

携帯電話用ソフトウェアプラットフォーム

Platform Software for Mobile Phones

井上 栄

■ INOUE Sakae

携帯電話用ソフトウェアは、事業者サービス及びメーカー独自サービスの充実・拡張により、ますます開発規模が増大している。このため、ソフトウェア開発の効率化と品質の確保が求められている。

東芝は、この要求に応えるため、アプリケーションの移植性向上、保守性向上、オープン性向上を目的として、“東芝フレームワーク構想”を立ち上げた。現在、当社のフレームワークは、自社開発のアプリケーションだけでなく、導入ソフトウェア(プラットフォーム及びアプリケーション)を含んだ形で成長・進化している。当社のフレームワークと導入ソフトウェアを連携動作させる技術が重要である。

The volume of software for mobile phones continues to increase with the expansion and evolution of both carrier-specific services and device vendor-specific services. As a result, it is necessary to develop software more efficiently while maintaining software quality.

To meet these requirements, Toshiba has constructed the "Toshiba Framework Plan" composed of the following three elements: improvement of porting, improvement of maintenance, and improvement of openness. The Toshiba Framework is evolving together with not only Toshiba applications but also other software platforms provided by software vendors. It is important to design software mechanisms that control both the Toshiba Framework and the software platforms of software vendors in a coordinated manner.

1 まえがき

近年、携帯電話機を利用したサービスが豊富になってきている。静止画、動画、音楽、テレビ(TV)電話などマルチメディアを中心としたサービスが急激に増加している。これは、従来から存在する音声通話、Eメール、コンテンツの閲覧といったベース機能にマルチメディア機能を付加することにより実現している。これらマルチメディア機能を持つ携帯電話向けサービスとしては、音楽、動画といったコンテンツ配信サービスが中心である。コンテンツには、著作権というものが存在し、これをめぐる新たな機能が追加されている。

サービスの複合化も進み、Eメールの中にURL(Uniform Resource Locator)記述があれば、ブラウザを起動し表示するなどがそのよい例である。このような連携の機能は今後ますます多くなっていく。更に、音楽を聴きながらEメールをするといった同時に二つ以上の機能が動作するようなユーザー利用の場面も多くなるだろう。

今や携帯電話は、1990年代のパソコン(PC)やワークステーション(WS)より高性能になってきている。CPUクロックは150MHzを超え、メモリ容量はROMとRAMを合わせて96Mバイトを超える機種が多くなっている。磁気ディスク装置(HDD)を搭載した機種も登場し、まさしくウェアラブルコンピュータの名にふさわしい姿となっている。

携帯電話の機能とサービスの進歩に呼応するように、これ

らを実現するためのソフトウェア規模と複雑さは増加の一途をたどっている。携帯電話1台のソフトウェア量は、実行コードで500万ステップを超える機種も登場してきている。この増大に伴い、いかにソフトウェアの品質を高めていくかが大きな課題となってきている。開発量の増大には開発人員の増加で対応せざるをえないため、スキルの高いメンバーや経験豊かなメンバーだけで作業を進めていくわけにはいなくなった。このため、ソフトウェアの開発工程を標準化し、作業の効率化や品質の安定化を図ろうとする動きも盛んである。また、開発作業を標準化し効率化を進めていくためには、ソフトウェアプラットフォーム、モデル化されたアプリケーション、アプリケーション開発環境を含む開発プロセスを構築していくことがたいせつである。

ここでは、開発プロセスのトータル概念である“東芝フレームワーク構想”について述べるとともに、ソフトウェアプラットフォームにかかわるキー技術を紹介する。

2 ソフトウェア開発規模の増大

2.1 開発ステップ数の増加

携帯電話の機能やサービスの多様化及び複雑化は、ソフトウェアの増加を生み出している。東芝が商品化した主要機種のソフトウェア開発のステップ数も年々増加している(図1)。

