

## IV-2 「Java 言語を利用した機種互換可能なアプリケーションを開発できる

### MFP/LP 向けのソリューションプラットフォーム」に関する技術解説

(採用機種：一部機種を除いた 2004 年秋モデル以降全機種)

大石 勉 南雲 拓

株式会社リコー グローバルマーケティング本部 NSS 事業センター

#### 1. はじめに

オフィス業務の生産性向上のため、OA機器に対して、「ワークフロー」、「セキュリティ」、「TCO (トータルコストオブオーナーシップ)」、「コンプライアンス」、「環境」に焦点を当てた顧客価値の創造が求められている。

リコーでは、1999年に「ドキュメントハイウェイ」構想を打ち出した。これは簡単にいえば、エンドユーザーが、いつでも、どこからでも、ネットワークに接続されているさまざまな機器から、必要な情報を、最適な形で取り出し、やり取りするための仕組みを提供するというものである。即ち、ユーザーが今いる場所や利用するオフィス機器の違い、音声や動画といったデータ形式の違いを意識せずに、自由に情報をやり取りするための環境を提供するものである。

そしてこの「ドキュメントハイウェイ」構想において、デジタル複合機(MFP)やレーザープリンター(LP)といった機器の役割に大きな変化が要求された。これは、複数の複写機、プリンターを効率的にネットワークで連携させるプリンティングソリューション、ファイルサーバーによる集約的な保管、検索、出力のためのシステムを構築・運用するドキュメントソリューション、ワークフローにそったユーザーインターフェースのカスタマイズなどである。

例えば、MFP の機能をワークフローにそって使用するために、従来では、予め組み込まれたユーザーインターフェースを使用せざるを得ないため、煩雑な操作が必要な場合があった。これに対し、ワークフロー毎に抽出された機能がワンアクションで操作可能なユーザーインターフェースが要望されていた。

リコーでは、このような要望を実現するためのソリューションプラットフォームとして、C言語アプリケーション開発・実行環境を2000年から提供してきた。

さらに、2004年には、開発効率をさらに向上させ、より多くのアプリケーション開発技術者の参画を可能とするために、Sun Microsystems 社が開発した Java をベースとした機種互換可能なアプリケーション開発・実行環境を提供した。

#### 2. ESA の概要

##### 2. 1. ESA (Embedded Software Architecture)

リコーの MFP/LP は、共通のプラットフォーム上で各機能(コピー、プリント、スキャン、FAX)が動作する構成になっている。ESA は、こうした共通のプラットフォーム上で機種間および後継機での互換性を維持しつつ、MFP/LP の多彩な機能を生かした高度なシステム開発を可能とする Java言語を利用したアプリケーション開発プラットフォームである。

ESA は、C 言語の開発ツールをさらに発展させ、開発言語にネットワークとの親和性が高い Java を採用しており、OS への非依存、異機種での高い互換性といった Java の特徴を活かすことで、例えば、オフィス機器と社内システムの連携も図りやすくなり、開発期間の短縮や開発コストの大幅な低減を可能としている。

##### 2. 2. 機能

ESA では、Java ME (Java Platform Micro Edition) の CDC (Connected Device Configuration) 及び FP (Foundation Profile) という構成で実装している。

MFP/LP クラスは、MFP/LP の機器制御を行うことができる。MFP/LP クラスは、C言語の API に比べ、複雑な処理手続きを意識せずに効率良くアプリケーションが開発できるように機器制御における様々な特異性を、オブジ

ェクト指向の MFP/LP クラス群として組み込んで提供している。

ESA は、以下のような特徴を持つ。

- (1)ESA で開発した Java アプリケーションは、ESA 対応のすべての MFP/LP 機器で使用することができ、ESA クラスライブラリは、機種間で共通である。
- (2)機器の操作パネル、スキャン、プリント、FAX、ファイル保存等の機能を利用することができる。
- (3)HTTP(S), FTP, SMTP 等のネットワークプロトコルを提供する。また、アプリケーションが自身でオープンソースの jcifs を使用することによって SMB も利用可能となる。
- (4)豊富な Java クラスライブラリをアプリケーションに組み込むことで、様々な機能を容易に実現できる。

