

Middleware

Dr Shirazi

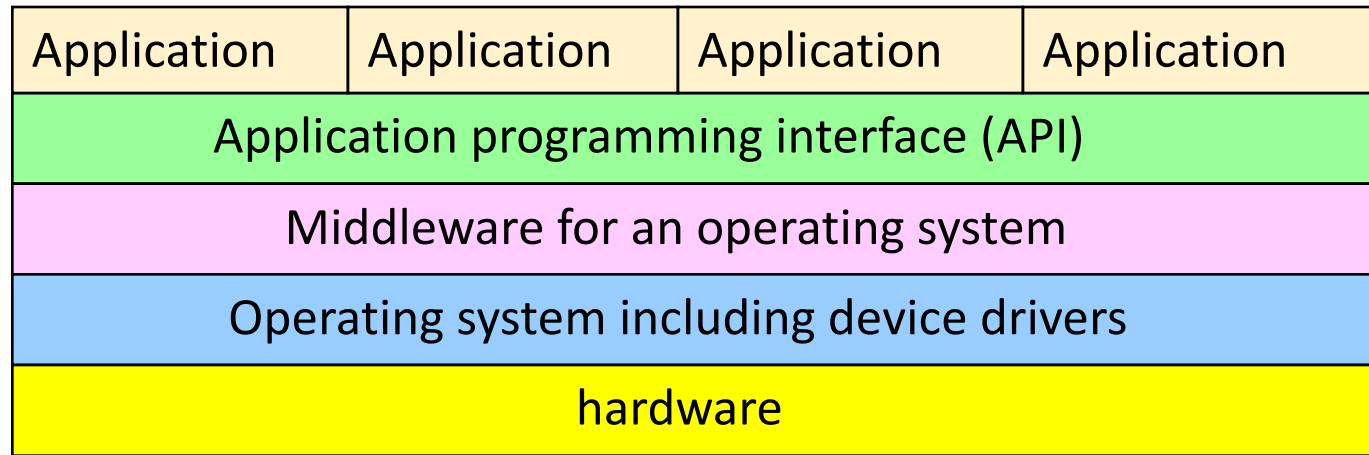
What is Middleware? Why do we need it?

- Middleware is an execution environment that lies between the operating system and the application.
- It allows applications to run in the same way on any STB without caring (or directly communicating with) the hardware or operating system.
- For TV based interactive services we expect everything to have a common look and feel (Common fonts, graphics, download functionality, menus etc.). However;
 - Services are provided by many different service providers.
 - STBs are manufactured by many different companies.
 - Services are broadcast by different broadcast providers across many different routes.
- Need to **unite all services under a common software platform.**

Middleware & API

- It allows the same application to run on different models of STB but to look and behave in the same way.
 - The application may be channel selection, an electronic programme guide, purchasing pay per view events, subtitles etc.
- It is like an API but is more than an API.
 - Like an API in that it offers a common set of functions which all applications must use in order to run.
 - Unlike an API in that it is required to communicate directly with the hardware (an API is not).

Software Model for Set Top Box



- Applications are the top layer and are what we directly interact with.
- An API operates between the applications and the middleware.

Software Model for Set Top Box

Application	Application	Application	Application
Application programming interface (API)			
Middleware for an operating system			
Operating system including device drivers			
hardware			

