

第2節 ソフトウェア

本節では、前章で検討されたサービス参照モデルを元に、ビジネスを提供する際に必要となる技術について、その具備すべき要件や既存技術の得失について検討する。

ただし、情報家電をプラットフォームとして提供されるサービスについては、新しいビジネスモデルのアイデアの創出や新しい技術の出現等により常に広がり続けることから、網羅的に限定列挙することはできない。このため、本節では、現在提供されているサービスの延長や、既に開発されている技術を前提として想定されるサービスについて、検討を行うこととする。

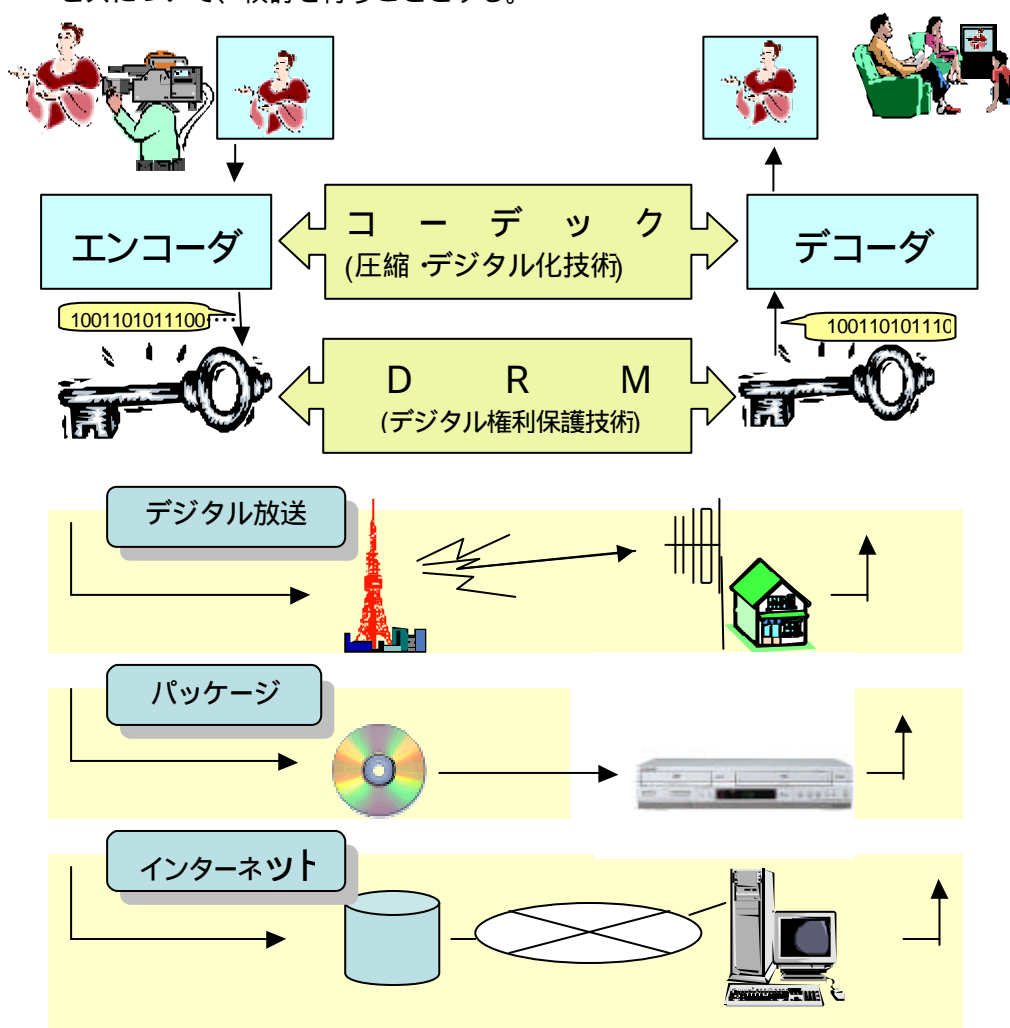


図1 情報家電におけるサービスの一例（コンテンツ配信サービス）

1. コーデック (enCOding & DECcoding)