2. 3. ソフトウェア構成

ESA は、C言語の開発ツールをベースに構築されている。ESA のアプリケーションには、Xlet と Servlet の2つの Type がある。Xlet Type は、MFP/LP 上の操作パネルにユーザーインターフェースを表示するアプリケーションモデルである。Servlet Type は、MFP/LP にネットワーク接続された Webクライアント上にユーザーインターフェースを表示するアプリケーションモデルである。Xlet Type 及び Servlet Type は、アプリケーションのライフサイクルを管理しやすい仕様となっている。アプリケーションは、それぞれ独立にリソースを管理され、複数の Java アプリケーションを同時に実行することが可能である。Fig. 1 は、ESA のソフトウェア構成図である。

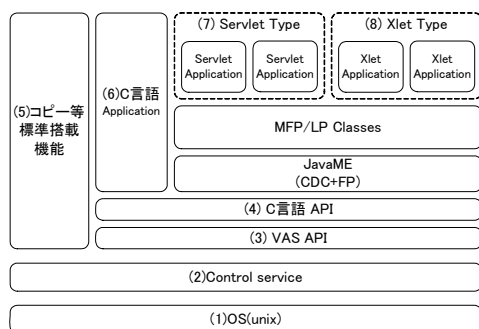


Fig.1 ESA Software Configuration.

ソフトウェア構成の各部分の概要を Table 1 に記す。

Table 1 ESA Software Configuration Overview.

部分	概要
(1)OS (unix)	オペレーティングシステム
(2)Control Service	機器制御用の機能群
(3)VAS API	機器サービスのラッパー層
(4)C 言語 API	C 言語の開発ツールが提供する機種互換ライブラリ群
(5) コピー等標準搭載機能	コピー/スキャナ/プリント/FAX などの標準搭載機能
(6)C 言語アプリケーション	C 言語 API を利用したアプリケーション
(7)Servlet Type	ESA アプリケーション (Servlet Type)
(8)Xlet Type	ESA アプリケーション (Xlet Type)

3. ESA の特徴

3. 1. MFP/LP クラスとデバイス仮想化技術

MFP/LP クラスは、MFP/LP の機器制御を行うことができ、以下のようなサービスを提供する。

(1) ユーザーインターフェース制御

「パネル制御 (パネルサービス : Panelパッケージ)」等により、ユーザーインターフェースの構築を可能とする。主なユーザーインターフェースのアイテムとして、Button, Buzzer, Icon, Label, LED, Pattern (線, 矩形を描画), Soft Keyboard, Window 等を提供している。

(2) スキャン制御

「スキャン (スキャンサービス : Scanパッケージ)」を利用することにより、原稿を MFP のスキャナを利用して読み取り、読み取った画像データを MFP 本体に TIFF や JPEG 等の標準的な画像フォーマットでの保存ができる。

(3) プリント制御

「プリント (印刷サービス : Printパッケージ)」を利用することにより、MFP のスキャナを利用して紙原稿から読み取った画像データや、ネットワーク上のクライアントから受け取った画像データの印刷を可能とする。

(4) FAX 制御

「FAX (FAXサービス : Faxパッケージ)」により、MFP のスキャナを利用して紙原稿から読み取った画像データなどを FAX 送信することが可能である。また、FAX でのデータ受信も可能である。

### (5) アプリケーション管理

アプリケーションに対して以下の制御を行うことができる。

- ・インストール
- ・アンインストール
- ・起動
- ・停止
- ・アプリケーション情報取得

それぞれのサービスは、システムから見ると仮想のアプリケーションとして振る舞うラッパー層である VAS (Virtual Application Service) を経由して実現している。VAS は機器本体側に置かれるサービスであり、提供する機能の仮想化を行っている。

例えば、表示に関する仮想化技術としては、インストールされるアプリケーションの情報として、そのアプリケーションの画面設計で想定されている画面サイズ情報を付加し、例えば HVGA 用に設計された画面を有するアプリケーションを WVGA、若しくは SVGA の操作部画面の機器で動作させた場合でも、表示部サイズの違いに応じて自動的に拡大して表示するようになっている。

### 3. 2. 機能のバンドル化

ESA Ver. 2 以降ではプラットフォームに OSGi が導入されており、Ver. 5 以降では、追加可能な機能をバンドル単位で行うことができるようになった。