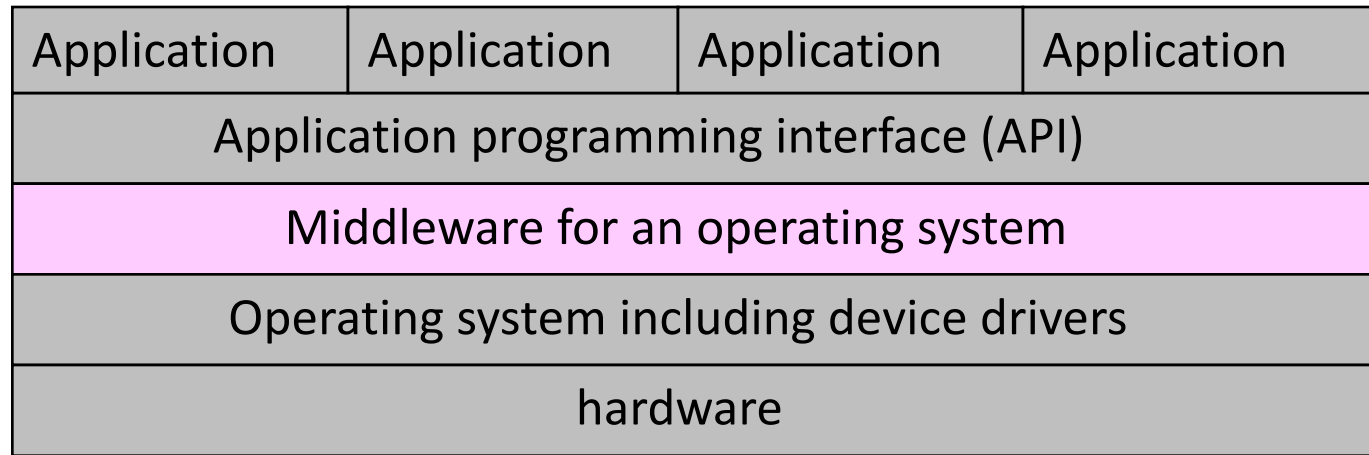
- Hardware layer
 - Enable fundamental functionality of STB.
 - MPEG decoder, RF front end.
 - CPU, hard disk, memory.

Software Model for Set Top Box

Application	Application	Application	Application
Application programming interface (API)			
Middleware for an operating system			
Operating system including device drivers			
hardware			

- The operating system
 - Makes hardware functions available to applications.
 - Software tends to be optimised for specific operating systems.

Software Model for Set Top Box



- Middleware decouples services and executing hardware and system software.
- The middleware is a software layer implemented specifically for the OS running on the receiver.

Examples of Applications

- Network PVR – networked personal video recorder
- VOD – video on demand
- Games – can play games on your STB
- DVD – it can talk to your DVD player
- Broadcast – usual use of STB
- IMS/SIP Apps – communications applications like VOIP

Middleware systems in common use

- MHEG-5 (Multimedia and Hypermedia information coding Experts Group)
- MHP (Multimedia Home Platform) – a DVB standard, not widely used
- OpenTV used by a number of broadcasters
- SeaChange mainly used in cable systems
- Microsoft TV (now abandoned but some aspects sold to Ericsson)
- Google TV
- Android TV
- Tivo

MHEG 5

- MHEG-5 is the system used for UK, New Zealand, Hong Kong and Ireland systems.
 - This is a public standard from the Multimedia and Hypermedia Information coding Experts Group
 - This is particularly suitable for low-end STBs as it makes limited demands on the hardware
- The next generation for the UK is YouView (see later lecture on Connected TV)

MHEG-5 goals

- The major goals were
 - To provide a good standard framework for the development of client/server multimedia applications intended to run on a memory-constrained Client.
 - To define a coded representation for interchange of applications across platforms of different versions and brands.
 - To provide the basis for guaranteeing that an application will run on all conformant terminals.
 - To allow the runtime engine on the Client to be small and easy to implement.

More MHEG 5 goals

- To be free of strong constraints on the architecture of the Client.
- To allow the building of a wide range of applications. This means also providing access to external libraries. An application using external libraries will only be partly portable.
- To allow for application code that is guaranteed to be "safe" in the sense that it cannot harm other code in the Client, nor put the Client in an abnormal state.
- To promote rapid application development by providing high-level primitives

