し、実行するインタプリタがROM(Read Only Memory)として搭載されている。

一般の開発環境では、データベース定義と画面定義は別々に行っている。これは、データベースが、様々な用途(画面)に利用されているため、あるいは、不定長のデータへの対応を行う必要があるためである。しかし、そのために操作性が良くない上に、処理が複雑になってしまっている。一方、携帯型端末においては、利用範囲が限られ

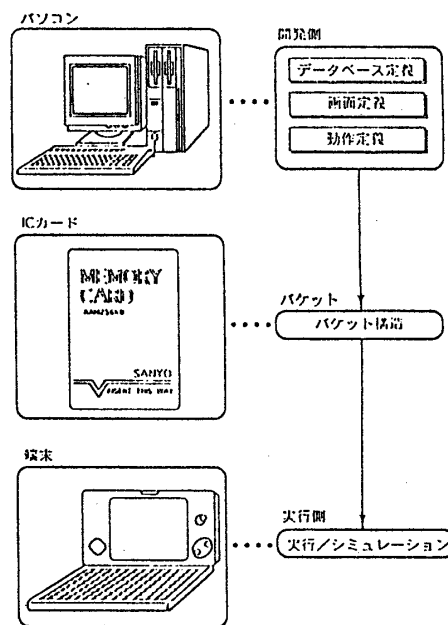


図1 システム構成

ており、そのような制約はない。その上、データ入力/表示項目とデータベース項目は1対1であることが、ほとんどである。そこで、データベース定義と画面定義の両定義を共用化することにした。

3. データベース定義と画面定義

業務用途のデータ入出力分野におけるアプリケーション構成要素としては、データ保存用データベース、入出力としてのユーザインタフェース、データベース検索、および、文字やボタンなどの画面の表示/動作がある。これらを「部品」として定義した。データベース定義も画面定義も、「部品」を用いて定義する。

3. 1 データベース定義

データベース定義は「定義用ウィンドウ」に対し、テキスト、日付、時間等の「文字入力部品」を貼り付け、定義する。例を図2に示す。図2は設備というデータベース名で店舗名、品名、品番等の各項目を定義した例であり、右側に定義内容を示す。

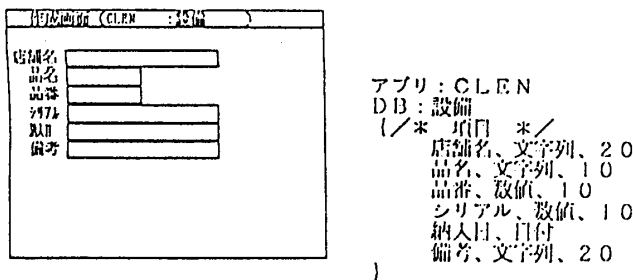


図2 データベース定義例

3. 2 画面定義

「画面」に対し、「文字入力部品」やボタンなどの「処理部品」の貼り付け、「画面」を分かり易くするための文字・図形の表示を指示することにより、画面定義を行う。例を図3に示す。図3

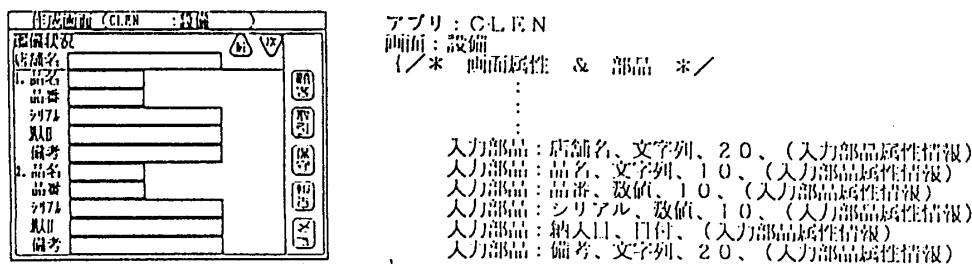


図3 画面定義例

は、図2で定義したデータベースを画面上に表示する場合の一例を示している。右側にデータベース関係の画面内情報を示す。

3. 3 共用化

図2、3からわかるように、データベース定義と画面定義は、同じ「部品」を使用している。両定義の違いは、画面定義側には「文字入力部品」以外の「部品」があることと部品につける名前が異なることである。そこで、両定義を共用するため、「部品名」の共通化とデータベース定義内容の抽出対応を行った。これにより、作業を軽減できると共に、データベース定義項目と入出力項目の不一致のようなミス無くし、容易に両定義を行うことが可能となる。つまり、図2のようなデータベース定義を行う必要がない。

4. おわりに

今回、データベース定義と画面定義の共用化等の「Visual Tools」を実用的システムとするための開発を行った。これにより、データ入出力を扱うアプリケーションの開発を容易に、しかも、短期間に行うことが可能となった。2つのアプリケーションをサンプルとして開発を行った結果、どちらも仕様解析からデバッグまで、約2週間という短期間で開発することができた。

今後は、上記サンプル以外のアプリケーション開発を行い、実用性を高めると共に、グラフィックスや音声処理への対応を行い、マルチメディアへの展開を図っていく。

参考文献

[1] 石橋他: "携帯型情報端末における中間言語形式", 平成5年電気学会全国大会論文集, pp.1750(1993)