バンドルは、追加可能なオプションバンドルと ESA リリース時に組み込まれ複数のバンドルから利用されるサービスバンドルに区別されている。オプションバンドルの追加・削除 (インストール・アンインストール) のほか、リリース時にあらかじめ組み込まれたサービスバンドルの起動・停止も行うことができる。これにより利用しないサービスバンドルを停止することで、メモリリソースの有効利用などを行うことが可能である。

### 3. 3. ライセンス管理

#### 3. 3. 1. ESA プラットフォームのコピー及び改ざん防止

ESA プラットフォーム本体は、認証付きの SD カードに格納されており、SD カードをコピーしても起動しない。ESA を動作させるには、ESA プラットフォーム本体が格納された SD カードが、機体に挿入されている必要がある。これにより、SD カード単位で ESA プラットフォームのライセンス管理を行っている。

#### 3. 3. 2. アプリケーションのコピー及び改ざん防止

ESA のアプリケーションが、リコーにより発行される認証データにより認証を受けた後、第三者によって不正にコピーされると、アプリケーションインストール時に、システムにより不正なアプリケーションと見なされインストールされない。

また、インストールおよびインストール後の起動時において、改ざん防止確認を行っている。これによりアプリケーションに改変や置き換え等の不正が認められた場合、そのアプリケーションは起動されない。上記の認証処理は、リコーから発行される認証データと電子署名を利用して実現される。

アプリケーション認証データを使用することで、MFP/LP 上で動作するアプリケーションが機器を破損するようなものではないこと、また正規のベンダにより作成されたアプリケーションが不正にコピーされたものではないことを証明できるようにしている。

この認証は ESA のプラットフォームが行う。プラットフォームは、アプリケーションのオブジェクトに組み込まれたアプリケーション認証データを使用し認証を行う。MFP/LP のリソース保護やアクセス制限などを行う認証データは、電子署名を行う際にアプリケーションのオブジェクトに組み込む。

電子署名は、アプリケーションを構成する全ての JAR ファイルに対して行う必要がある。

### 3. 4. アクティベーション

近年、ネットワークを利用して、ソフトウェアアプリケーションのパッケージ単位での販売、サービスや課金が行われている。これは、ユーザーからみると必要なサービスを必要とときに必要な分だけ利用できるためコスト面で優位であり、また必要なサービスの検索・導入が容易であるため利用面でも優れている。

ESA では Fig. 2 に示すように、ネットワーク上の PC が Activation Client となり、複数の MFP/LP のアプリケーションのライセンスを管理することができるようになっている。ユーザーがアプリケーションを使用する際には、Activation Client が Activation Server と通信し、ライセンスを取得して MFP/LP に導入する。ライセンスの解除なども Activation Client から可能となっている。また複数のプラグインから構成されるアプリケーションなど、複数のライセンスを必要とするアプリケーションにも対応している。

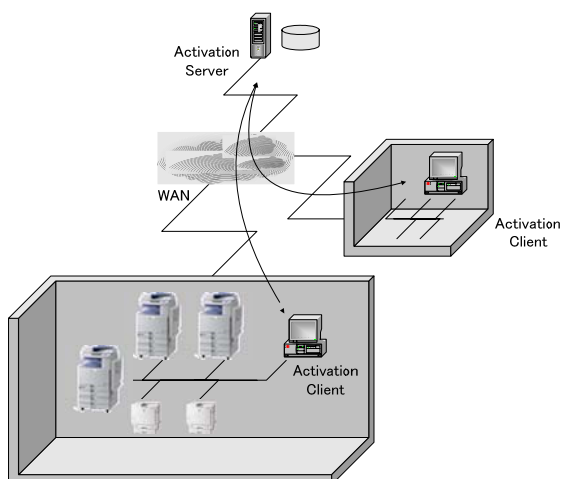


Fig. 2 Activation.