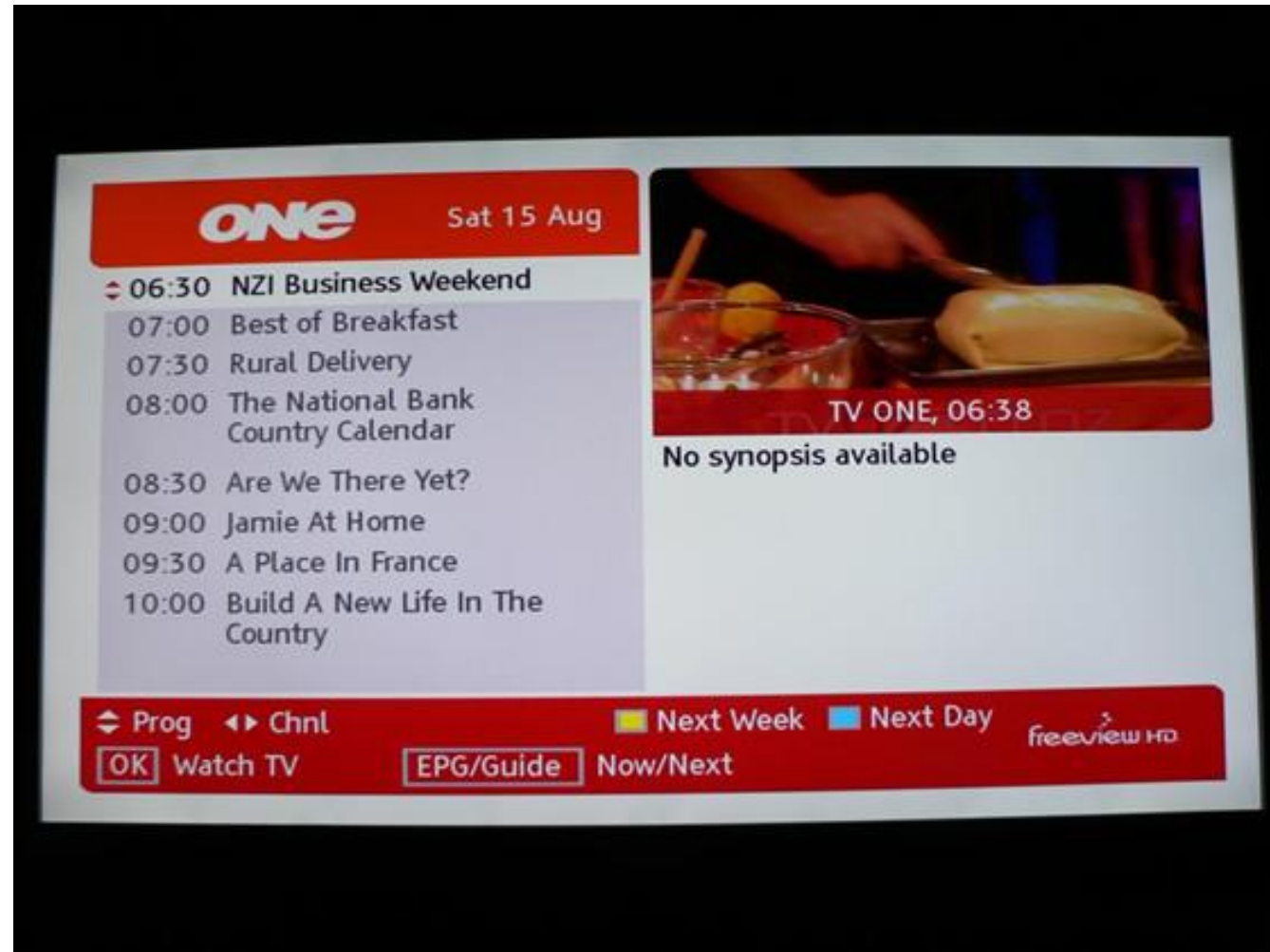
MHEG-5 essentials

- An MHEG-5 application is
 - a set of multimedia objects at the TV operator's head end
 - streamed to the STB
 - MHEG-5 engine interprets the data and displays information on the screen
- The MHEG-5 language describes multimedia presentations as a collection of objects. It includes:
 - Visible objects (bitmaps, text boxes, buttons, sliders).
 - Links (objects that trigger a sequence of actions).
 - Programs (local on a set top box or running on a server).
 - Variables (integers, boolean, string etc.).
- MHEG-5 is object orientated

