

# Javaプログラムの基礎と JavaOS for NCsの利用環境

## ○Java 言語プログラムについて

Java 言語で記述されたプログラムの形態には、大きく分けて「Java アプレット」と「Java アプリケーション」がある(図1)。

Java アプレットは、Java 言語プログラムとしてポピュラな形態で、HotJava や Netscape などのJava 対応のブラウザで実行される。

Java アプレットが実行される時、アプレットとその実行に必要なパッケージや各種ファイル(HTMLファイル、データ・ファイルなど)が、WWWサーバやインターネット/LANを介して、Java OS for NCsの作業メモリ DRAM内)にロードされる(セキュリティの管理は、ブラウザで行われる)。

一方、Java アプリケーションは、ブラウザなしに独立したプログラムとしてインタプリタ Java 仮想マシンと呼ばれる)で実行される。

Java アプリケーションは、ローカル・ファイルとして格納する必要があり、実行される時に、JavaOS for NCsの作業メモリ DRAM内)にロードされる。いわゆる、アプリケーション・ソフトと呼ばれるものについては基本的にこの形態で作成する必要がある。なぜなら、大きいプログラムになると、ネットワークからのダウンロードに時間がかかり、WWWサーバからダウンロードしてくるJava アプレットの場合は、ローカル・ファイル・システムにアクセスできなったり、ダウンロード元のサーバ以外にアクセスできないといった制限があるためである。

Java アプレットとJava アプリケーションのプログラム記述上の違いを簡単にいうと、Java アプレットは java.applet.Applet クラスのサブクラス(またはそのサブクラスを含む必要があり、一方、Java アプリケーションは public static void の main メソッドを含む必要がある(Java アプレットの場合は、この main メソッドがブラウザに存在することになる)。

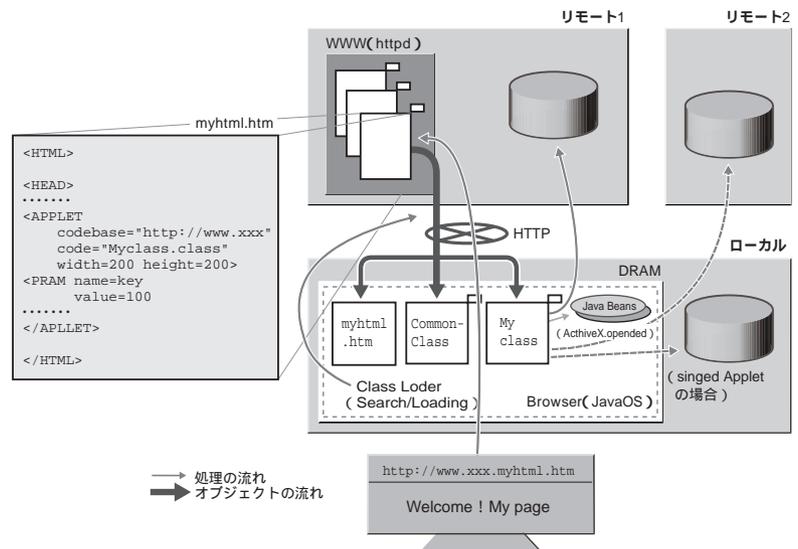
形態としては、ほかにRMIやServletなどがあるが、JavaOS for NCsの説明には関係ないので、省略する。

## ○Java プログラムの開発の進め方

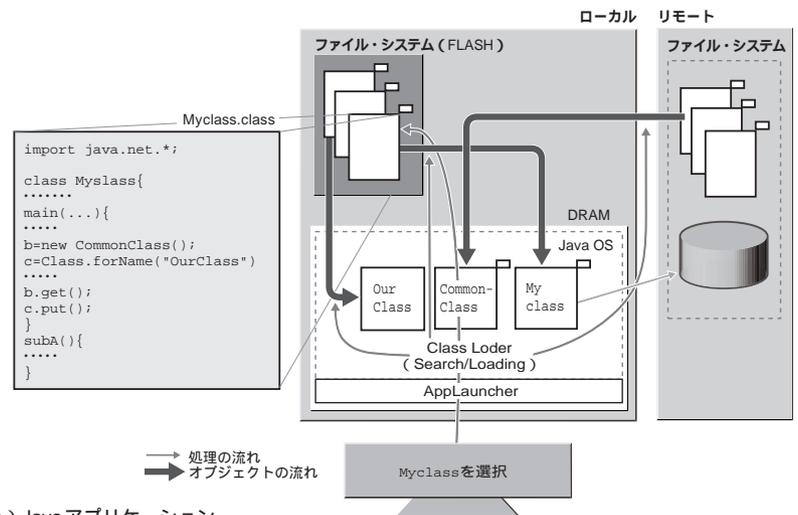
Java プログラムの開発には、Sun Microsystems社から無償提供されているJDK (Java Development Kit)、または、市販のJava統合開発環境が必要になる。Javaは実

行形式がプラットフォームに依存しないので、プログラム開発環境をプラットフォームごとに選ぶ必要がなく、パソコンでもワークステーションでも入手しやすい開発マシンとツールで作成することができる。この点が、他言語での開発と大きく違うところである。

重要なことは、JDK1.1レベルのJava APIで記述されることである(現在のJava OS for NCsのバージョンはV1.1だが、こ



(a) Java アプレット



(b) Java アプリケーション

【図1】Javaプログラムの実行形態

れに対応するのはJDK1.1.4)。

なお、J-StarterKit for SPARCliteには、Javaのプログラムを開発する環境が含まれていない。したがって、開発が(デバッグも含めて)完了したプログラムをJ-StarterKit for SPARCliteで実行することになる。