(1) 技術の概要

コーデックとは、映像や音響(音楽・音声等)を0と1の羅列によるデジタル情報に変換し、またデジタル情報から元の映像・音響を復元する技術の総称である。デジタル変換にあたっては、できるだけデータ量を少なくするよう様々な工夫が行われることから、「映像圧縮技術」「音声圧縮技術」等とも呼ばれる。

(2) サービスモデルを実現するためにコーデック技術が具備すべき要件

インターネット等を通じたコンテンツ配信サービスモデルを念頭に、コーデックが具備すべき主な要件を考察する。

圧縮効率

データ量は通信(伝送)コストや記憶(メモリ)コストに直結する指標であることから、元の情報をできるだけ損なうことなく、データ量を出来る限り小さくすることが要求される。圧縮効率(元の情報に対するデジタル変換後のデータ量)は、コーデックを評価するための重要な指標である。

スケーラビリティ

圧縮効率と元の情報の復元の忠実性との間にはトレードオフの関係が成立する。このため、「最高の技術」は存在せず、利用用途に応じてバランスのとれた技術を採用する必要がある。具体的な例を挙げれば、テレビジョン放送やDVD等では、映像品質を優先する必要があることから、元の情報の損失を極力少なくできるMPEG-2が採用されている。一方、近く開始される予定の携帯電話等向けの放送(1セグメント放送)では、映像品質が若干犠牲になるものの圧縮効率を飛躍的に向上できるH.264/AVCの採用が決定している。

スケーラビリティとは、情報復元の忠実性と圧縮効率のトレードオフに対する柔軟性のことである。特に、インターネットでは放送やパッケージと異なり伝送速度や記憶容量が個々のユーザーによって大きく異なるため、柔軟なスケーラビリティを設定できることが重要である。

ストリーミングへの対応

映像・音楽配信サービスモデルは、蓄積型とストリーミング型に大きく分類できる。蓄積型とは、インターネット等のメディアを通じて送られるデー

タをユーザー側でハードディスク等に保存した上で読み出す方式である。一方、ストリーミング型は、送られてくるデータを逐次デコードする方式である。具体的には、テレビジョン放送をビデオ録画することは蓄積型にあたり、そのままテレビジョンで視聴することはストリーミングにあたる。

デジタル信号は複製しても品質の劣化がないことから、複製に対してナーバスなコンテンツホルダーは、蓄積型を避けてストリーミング型のサービスを指向する傾向がみられる。しかしながら、ストリーミング型のサービスからデータ列を抜き出しHDD等に記録するツール等が存在しており、ストリーミング型サービスのビジネスモデルの根幹を揺さぶりつつある。後述するDRMとの緊密な一体化を図ることで「データ抜き取り」を防ぐなど、ストリーミング型サービスを確実に実現するための技術開発が重要である。

将来にわたる継続的なサポート

1960年9月に我が国でカラーテレビジョン放送が開始されて以来、アナログTVの方式であるNTSC方式で制作された映像コンテンツは膨大な量に及ぶ。現在の計画では、2011年7月にアナログのNTSC方式による地上波テレビジョン放送は終了する予定であるが、予定通り終了したとしても、NTSC方式は50年以上にわたり継続されたこととなる。

デジタルでのコンテンツ配信サービスにおいてコーデックはサービス基盤そのものであり、OSのように数年に一度のバージョンアップを繰り返し、そのたびに仕様変更されることは、コンテンツ制作者のみならずユーザーにも多大な混乱をもたらすことが懸念される。このため、頻繁なバージョンアップを繰り返すのではなく、将来にわたる継続的なサポートが約束されることが重要である。

ちなみにOS分野では、頻繁なバージョンアップを続けるWindowsにおいて2世代前のバージョンであるWindows98のサポートの打ち切りをMicrosoft社が発表したところ、継続的なサポートを求めるユーザーの声が高まり、2006年までサポートを延期するという事態も生じている。

