デバイス・ドライバとミドルウェアの基礎

矢野 越夫

デバイス・ドライバはハードウェアとOSを橋渡しするソフトウェアである．しかし，現在はデバイス・ドライバにも多くの種類があり，OSによってもその構造は大きく異なる．また，一般的なデバイス・ドライバの形とは異なる「勝手ドライバ」も存在する．本稿では，これらデバイス・ドライバの種類について概観する．（筆者）

はじめに

筆者が本誌のデバイス・ドライバ特集で執筆したのは，思い起こせばもう11年も前です．当時の目次を見ると，今とは環境がずいぶん違っていることに愕然（がくぜん）とします．

と同時に，デバイス・ドライバも種類が増え，作り方や実装方法もかなり進化しました．しかし，オペレーティング・システム（OS）による差異はいまだに存在し，万能のドライバは存在しません．やはり，各OSごとに丹念に作る必要があります．

現在のデバイス・ドライバが昔に比べ，ハードルが高くなったのか，逆に低くなったのかは，場合によって異なります．

本章では，一般的なデバイス・ドライバやミドルウェアの概略をまとめます．OSごとの差異も簡単に説明しますが，具体的な事例は次章以降を参照してください．

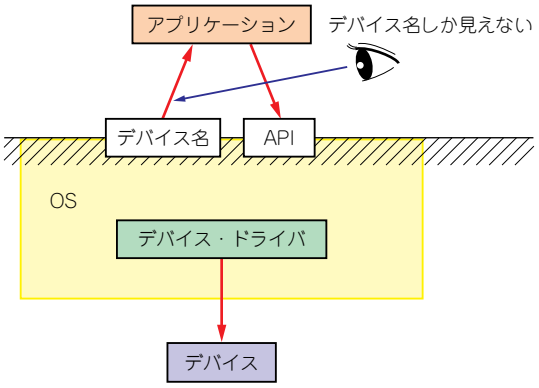


図1　デバイスの隠ぺい

1．デバイス・ドライバとハードウェアとの関係

一昔前なら，デバイス・ドライバとは「ハードウェアを隠ぺいするソフトウェア」という理解でした．**図1**に示すように，アプリケーションから見えるのはデバイス名だけです．

もちろん，現在でもこの解釈は生きています．デバイスを隠ぺいするのもデバイス・ドライバの役割の一つです．けれども，それだけでは不十分です．

2．デバイス・ドライバとOSの関係

現在はOSが発達し，パソコンはもちろんのこと，組み込み用のCPUボードですら複雑なOSを載せる時代です．このように，システム・ソフトウェアの環境がよくなると，おのずからデバイス・ドライバの意味も変わってきます．

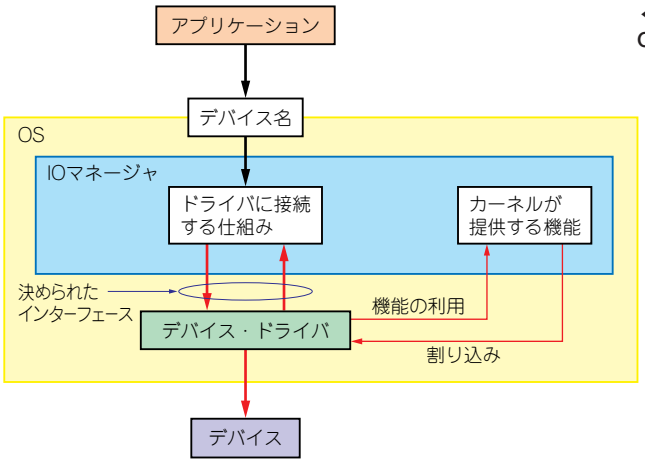
● OSの中のデバイス・ドライバ

普通，アプリケーションが直接デバイス・ドライバを使うことはありません．常にOSを経由し，「デバイス名」を使ってデバイスにアクセスします．

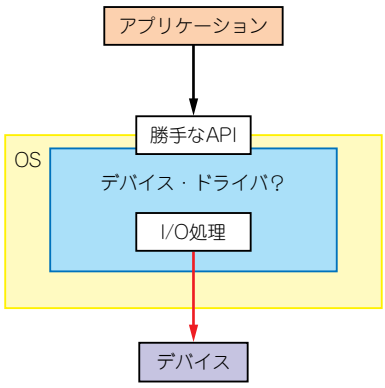
Linuxなら，デバイス名は/dev/nameがアプリケーションから使えます（/dev/sda1など）．