### 3. 5. アプリケーション開発環境

#### 3. 5. 1. SDK (Software Development Kit)

アプリケーションを開発するための環境として、SDK が提供される。SDK は、各種マニュアル、サンプルアプリケーション、エミュレータ、開発サポートツール

などを含んでいる。

#### 3. 5. 2. エミュレータ

ESA では、開発支援ツールとして、Windows 版および Linux 版のエミュレータを提供している。エミュレータは、MFP と同じ操作環境 (Fig. 3) を PC 上に構築することで、実機を使用せずに開発を始めることができ、開発作業の大半を行うことができる。実機の使用は、最終的な評価に限定できる。本エミュレータは、実機の動作環境を正確に再現させるため、実機と同一の動作環境 (Java ME) を使用し、表示や機能のエミュレーションに PC 上の Java (Java SE) を利用することにより、実現されている。

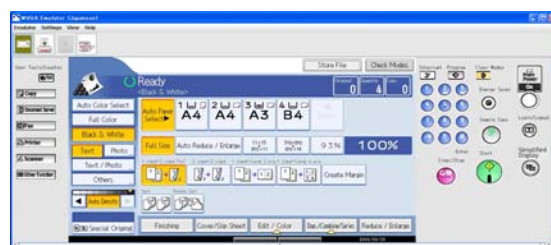


Fig. 3 Emulator Screen

#### 3. 5. 3. アプリケーション開発サポートツール

ESA ではアプリケーション開発を補助するためのツールとして、アプリケーション操作レコーダーとアプリケーション互換性チェックツールを提供している。

##### (1) アプリケーション操作レコーダー

Fig. 4 は、Xlet アプリケーションに対するユーザー操作の記録・再生と、画面遷移の検証を行うツールである。ツールの起動・停止などは Web クライアントから行う事ができる。本ツールを利用することで、テストの効率化や障害発生状況把握が容易となる。

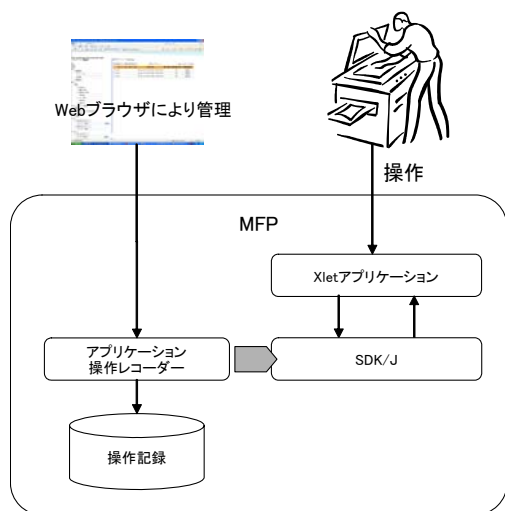


Fig.4 Application Operation Recorder.

(2) アプリケーション互換性チェックツール

アプリケーション互換性チェックツールは、アプリケーションと、その動作環境となる ESA の各バージョンや対象機器との間の API レベルでの整合性を静的に解析し、レポートするツールである。本ツールは、コマンドラインやプログラム開発環境(Fig. 5)から利用でき、ESA の各バージョン間や機種間のアプリケーションのポーティングを容易にする。

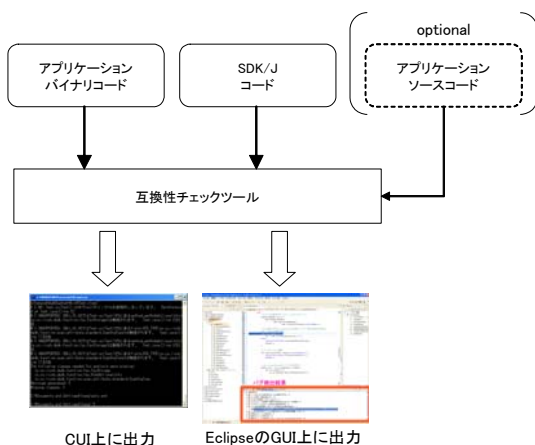


Fig.5 Application Compatibility Check Tool.