(3) コーデック技術の現状と将来展望

映像コーデックだけに限定しても、現在提供されているコーデックは、MPEG-2、MPEG-4 visual、H.264/AVC、Windows Media Technology-Video、Real Video、Quick Time等多くの種類がある。

この中で、MPEGについては、ISO/IECにおいて国際標準として策定されたものであり、将来にわたるサポート継続性という点では確実性が高い。このため、特に継続性を重視するテレビジョン放送においては、全世界的にMPEGが採用され

ている。同様に、DVDについてもMPEGが採用されている。

また、国際標準であるMPEGについては、関連特許を保有する企業が多いことから、MPEG-LAという組織によってパテントプールが行われている。

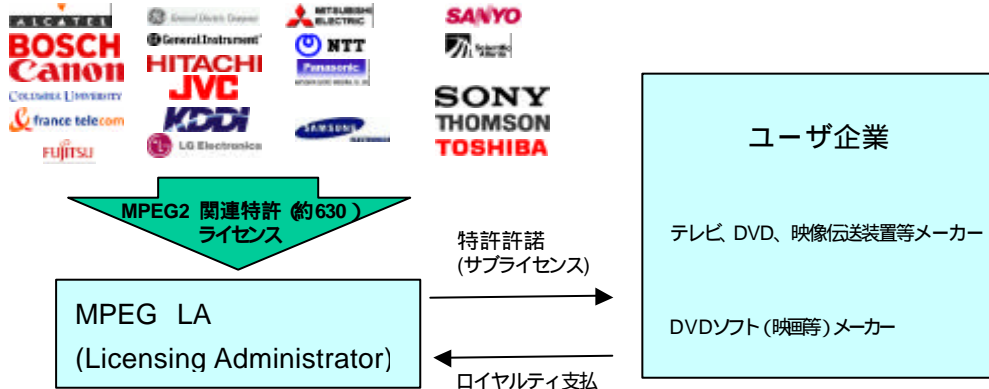


図2 MPEGのパテントプールの仕組み (MPEG-2のケース)

方式	MPEG-4 Visual	H.264/MPEG-4 AVC		Windows Media Video 9
ライセンス管理者/団体	米MPEG LA, LLC	米MPEG LA, LLC	米Via Licensing Corp.	米Microsoft Corp.
符号化料	0.25米ドル (ただし、年間5万個以内であればライセンス料は徴収せず)	ライセンス料を徴収せず (10万個未満) 0.2米ドル (10万個以上500万個未満) 0.1米ドル (500万個以上)	0.25米ドル	0.2米ドル
符号化料のライセンス料の上限値 (1年当たり)	100万米ドル	350万米ドル (2005年~2006年) 425万米ドル (2007年~2008年) 500万米ドル (2009年~2010年) ただし、2005年まではライセンス料を徴収せず	250万米ドル (パソコン用以外のソフトウェアの場合)、400万米ドル (パソコン用のソフトウェアの場合)	80万米ドル
復号化料	符号化料と同じ	符号化料と同じ	0.25米ドル (ただし復号化の回数が20回以内、あるいは最初の復号化から30日以内といった使用期間が設定される場合は、0.0025米ドル)	0.1米ドル
復号化料のライセンス料の上限値 (1年当たり)	符号化料と同じ	符号化料と同じ	符号化料と同じ	40万米ドル
符号化/復号化料	記述なし	符号化料と同じ	符号化料と同じ	0.25米ドル
符号化/復号化料のライセンス料の上限値 (1年当たり)	記述なし	符号化料と同じ	符号化料と同じ	100万米ドル
放送事業者	年間最大100万米ドル	年間1万米ドル (ただし加入世帯数が10万未満の場合、ライセンス料は徴収せず)	ライセンス料は徴収せず	ライセンス料は徴収せず
DVD媒体	0.04米ドル (5年以内の新作映画) 0.02米ドル (5年が経過した映画)	ライセンス料は徴収せず (12分未満) タイトル1本につき0.02米ドルあるいは売り上げの2% (12分以上)	0.05米ドル (30分未満) 0.15米ドル (30分以上90分未満) 0.25米ドル (90分以上)	ライセンス料は徴収せず
コンテンツ提供者に対する課金	0.000333米ドル/分	DVD媒体と同じ	DVD媒体と同じ	ライセンス料は徴収せず