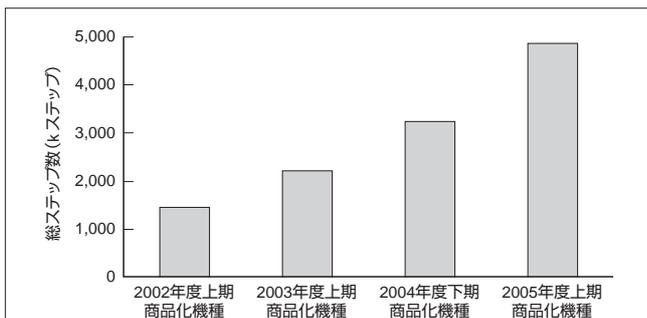


図1. 東芝の主要携帯電話用ソフトウェアの開発規模推移 — 東芝が2002年～2005年に商品化した主要携帯電話のソフトウェアの開発規模を示している。カメラ、動画、音楽をはじめとする各種マルチメディアサービスの進化・発展とともに、ソフトウェアの開発規模が年々増加している。

Total number of steps of software for Toshiba mobile phones

2.2 データ量の増加

機能やサービスの多様化を加速している背景には、液晶ディスプレイ(LCD)、カメラなどの入出力装置及び無線通信方式の進化、コンテンツサーバ類の充実などネットワークインフラの発展・進歩がある。これに伴い、ソフトウェアが扱うデータ量も飛躍的に増加している。

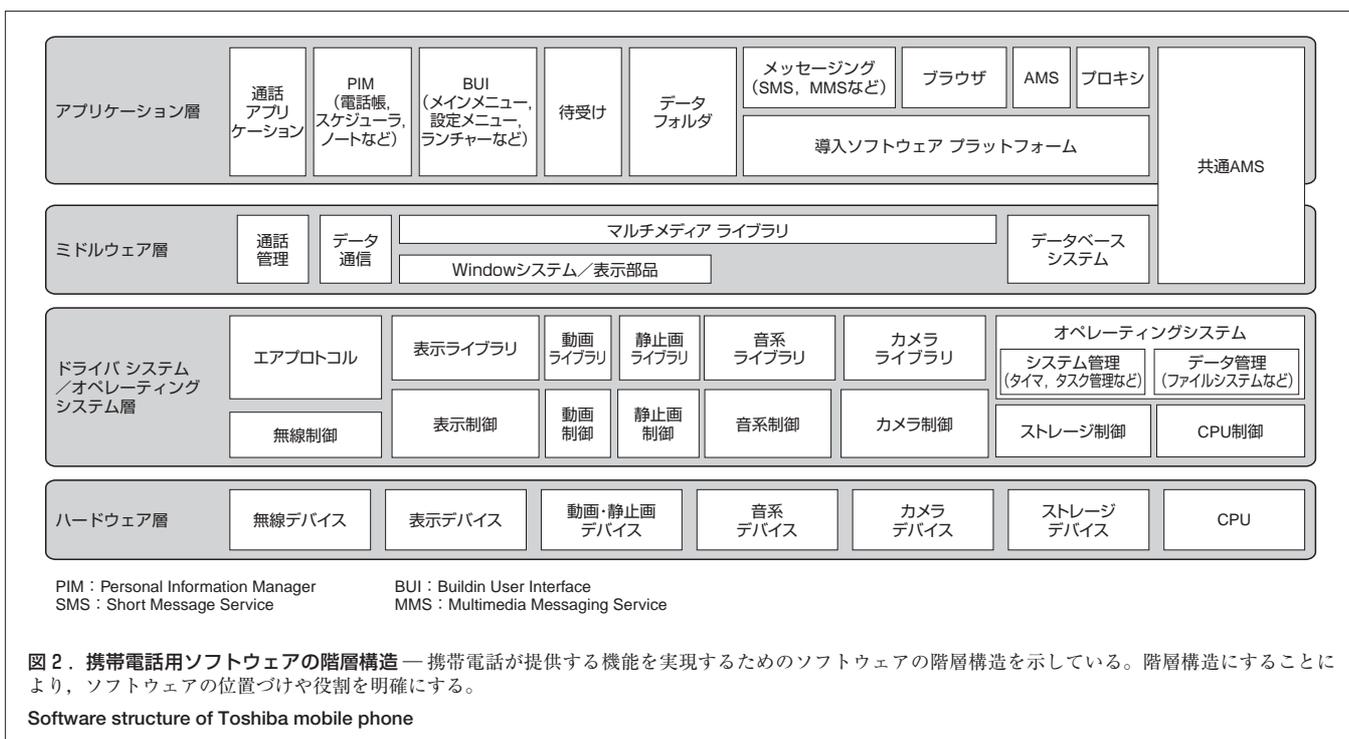
3 複雑化するソフトウェア

3.1 ソフトウェアの階層化

機能やサービスが複雑化すると、それらをつかさどるソフト

ウェアは膨大な量となる。そのため、ソフトウェアのモジュール化、階層化を行い、開発の効率化及び生産性向上を目指す。ソフトウェアは、通常3階層に分離できる。ハードウェアを最下位層に位置づけると、下位からドライバシステム／オペレーティングシステム層、ミドルウェア層、アプリケーション層となる(図2)。