作成されたJavaアプレットまたはJavaアプリケーションは、Javaコンパイラ(JDKの場合は、javacコマンド)で翻訳すると、実行体であるクラス・ファイルが作成される。そして、クラス・ファイルと、HTMLファイルやGIF、JPEG、オーディオなどのデータ・ファイルなど実行に必要なすべてのファイルを用意する。

ここまでは、J-StarterKit for SPARC-liteがなくても作業が可能である。

### ○セットアップ

JavaOS for NCsは、ネットワーク・コンピュータ(NC)をベースにしたOSである。したがって、評価などで利用する場合は、このシステムを管理するためのサーバ・システムを用意する必要がある(図2)。現在、サーバ・システムとしては、Solaris2.6以上が必要である。

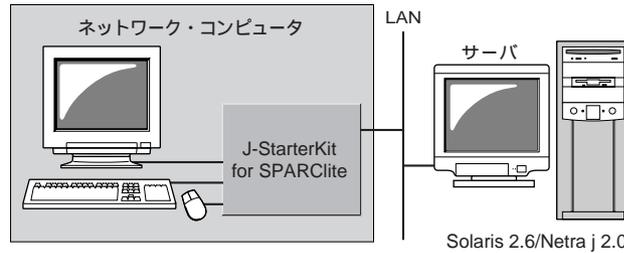
また、管理ツールとして、Netra j 2.0(日本サン・マイクロシステムズ製)が必要になる。これは、ネットワーク・コンピュータなどのクライアント・システムを、ブラウザによって簡単にSolarisから管理できるソフトウェアである。

### ○サーバの起動

サーバでは、DHCP(Dynamic Hosts Configuration Protocol)、NFS(Network File System)などネットワーク用の複数のサーバ・プログラムを起動することになる。

DHCPサーバは、JavaOS for NCsに対するネットワーク設定(IPアドレス、サブネット・マスク、および、DNSやNISなどの各サーバのアドレス)と、JavaOS for NCsの起動のための設定(プロパティ)のために使用する。ロケール、ファイル・システムのマウント、フォント、使うアプリケーションなどに関する情報を定義する。

NFSサーバは、JavaOS for NCsからサーバ上の資源にアクセスするために使用する。評価ボードには、実行環境であるJavaOS for NCsがフラッシュ・メモリにあるだけで、ハード・ディスクなどのストレージ・デバイスは使用しないため、Javaプログラム(クラス・ファイル)などのファ



【図2】ネットワーク・コンピュータの構成

イルはすべてNFSサーバ・システムから提供される。

ほかに、DNS(Domain Name System)やNIS(Network Information name Service)サーバなどのネーム・システムも利用できる。

### ○コンソールの起動

JavaOS for NCsのコンソールには、ブート処理の経過やデバッグ情報などが表示される。また、Javaプログラムからの標準出力(System.out)などもコンソールに表示される。

コンソールは、評価ボードのシリアル・ポートに出力され、接続先のターミナル・ツール(たとえばSolarisのtipコマンド)で表示する。

### ○プログラム実行

#### ▶ Javaプログラムの配置

「Javaプログラムの開発の進め方」のところで用意したクラス・ファイル、および、実行に必要なファイル(HTMLファイル、データ・ファイルなど)は、Java OS for NCsで実行、アクセスできるように配置する必要がある。

プログラムの配置方法は、JavaアプレットとJavaアプリケーションとで異なる。

Javaアプレットは、一般的なWWWサーバに配置する場合と、ファイル・システムに配置する場合がある。WWWサーバへの配置は、通常行われるのと同じである。つまり、“http:”で始まるURLで指定されるHTML(のAPPLETタグ)に書かれた位置に、アプレットのクラス・ファイルを配置する。

一方、ファイル・システムへの配置は、JavaOS for NCsのファイル・システムとしてマウントされたNFSサーバのディレクトリに配置する。

なお、Javaアプレットの実行に必要なHotJavaブラウザは、それ自体がJavaアプリケーションになっている。

Javaアプリケーションは、ファイル・シ

ステムに配置する必要がある。また、そのときに、Javaアプリケーションのクラス・ファイルと、必要なファイル群はすべて、zipまたはjarコマンドによって、一つのファイルにまとめられる。このファイルをJavaOS for NCsの起動のための設定情報(プロパティ)として定義しておく。

#### ▶ Javaプログラムの実行

いよいよプログラムの実行になる。

電源を入れると、JavaOS for NCsが起動される。初期画面は、JavaOS for NCsの起動の設定によって、ログイン画面またはJavaアプリケーションの選択画面などが現れる。

前述のように、JavaOS for NCsの現バージョンでは、複数のJavaアプリケーションの実行ができない(空間は一つのみ)ので、実行するJavaアプリケーションを一つだけ、最初に選択する。

Javaアプレットを実行する場合は、JavaアプリケーションとしてHotJavaブラウザを選択することになる。

HotJavaブラウザは、マルチタスク環境を実現する仕組みをもつため、マルチウィンドウや、複数のJavaアプレットを実行させることが可能である。

HotJavaブラウザ画面のURLを指定するエリアにJavaアプレットの定義(APPLETタグ)のあるHTMLファイルを指定する。すると、Javaアプレットは、WWWサーバまたはNFSサーバからダウンロードされてブラウザ内で実行される。

Javaアプリケーションの場合は、JavaOS for NCsが起動された後、すぐに、NFSサーバからダウンロードされて実行される。

Javaアプリケーションが終了すると、直ちに、JavaOS for NCsも終了する。

JavaOS for NCsはローカル・ストレージを必要としないため、OS終了時のファイル・システム更新(sync)のペンディングといった心配がない。処理が終わったらいきなり電源を抜いても大丈夫である。

(小森裕之)