図3 代表的なコーデックにおけるライセンス料

一方、インターネット配信については、サービス事業者ごとにコーデックを選択しており、現状では、Microsoft社のWindows Media Technology-Videoと、RealNetworks社のReal Videoの両方を採用するサービス事業者が多い。しかしながら、インターネットでのコンテンツ配信ビジネスが本格化すれば、いずれかの方式が一気にデファクトとなる可能性も高い。Microsoft社の戦略は、インタ

ーネットを通じたコンテンツ配信においてWindows Media Technology-Videoでデファクトを獲得し、同方式を採用するコンテンツが蓄積されることをレバレッジとして、DVDやテレビジョン受信機にWindows Media Playerのデコーダを搭載する方向に誘導することにあると考えられる。もし、そのような状況になれば、PCの世界で生じているWindowsの寡占と類似の状況が、DVDやテレビジョン受信機において生じることも懸念される。

なお、我が国のコンテンツホルダーは、1社毎の市場占有率は比較的小さいことから、デファクト競争のゲームメーカーにはなりにくいのが現状である。

2. DRM (Digital Rights Management)

(1) 技術の概要

情報家電をプラットフォームとする様々なコンテンツ配信サービスにおいては、著作権保護技術がビジネス成立の重要なファクターとなる。DRM (Digital Rights Management) 技術とは、コンテンツに対する暗号化技術、ユーザー及び機器の認証技術等の著作権保護技術の総称である。

(2) サービスモデルを実現するためにDRM技術が具備すべき要件

DRM技術の仕様は、サービスモデルによって異なる。

例えば、デジタルテレビジョン放送(地上波/BS/CS)では、B-CAS (BS-Conditional Access System) という方式が採用されているが、同方式ではコピー世代制御(1回の録画は可能だが、そこからのダビングは不可)等が行われている。



注1: 「1回だけ録画可能」の信号が加えられた番組は、コピー制御に対応した記録メディアとデジタル録画機器の組み合わせで録画可能。

図4 デジタルテレビジョン方式で導入されているB-CASの仕組み

また、DVDについては、松下電器産業等が開発したCSS (Content Scrambling System) というDRM技術が採用されている。これは、DVDに記録

されるコンテンツそのものを暗号化し、DVDメディアに書き込まれている暗号鍵と、機器の持つ暗号鍵からコンテンツの復号に必要な鍵を生成する技術である。この技術により、市販のDVDソフトに記録されているデータをコピーしても暗号化されていて再生できず、また、正規に許諾を受けた機器（又は再生ソフト）でなければDVDソフトを再生できない仕組みになっている。

