

開発環境

関 武夫、吉田 和樹、金地 克之、野村 圭祐、渡瀬 慎一郎

(株)東芝

1 APF 開発環境とは

C Solution APF を使ったアプリケーション開発では、アプリケーションの実装は、一からコードを記述するのではなく、ビジュアルな開発環境下で基本機能を実現するベースコンポーネント（以下 BC）を組み合わせ、そこに業務に固有の必要最小限の処理コードを記述したカスタムコンポーネント（以下 CC）を差し込んでいくという手順になる。

BC とは、C Solution APF が提供する基本機能を表す少数個の部品である。これら BC が行う内部処理は C Solution APF によって予め提供されているので、開発者は APF 開発環境を使ってコンポーネントを組み合わせるだけで、すばやくアプリケーションの構築を行なう事ができる。

CC とは、個々のアプリケーションに固有の処理を行う部品である。BC がアプリケーションの基本機能を実現した後、BC でカバーできない処理を CC として作成し、BC から呼び出して利用する。

このようにして、APF で開発した BC と CC から成る一連の業務処理ルーチンを APF の外側から呼び出すことにより、アプリケーションシステムを構築する（図 1）。

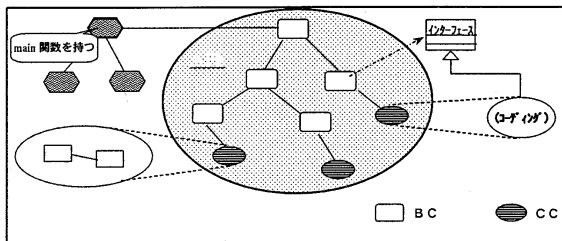


図 1. APF 利用時のアプリケーション構造

C Solution APF によって C/S 型構造の業務システムを開発する場合は、開発組織を、GUI（クライアントアプリケーション）開発、CC 開発、そして BC の組み合わせの三つの部隊に分けて行えるため、開発の並行性が向上する。また、サーバアプリケーション処理手順の変更には、APF 開発環境上での BC の組換えで素早く対応することができる。そして品質の高い BC を組み合わせるため、アプリケーションは高い信頼性を備えることができる。

2 APF 開発環境を利用した開発手順

C Solution APF の一つである業務トランザクション処理(Accounts)を利用した施設予約システムの構築を例にして、APF 開発環境を使ったアプリケーション開発手順を説明する。

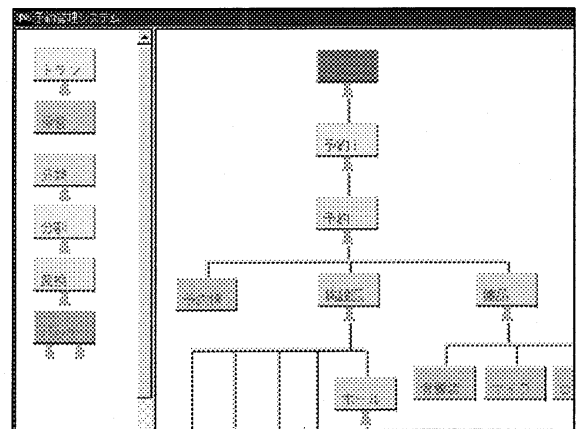


図 2. APF 開発環境の画面

図 2 は APF 開発環境上で Accounts によるアプリケーションを開発する時の画面である。画面の左側のパレットには、Accounts の六つの BC が並んでおり、

Component-based Framework Technology (C Solution APF) : Visual Editor

Takeo Seki

Toshiba Corporation

3-22, Katamachi Fuchu-shi, Tokyo 183-8512 Japan

これらをドラッグ&ドロップすることで、右側のワークエリアに BC を配置する。次に、配置したそれぞれの BC に「予約トランザクション」や「301 会議室」といった業務的な意味を持つ名前をつける。そして、配置した BC 間の入出力の関係線を引くと、予約トランザクション処理の骨格が出来上がる。

予約受け付け情報の生成や、先約確認といった、BC の組み合わせだけでは実現できないアプリケーションに固有のロジックは、CC として Java により実装し、関係する BC に登録することによってアプリケーション全体として実行することができる。

図 3 は、Java のクラスとして実装された CC を、BC に登録する画面である。



図 3. カスタムコンポーネントの登録

Accounts の場合、BC に入力されるデータは、複数の入力項目で構成される。たとえば、「301 会議室」として配置された BC の入力項目は、「利用日」「利用開始時刻」「利用終了時刻」である。このような、入力項目等の、BC に関する基本的な設計情報の設定も、APF 開発環境上で GUI 操作により行うことができる (図 4)。

このように、アプリケーションの処理の流れは BC の組み合わせで行ない、アプリケーション固有の機能は CC の実装により開発できる。従って、CC は処理の流れに依存しない再利用性の高い機能部品として作りこむ事ができる。また、BC の組み合わせや基本的な項目の設定を変える事で、処理の流れに変更があっても素早く対応できる。例えば、新たな業務

機能が付加された場合でも、それを実現する CC を実装し、それを登録する BC をモデルに追加することで、アプリケーションの処理の中に容易に、その機能を組み込むことができる。



図 4. 入力項目の設定

BC の配置、項目設定、関係付け、そして CC の登録が終わると、APF 開発環境のソースコード自動生成機能を使って、そのままコンパイルして動作するアプリケーションの Java ソースコードが出力される。このアプリケーションを業務処理層としてサーバ上に展開し、別途開発した GUI 層とデータベース層と組み合わせる事で、アプリケーションシステム (この例では施設予約) が完成する。

また、APF 開発環境では、開発中のアプリケーションをサーバ上で実行時オブジェクトを生成し、実行させることもできる。この機能により、実行時の動作を開発時に確認できるため、より生産性の高いアプリケーション開発が可能になる。

3 まとめ

この施設予約システムの構築例に見られる通り、APF 開発環境は優れた GUI 開発環境と基本機能、そして処理フロー・業務機能分離の構築技法により、品質が高く、仕様変更に対応できるサーバアプリケーションの RAD (高速アプリケーション開発) を強力にサポートすることができる。