- (1) ドライバシステム／オペレーティングシステム層
ドライバシステムは、ハードウェアを抽象化し上位ソフトウェアから利用しやすいインタフェースを提供する。オペレーティングシステムは、システム管理とデータ管理のシステムに分解できる。システム管理は、タスクのディスパッチやスケジューリングといったタスク管理機能と、イベントハンドリングやタイマなどのリソース管理機能を提供する。データ管理は、ファイルシステムやアクセスメソッドなどデータ操作にかかわるインタフェースを提供する。近年は、マルチメディア系の処理も多くなり、各種コーデックにかかわるエンコード／デコード処理とのインタフェースを提供する。
- (2) ミドルウェア層
アプリケーション層に必要な表示システムインタフェース(Windowシステム、表示部品)、データハンドリングインタフェース(データベースシステム、各種メソッド)を提供している。近年は、マルチメディア系の処理も多くなり、カメラインタフェース、各種コンテンツプレーヤ(MP3(MPEG-1(Moving Picture Experts Group-phase 1) Audio Layer 3)など)インタフェースを提供する。



(3) アプリケーション層 ユーザーインターフェースをつかさどるソフトウェアである。各種設定系メニュー、メッセージング、ブラウザなどがこの層に属する。

3.2 各種データのデータベース化

扱うデータ件数やデータ量の増加に加え、データに対するアクセスも多様化してきている。単純な読み込み・書き込み処理だけではなく、検索処理、ソーティング処理など複雑な処理も増えている。これらを一般的なファイルシステムとこれに関係するメソッドだけで処理するには、アプリケーションの負担が大きい。この負担をなくすため、データベースシステムとこれに関するメソッドをミドルウェア層に準備し、アプリケーションに提供している。アプリケーションは、検索処理やソーティング処理といった複雑な処理を単純な記述で実現し、パフォーマンスも確保している。

3.3 アプリケーション連携サービス

携帯電話の各種サービスは、アプリケーションと呼ばれるソフトウェアで実現している。アプリケーションは、メッセージングやブラウザに代表される。これらアプリケーションは、連携して動作するサービスが主流である。また、Eメールの裏受け処理、ながら機能、マルチアプリケーションサービスなど複雑なサービスが登場してきている。これらは、それぞれのアプリケーション単独で制御するものではなく、全アプリケーションを統合管理する共通AMS (Application Management System) の下で制御される(図3)。

4 ソフトウェア プラットフォーム

4.1 東芝フレームワーク構想

4.1.1 フレームワークとは アプリケーションプログラムの規模増大のため、携帯電話用ソフトウェアは増大の

一途をたどっている。何機種も並行開発し、かつ予定した開発期間で開発を完了させるため、ソフトウェア開発効率の向上を目的とした活動がなされている。この目的のため、当社は次のようなフレームワーク構想を立ち上げ、携帯電話用ソフトウェアの開発を実施している。

- (1) フレームワーク導入の目的 ソフトウェア開発効率の向上を目的としている。このために次の3点が必要である。
 - (a) アプリケーションの移植性向上
 - ・設計書の再利用率を高める。
 - ・ソースコードの再利用率を高める。
 - (b) アプリケーションの保守性とメンテナンス性向上
 - ・ソフトウェア品質を高める。
 - ・仕様追加による機能拡張を容易にする。
 - (c) アプリケーションのオープン性向上
 - ・新規開発者投入時のコストを削減する。
 - ・アウトソーシングの自由度を高める。
- (2) フレームワークの構成要素
 - (a) ソフトウェアプラットフォーム
 - ・ミドルウェア層に位置づけられ、再利用重視のソフトウェア構造を構築する。
 - ・アプリケーションの動作原理(起動・終了、連携動作)を確立する。
 - (b) 開発環境
 - ・デバックツールにより開発を支援する。
 - ・ロジック検証ツールにより開発を支援する。
 - (c) アプリケーション標準サンプル
 - ・アプリケーションのモデル化(アプリケーション構造の統一)を実施する。
 - ・実装サンプルソフトウェアを提供する。