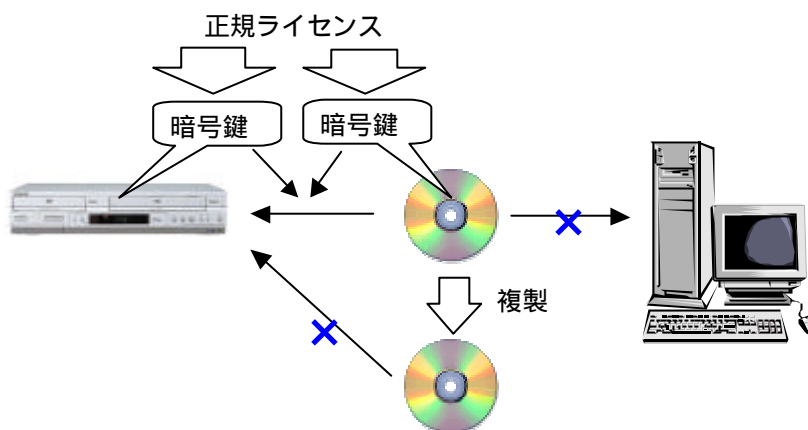


図5 DVDにおけるアクセス制御（CSS）のしくみ

（3）DRM技術の現状と将来展望

上述のとおり、テレビジョン放送及びDVDにおいては、すでに一定のDRM技術が導入されており、コピー防止やコピー世代管理等のサービスモデルからの要求に応えている。

しかしながら、1999年10月にヨーロッパの2人のハッカーによってCSSの暗号解読ソフト（DeCSS）が開発され、アンダーグラウンドでは、ライセンスを受けていないパソコン（のソフトウェア）でDVDソフトが再生・複製されるようになった。また、このようにしてPCに取り込まれたDVDソフトが、Winny等のファイル交換ソフトにより、個人から個人に次々に不正コピーされる事態になっており、社会問題となっている。

インターネットによるコンテンツ配信サービスにおいては、Microsoft社はWM-DRM(Windows Media-Digital Rights Manager)により、コンテンツの暗号化、コンテンツの保護再生等の技術のライセンスを供与している。一方、日立製作所、インテル、松下電器産業、ソニー、東芝の5社が共同して、DTCP/IP(Digital Transmission Content Protection over Internet Protocol)というDRM技術

を開発しており、2003年9月に仕様が公開されている。

インターネットによるコンテンツ配信サービスにおいては、コーデックと同様、DRM技術の選択はサービス事業者ごとの判断に委ねられており、デファクト競争が生じつつある。例えば、Microsoft社のホームページによれば、既にディズニー社がWM-DRMのライセンス供与を受けたことがPRされている。

DRM技術は、ビジネスの重要ファクターである課金システムとも密接に関連することから、インターネットでのコンテンツ配信ビジネスが本格化すれば、早いタイミングでデファクト競争が一気に決着する可能性が高い。コーデックのケースと異なり、DRMについてはISO/IEC等の国際標準が策定されていないことから、TCP/IPのような業界でのフォーラム活動の重要性が極めて高いと考えられる。

3. 多プラットフォームに対応した処理記述(プログラミング)言語

(1) 技術の概要

オンラインゲームや電子メールなどのアプリケーション・サービスを提供するためには、クライアント(ユーザー側の機器)においてアプリケーション・ソフトウェアを実行する必要がある。クライアントがPCに限定されたアプリケーション・サービスであれば、Windowsで動作するアプリケーション・ソフトウェアを配信することでこれらサービスを提供できるのに対し、クライアントが情報家電である場合には、用いられているCPUやOSがまちまちであるため、多プラットフォームに対応するための工夫が必要となる。

アプリケーション・ソフトウェアは、一般に、プログラミング言語を用いて開発され、コンパイラによってオブジェクトコードと呼ばれる形式のファイルに変換されて、クライアントに配信される。クライアントは、ユーザーによ

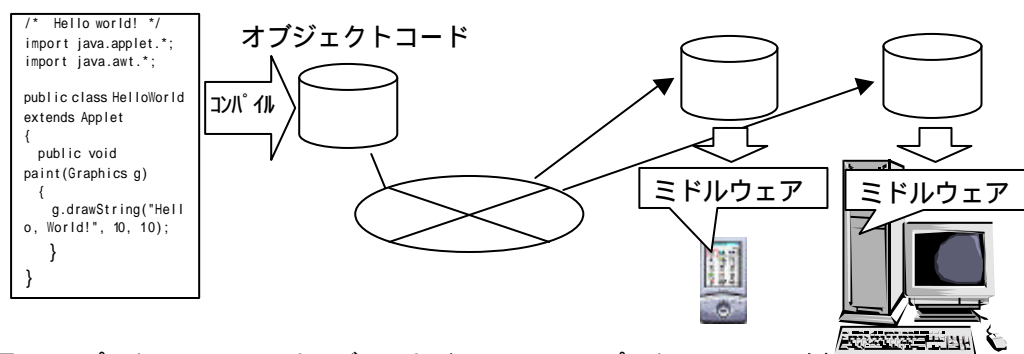


図6 アプリケーション・サービスのしくみ(JAVAアプリケーションの例)