Windowsのデバイス名は少々複雑です．カーネル内部のネイティブ・デバイス名が別名でアプリケーション側にシンボリック・リンクされます．例えば，COM1:とかLPT1:のような名前や¥¥.¥C:のようなデバイス名がアプ

KEYWORD ―― デバイス・ドライバ，ミニ・ドライバ，デバイス名，キャラクタ・デバイス，ブロック・デバイス，勝手ドライバ，PnP



◀図2
OSのドライバ・サポート



▶図3
勝手ドライバ

リケーションから使えます．ネイティブ・デバイス名はアプリケーションからは見えません．

デバイス・ドライバはOSの一部として作られ，OSとの橋渡し部は厳密に取り決められています．これを「デバイス・ドライバ・インターフェース」と呼びます．

言い換えれば，OSからデバイスとして認識されるには，OSの用意したデバイス環境に基づいてドライバを作る必要があるということです．

● IOマネージャ

ほとんどのOSはデバイス・ドライバの実装に便利なようにI/O管理のためのユーテリティを備えています．OSにより呼び方は異なりますが，ここでは「IOマネージャ」と呼ぶことにしましょう．

図2に示すように，IOマネージャはOSとデバイス・ドライバを接続する仕組みを提供し，さらにデバイス・ドライバの実装に役立つ関数も提供します．デバイス・ドライバは，割り込みやタスク・スケジューリング，タイマ機能を使うことにより，入出力待ちの時間を無駄なくOS側に渡します．

IOマネージャのメモリ管理機構は，OSのメモリ・プールを効率的にデバイス・ドライバへ割り当てるために使います．ほとんどの場合，ドライバのメモリはスワップしない固定メモリを使用するからです．

I/Oポート管理やDMA管理は，実際にデバイスからのデータ転送を間違いなく実行するために提供されます．もちろんOSによっては直接CPUの入出力命令を許します．

通常，これらのIOマネージャ関数群は，アプリケーションからは使用できないよう保護されています．

3. デバイス・ドライバの種類

● 標準的なデバイス・ドライバ

OSと協調して動作するドライバが標準的なデバイス・ドライバです．しかし世の中にはOSとは独立したデバイス・ドライバを許す環境も存在します．例えば，μITRONはデバイスの概念がありませんでした．すべてが「勝手な」ドライバだったのですが，最近，トロン協会がデバイス・ドライバの枠組みを導入したようです(1)．

● 勝手なAPIを持つドライバ

「OSに含まれ，ある程度OSの助けを受けて動作しているけれど，完全にOSの一部ではない」デバイス・ドライバも存在します．カーネル・モードに切り替えるためだけにデバイス・ドライバを使うわけです．**図3**に示すように，勝手なAPIを作り，アプリケーションに公開し，デバイスの読み書きを提供するというものです．

OSのデバイス・ドライバ・インターフェースを使うには使うものの，IOCTL（WindowsではDeviceIoControl）を利用して勝手なパラメータでデバイス制御を行うのも，勝手ドライバの一種といえます．

これは，一般的な制御インターフェースが定まっていない特殊なデバイスの場合で，自分で制御コマンドを決めるときなどに使います．特に新しいデバイスの場合で，アプリケーションでデバイス内部のレジスタを読み書きしたいときに勝手ドライバが便利です．

このような場合，デバイス開発終了後，アプリケーションで実施していたレジスタの制御などをデバイス・ドライバ内部に隠ぺいし，完全なデバイス・ドライバとして作り直します．