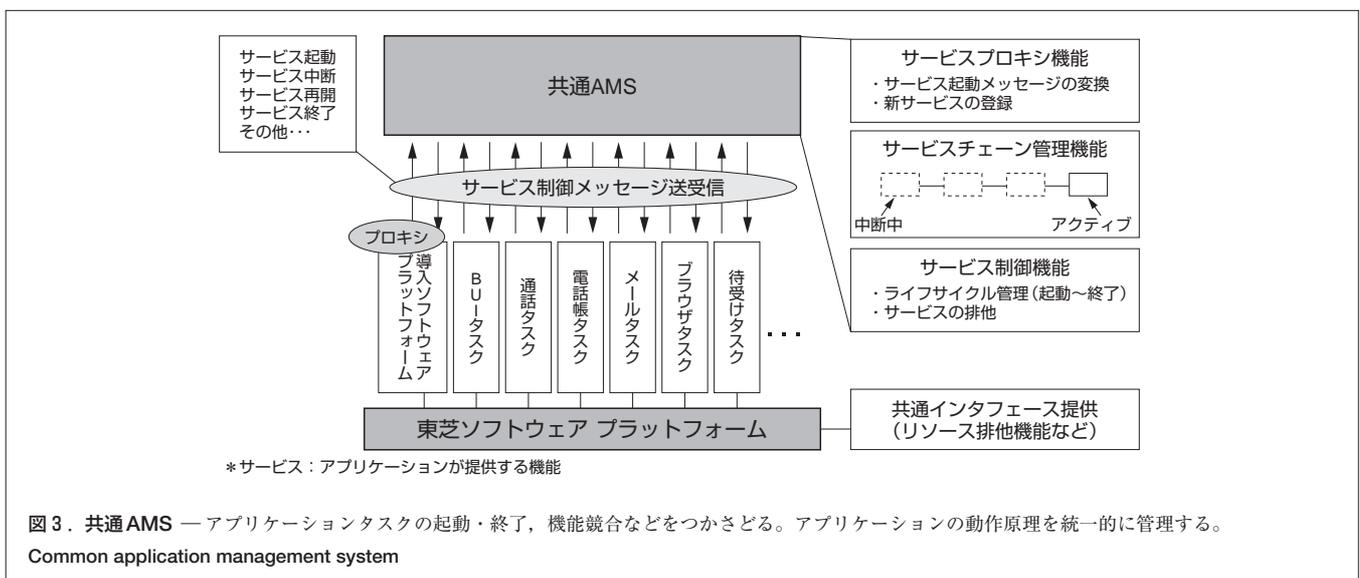


図3. 共通AMS — アプリケーションタスクの起動・終了、機能競合などをつかさどる。アプリケーションの動作原理を統一的に管理する。
Common application management system

- (d) 標準的設計プロセス
- ・設計手順を整理する。
 - ・設計ガイドを充実する。
 - ・設計ツールを利用する。

4.1.2 フレームワーク適用効果

(1) コード流用率 フレームワーク導入機種が増えるとともに、ソフトウェアの流用率が上がっている(図4)。特に注目すべきは、アプリケーション層の下位に位置するミドルウェア層(ソフトウェアプラットフォーム)が各機種で共通化できたことによる効果が現れている。