て CPU や OS がまちまちであるため、一般にオブジェクトコードをそのまま実行することはできない。このため、クライアント毎の CPU や OS の差異を吸収するために、ミドルウェアと呼ばれるソフトウェアが用意されており、このミドルウェアによって CPU や OS の差異を超えて、アプリケーション・ソフトウェアが実行できるようになる。

(2) アプリケーション・サービスを実現するために処理記述(プログラミング)言語が具備すべき要件

多プラットフォーム対応・汎用性

PC では、OS はマイクロソフト社の Windows、CPU はインテル社の Pentium 又はその互換 CPU からなる PC/AT と呼ばれるアーキテクチャーの寡占化が進んでおり、ほぼ全ての PC がこのアーキテクチャーを採用している。このような単一プラットフォームでは、わざわざミドルウェアを用意してオブジェクトコードを実行しなくても、そのプラットフォーム上で直接ソフトウェアを実行することも可能である。このような直接実行可能なアプリケーション・ソフトウェアをネイティブコードと呼ぶ。PC 上で使われているワープロ・表計算・ゲーム等のアプリケーションの多くはネイティブコードである。

一方、情報家電では、製品毎に多種多様なアーキテクチャーが採用されており、情報家電をプラットフォームとするアプリケーション・サービスは、様々なアーキテクチャーに対応する必要がある。このため、汎用性、すなわち「Write Once, Run Anywhere(一度プログラムを書けばどんな環境でも動作する)」が極めて重要である。

前章でも指摘されているように、多プラットフォームに対応するためには、クライアントの仕様(OS や CPU) に依存しない形で、処理記述の方法とデータの構造(サービス・アーキテクチャー及びデータアーキテクチャー)が統一されることが必要である。

セキュリティ(安全性)

アプリケーション・ソフトウェアは、クライアントに配信されて実行されることから、万が一、コンピュータ・ウィルスが混入したり、悪意あるオブジェクトコードやデータが配信されると多大な被害を受けるおそれがある。

このため、処理記述(プログラミング)言語でいかに悪意あるプログラムを書こうとも、クライアントのリソース(HDD 等)にアクセスするこ

とがないようにするなど、セキュリティが確保されていることが重要である。

実行性能（高速性、機能性(ネットワーク対応等)）

オブジェクトコード及びミドルウェアを介して実行されるアプリケーション・ソフトウェアは、ネイティブコードによって直接実行されるアプリケーション・ソフトウェアに比べ、一般に実行速度が低下する。これは、ミドルウェアによって、オブジェクトコードを解釈する時間が必要になるためである。最近では、ミドルウェアを半導体 IC に組込むなど、オブジェクトコードを高速度で実行する技術も提案されている。

また、機能面では、先述のセキュリティを確保しつつ、高度で多様な処理・制御を実現できることが求められており、特に、ネットワーク関連機能に対するニーズが大きい。しかしながら、便利なネットワーク機能（例えば、遠隔サーバーからクライアントのHDDへの直接アクセス機能など）は、ともすればコンピュータ・ウィルスの侵入に悪用されかねず、セキュリティとネットワーク機能にはトレードオフの関係が生じる。

仕様の公開及び将来にわたる継続的なサポート

コンテンツ配信サービスにおけるコーデックとDRM技術と同様、アプリケーション・サービスの提供においては、処理記述（プログラミング）言語が重要なサービス基盤となる。

特に、多プラットフォームに対応するためには、仕様がオープンになっていることが重要である。万が一、オープンにされていない機能（いわゆる隠しコマンド）が存在し、その機能を利用するアプリケーション・ソフトウェアが存在すると、その機能をサポートしていないミドルウェアを搭載した製品ではそのアプリケーション・ソフトウェアは動作しない。