3. 6. マルチプラットフォーム対応

MFP/LP は、ほとんどの機種において共通のプラットフォームをもち、その上で稼動する ESA は、機種間・後継機での互換性を維持してきた。一方で、一部のローエンド機器や、OEM 製品など、異なるプラットフォームを

もつ機種も存在する。このような機種に対しても ESA をサポートし、機種互換の幅を広げることで、既存の ESA アプリケーションのリソースを有効に活用する試みがなされている。

ESA は、マルチプラットフォーム対応のため、各種プラットフォームに応じて実装が可能な構造となっている。各プラットフォームに依存する処理は、ESA とネイティブの部分の中間層にあたる Multi Platform Porting Layer に実装される (Fig. 6) 。 Multi Platform Porting Layer は、実装形態を特定していない。このため、各種プラットフォームが持つ API に応じた実装を行うことにより、さまざまな機器上で ESA を動作させることができる。

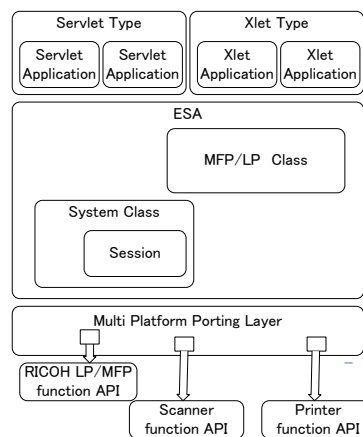


Fig. 6 Multi-Platform Architecture.

マルチプラットフォーム対応への研究テーマとして、OEM 提供されている機器に対し、ESA をサポートし、いくつかの ESA アプリを同機器に搭載させる試みが行われた。対象となった機器のスペックは下記の通りである。

- OS :VxWorks PCD2.0(WindRiver)
- CPU :Tx4937\_MIPS 300Mz(Toshiba)
- Memory :64MB for Java(Total 128MB)

本件では、同機器に複数の Xlet アプリを搭載させ、基本機能についての動作を確認した。性能面についても、JavaVM の高速化、キャッシュの利用などにより、CPU クロックで倍程度の上位機種と比較しても、アプリの性能を3割程度の低下に押さえることができた。一方、課題として、同機器のハードウェア・OS・ファンクション API の制約、取得可能な機器情報の制限などにより、既存の ESA と完全な互換性を確保することが難しいことや、ま

た、PC や外付けBox のように、外部機器上で ESA をサポートした例もあるが、それぞれの電源状態(ON/OFF/省エネ等)が異なり制御が複雑になるという課題が明らかになった。

しかし、イベントのエミュレートや、搭載するプラットフォームがもつ API 機能を組み合わせたり、ネイティブ側の機能実装による対応などにより、各種プラットフォームへの対応が、技術的に可能であることがわかった。

### 4. おわりに

ESA というソリューションプラットフォームにより、新たな価値を容易に MFP/LP に組み込むことが可能となり、サードベンダーを巻き込んだ、お客様に近い場所でのソリューションの開発が可能となり、市場の要望を反映したより価値の高いソリューションを提供できるようになった。現在では 120 社以上のベンダーと協力し、リコーの MFP/LP の付加価値を創出し続けている。

現在、多くのソリューションが全世界で提供されているが、類似したものも多く、ソリューションの品質を均一化する必要がある。例えば、特定の機能のある一定の使い方に統一するような施策が必要である。

MFP/LP に対しては常に新たな機能が要求されている。ソリューションとして追加した機能を新たな機種標準機能あるいは API として提供するには、互換性の維持やテストなどに多くの工数が必要である。SDK は API の集合であり、これを利用したソリューションは本体機能とのリソース競合が必ず発生する。現在では量産開始までに ESA に必要な機能やソリューションのためのメモリリソースなどをあらかじめ確保し、すべて組み込んで出荷しているが、OEM 製品などではそれも難しい。メモリリソースの監視機能と組み合わせることで、出荷後にソリューションに必要な機能を動的に組み込むことができれば、MFP/LP だけでなく、様々な Embedded System に対してソリューションプラットフォームを組み込むことができるようになる。

### 参考文献

1) JSR-000927 Java TV API 1.1

2) JSR-000291 Dynamic Component Support for Java SE

禁 無 断 転 載

2009年度「ビジネス機器関連技術調査報告書」“IV—2”部

発行 2010年4月

社団法人 ビジネス機械・情報システム産業協会 (JBMIA)

技術委員会 技術調査小委員会

〒105-0003 東京都港区西新橋三丁目25番33号 NP 御成門ビル

電話 03-5472-1101(代表) / FAX 03-5472-2511