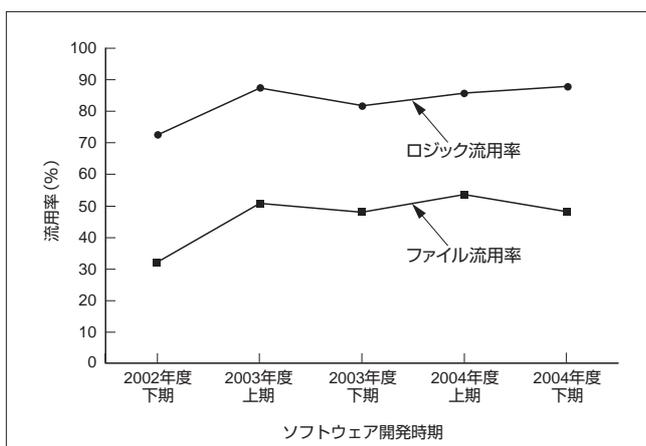
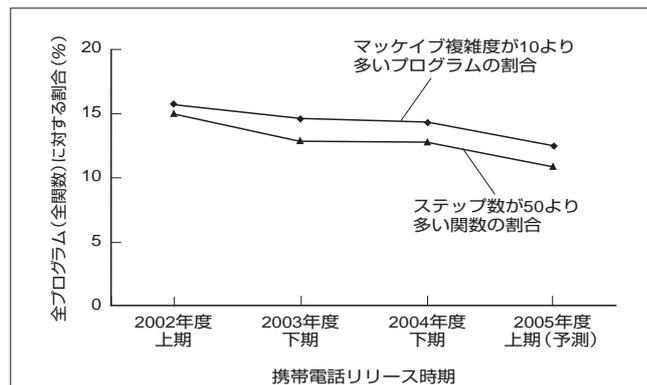


図4 東芝の携帯電話用ソフトウェアの流用率 — 2002年下期の東芝フレームワーク適用第1号機をリリース後、ソフトウェアの流用率(ロジック流用率、ファイル流用率)が向上した。事業者サービスや東芝独自サービスの進歩・進展に伴ってソフトウェアの開発規模が増加し、流用率が低下するのが通常であるが、東芝フレームワークの効果やいろいろな工夫により流用率は安定している。

Reuse rate in Toshiba of software for mobile phones

- (2) プログラムの複雑度低減 ソフトウェアプログラムの複雑度を示す指標として、マッケイブ複雑度がある。これは、理解性(可読性)が良いソフトウェアプログラミングをしているかを示す指標である。2002年のフレームワーク導入後のソフトウェアにはその効果が現れ、複雑度が年々低くなっている(図5)。これは、保守性と理解性が増していることを示しており、品質確保にも有効と判断できる。
- (3) アプリケーションのモデル化と開発環境 アプリケーションは、表示処理部分、処理実行部分、制御部分の三つに分割できる。この考え方をベースにアプリケーションのモデル化を行い、理解しやすいソフトウェア構造を実現する(図6)。このモデル化したアプリケーションを開発するため、開発支援ツールと当社の携帯電話用ソフトウェアをシミュレートするPC上の開発環境を開発した。



マッケイブ複雑度: プログラムの複雑度の指標であり、グラフ理論により算出される。プログラム内部の条件関係などをグラフで表現し、グラフ中の数値化データ(辺の数など)から複雑度を算出している。

図5 東芝の携帯電話用ソフトウェアの複雑度推移 — マッケイブ複雑度が10より多いプログラム、又はステップ数が50より多い関数は理解性が落ち(複雑度が高く)、それぞれ10以下、50以下が望ましいと言われていた。2002年以降、これらの割合が年々減少傾向にあり、複雑度が低くなっていることがわかる。