また、サービス基盤であるがゆえに、継続的なサポートが重要であることも、コーデック及びDRM技術と同様である。

（3）処理記述（プログラミング）言語の現状と将来展望

「Write Once, Run Anywhere」というコンセプトを最初に提案したのは、Sun Microsystems社が1995年に発表したJAVAである。現在、JAVAは多プラットフォームに対応したアプリケーション・サービスのデファクトとなっており、例えばNTTドコモの携帯電話向けサービス「i-mode」の中で提供されるアプリケーション・サービス「iアプリ」等において広く採用されている。

一方、PC の分野では、特定のプラットフォームの寡占が進んでいることから、現在のところ、多くのアプリケーション・ソフトウェアがネイティブコードによって作られている。しかしながら、ネットワークを利用するアプリケーションでは、先述のようにセキュリティの向上が喫緊の課題となっていることから、PC の分野でデファクトを握る Microsoft 社も、ミドルウェアを介して安全にアプリケーション・ソフトウェアを実行するというコンセプトを採用した .net (ドット・ネット) を発表している。

多プラットフォームである情報家電では、現在のところアプリケーション・サービスはまだ本格的に開始されていない。ただし、NTT ドコモの携帯電話向けサービス「i-mode」が壁紙サービスと着信メロディーサービスによって大ヒットしたように、キラー・アプリケーションの登場によってブレイクする可能性も高い。また、技術開発の観点からは、ネイティブコードに比べ処理が遅くなる欠点を補うために、ミドルウェアを組み込んだ半導体 IC の開発など取り組みも積極的に行われている。

情報家電をプラットフォームとするアプリケーション・サービスが本格的に立ち上がるためには、先述のように、処理記述 (プログラミング) 言語の仕様がオープンであり、将来にわたるサポートがある程度保障されていることが重要である。

4 . その他

これまで検討してきたサービスモデル実現するために要求されるコア技術の他にも、ソフトウェア分野における重要なコア技術が存在する。これら技術の開発・デファクト競争の主眼は、「サービスモデルの実現」というよりも、むしろ、製品コストの低減、開発期間の短縮、ユーザーへの機能提供 (使い勝手) 等にある。

本節の最後に、OS (Operating System) とヒューマン・インターフェイスについて考察する。

(1) OS (Operating System)

最近の家電製品は、AV 機器はもちろんエアコン・洗濯機などの白物に至るまで、CPU が内蔵され、CPU と内部に組み込まれたソフトウェア (以下、組込ソフト

ウェアという。)によって、様々な機能(例えば、エアコンの場合であれば、温度・湿度制御、タイマー、風向き調整など)が提供されている。

HDDや半導体メモリの管理、ネットワークとのやりとり、キーボード・マウス(情報家電の場合はボタン・スイッチ等)の制御など、多くのアプリケーション・ソフトウェアや組込ソフトウェアから共通して利用される基本的な機能を提供し、システム全体を管理するソフトウェアがOSであり、「基本ソフトウェア」と呼ばれる。PC分野では、Microsoft社の提供するWindowsが寡占的なシェアを獲得してデファクトとなっている。

一方、情報家電の分野では、OSを用いなくて、アプリケーション・ソフトウェアや組込ソフトウェアがボタン操作やネットワークとのやりとりを直接制御するケースもあるが、多くの情報家電ではOSを採用する方向にある。

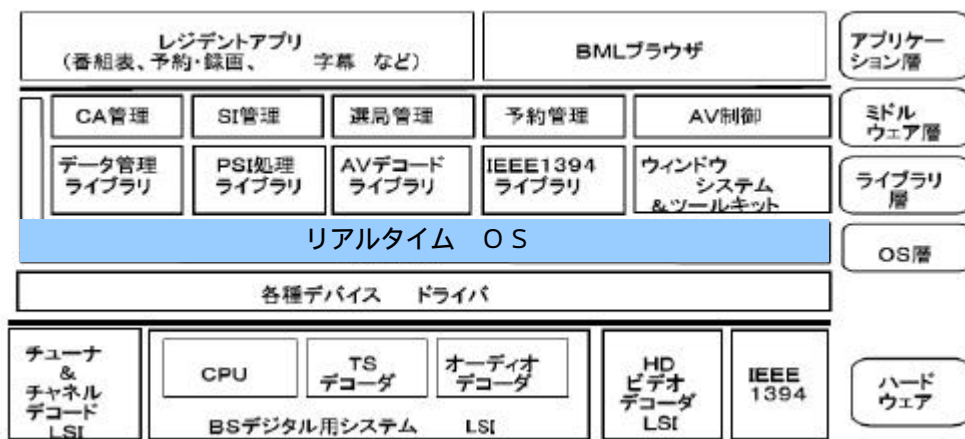


図6 情報家電の構造 (デジタルテレビジョンの例)

OSに求められる要件として、機能、リアルタイム性、仕様の公開性及び継続的なサポート、迅速な起動、等が挙げられる。

特に情報家電においては、例えば、「録画ボタンを押してもすぐに録画が始まらない」「チャンネルボタンを押してもすぐにチャンネルが変わらない」等は製品のクレームに直結することから、リアルタイム性の確保(ある処理を一定時間以内に終了するよう、実行中の他処理を中断するなどきめ細かい時間制御を行うこと)が極めて重要な課題である。このようなリアルタイム処理を行うOSはRTOS(Real Time OS)と呼ばれ、TRONプロジェクトで開発されたITRONなどがある。

PCの分野でデファクトとなっているMicrosoft社のWindowsは機能面では優れているものの、リアルタイム性や組込ソフトウェアでのノウハウの蓄積ではITRONが勝るなど、得意とする領域が異なっている。このため最近では、ユビキタス・コンピューティング環境構築のためのオープンなリアルタイムシステム標準開発環境の実現を目指してTRONプロジェクトが発足させたT-EngineフォーラムにMicrosoft社が参加するなど、コラボレーションの動きが見られる。

また、1991年にヘルシンキ大学の大学院生(当時)Linus Torvalds氏によって開発されたLinuxは、オープンソースソフトウェアのOSとして注目されているが、このLinuxを情報家電をはじめとする組込分野に使用しようとする動きもある。Linuxはオープンソースソフトウェアであるため、仕様が完全に公開されており、また、自由に改変することもできるため継続的なサポートが容易であるとされている。

(2) ユーザー・インターフェイス(User Interface)・デザイン(design)

PCでは、ユーザーによる操作などの入出力は、主にキーボード・マウス及びディスプレイを介して行われる。これに対し、情報家電では、お年寄りや子供など、必ずしも操作に慣れていないユーザーにも容易に操作できることが求められる。また、利用されるケースも、例えば車内や台所など、必ずしもキーボード・マウスが使えないことも想定される。

このため、情報家電では、PC以上に多様なユーザー・インターフェイスに対応することが求められる。これを実現するソフトウェア技術として、具体的には、音声による操作を実現する「音声認識・音声合成技術」、操作者の瞳孔の動き等から視線を読み取る「視線入力技術」、入力されたデータからその意味を解釈する「セマンテック技術」等のユーザー・インターフェイス技術が挙げられる。

サービスモデルを決定づけるコーデック、DRM技術等については、標準化・普及のためのフォーラム活動が重要であるのに対し、ユーザー・インターフェイス技術は個々の製品の競争力に関わる技術であり、それぞれの製品毎に機能・性能向上を目指した技術開発競争が行われる技術分野である。このため、大企業のみならず、中小ベンチャー企業においても積極的に研究開発が進められており、設立後まもない小さな会社であっても、世界に通じる技術を有するケースもある。

情報家電におけるコア技術の開発という観点のみならず、ベンチャー企業の活

性化という観点からも、我が国において積極的な技術開発が行われるよう支援が必要な分野であるといえる。

また、家電製品の市場競争では、機能・コスト以外に、デザインが極めて大きな要素であり、消費者市場では、「cool(カッコいい)」が一つのキーワードとなっている。単なる外観設計でなく、心理学や認知科学をベースにしたデザイン技術の開発が競争力を大きく左右する。