Trend of McCabe number in Toshiba of software for mobile phones

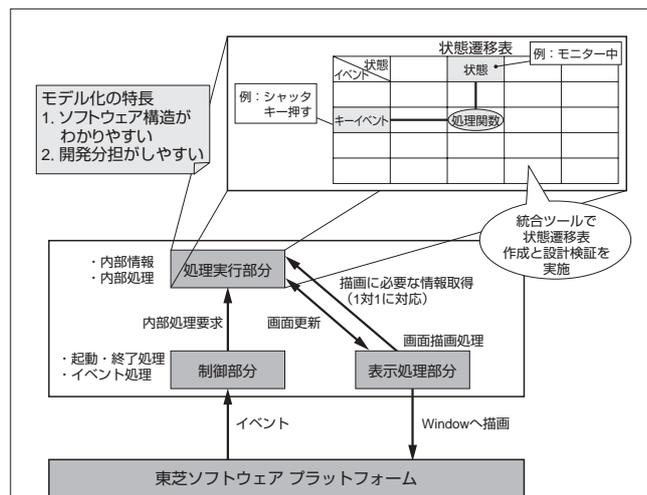


図6 モデル化されたアプリケーションプログラム — アプリケーションをモデル化することによって、理解しやすいソフトウェア構造を実現する。

Modeling of application program

4.2 導入ソフトウェアプラットフォーム

4.2.1 通信事業者の考え方と傾向

通信事業者は、サービスの統一を図る、サービス適用機種を同時に立ち上げる、携帯電話の品質を維持する、などの目的のため、アプリケーションの統一を図りつつある。特に、コンテンツ配信サービスについては、コンテンツが動作する環境を統一することは重要である。ブラウザや各種プレーヤ(MP3プレーヤなど)の統一が図られている。メッセージングについても同様である。これら主要アプリケーションの統一化のため、これらが動作するソフトウェアプラットフォームを統一化する方

向をも示している。この統一するソフトウェアプラットフォームの候補としては、JavaTM (注1)、BREW[®] (注2)、Symbian^(注3) /Series60^(注4)などがある。

4.2.2 OMA など標準化動向 通信事業者によっては、OMA (Open Mobile Alliance) などの標準化に準拠したソフトウェアをサービスの基本ととらえる通信事業者もでてきた。このため、標準化動向をキャッチし、これに準拠したソフトウェアを独自開発するか導入していくかを常に考えていかなければならない。

4.3 東芝フレームワークと導入ソフトウェアプラットフォームの連携

4.3.1 連携動作の課題 当社の携帯電話用ソフトウェアには、フレームワークとこの上で動作するモデル化されたアプリケーションプログラムが存在する。この環境の中に、JavaTM、BREW[®]といった新たなソフトウェアプラットフォームを導入していく。JavaTM、BREW[®]の上で動作するアプリケーションは、標準化されたメッセージングやブラウザが中心である。これら導入ソフトウェアと当社が過去から保持しているソフトウェア資産を有機的に結合し、携帯電話サービスを実現していく必要がある。このため、導入アプリケーションどうしの連携はもとより、“フレームワーク+当社資産のアプリケーション”と“導入ソフトウェアプラットフォーム+導入アプリケーション”の連携を実現するソフトウェアメカニズムが必要である。

4.3.2 連携動作解決のために 導入ソフトウェア上のアプリケーション連携のために、導入ソフトウェアプラットフォーム上にAMSを開発し連携手順を確立する。このAMSと当社のフレームワーク上の共通AMSを連携させる。

(注1) Javaは、米国 Sun Microsystems, Inc.の米国及びその他の国における登録商標又は商標。

(注2) BREWは、米国 QUALCOMM Incorporatedの米国及びその他の国における商標又は登録商標。

(注3) Symbianは、Symbian Ltd.の登録商標。

(注4) Series60は、Nokia Corporationの登録商標。

この二つのAMSの連携を図る目的で、両者をつなぐプロキシプログラムが存在する。プロキシプログラムは、AMS間のコマンドとデータの流れをつかさどる(図2, 図3)。

5 あとがき

携帯電話サービスは、今後も充実・拡張の一途をたどることが予測される。これに伴い、ソフトウェアの開発規模もますます増大していき、ソフトウェアの品質確保もよりいっそう重要になる。また、通信事業者サービスにタイムリーに対応して製品を市場に投入すること、及びユーザーが満足する独自のサービスやアプリケーションの充実も併せて実現していく必要がある。

これらを満たすためには、自社だけで開発していたのでは時間も工数も不足しがちとなる。したがって、何を導入ソフトウェアとしていくか、独自アプリケーションはどのようなものにしていくかの選択がたいせつとなっている。また、導入ソフトウェアプラットフォームの登場は、携帯電話用ソフトウェアの開発プロセス自体も変える可能性を示している。

当社は、今まで培ってきた技術やノウハウとソフトウェア資産を有効に活用していかなければならない。これらの資産と導入ソフトウェアプラットフォーム、及び導入アプリケーションのコラボレーションにより、エンドユーザーにとって魅力的なサービスを提供する携帯電話を開発していく。



井上 栄 INOUE Sakae

モバイルコミュニケーション社 モバイルデベロップメントセンター モバイルソフトウェア設計部長。携帯電話ソフトウェアの開発業務に従事。情報処理学会会員。Mobile Communication Development Center