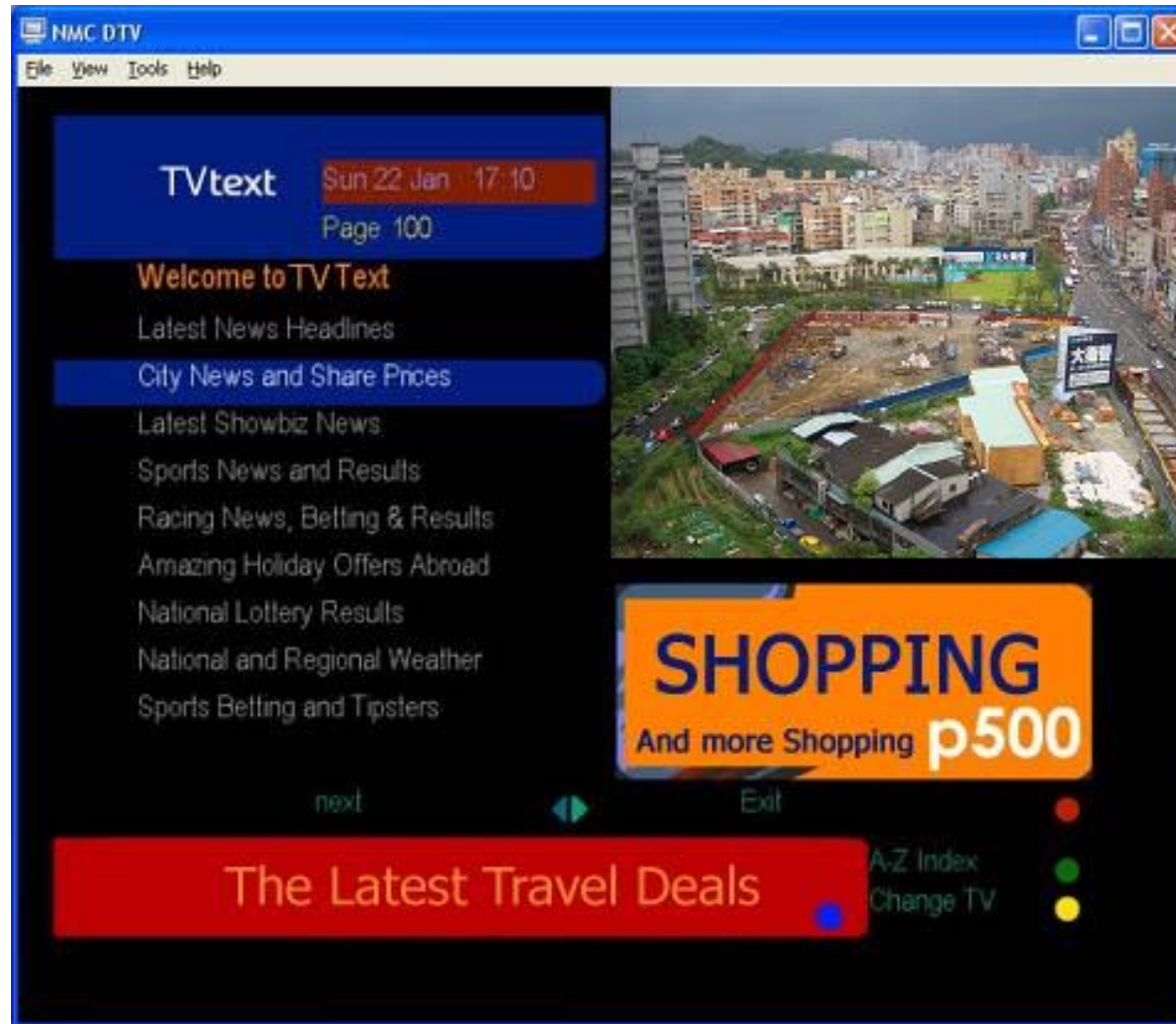
TV With Information Overlaid



TV Inset with EPG



Different Services



Concepts introduced in MHEG-5

- Some concepts introduced in MHEG-5 have also been used in the later MHP standard
- The concept of plug-ins.
 - allow the basic functionality to be extended to handle extra formats e.g. a plug-in could be used to display MPEG-4 encoded video
- Font and Graphics formats
 - a set of fonts, colours, screen layouts etc. that are defined by a broadcaster so that all STBs will give the same look and feel for the services provided by that broadcaster.

Multimedia Home Platform (MHP)

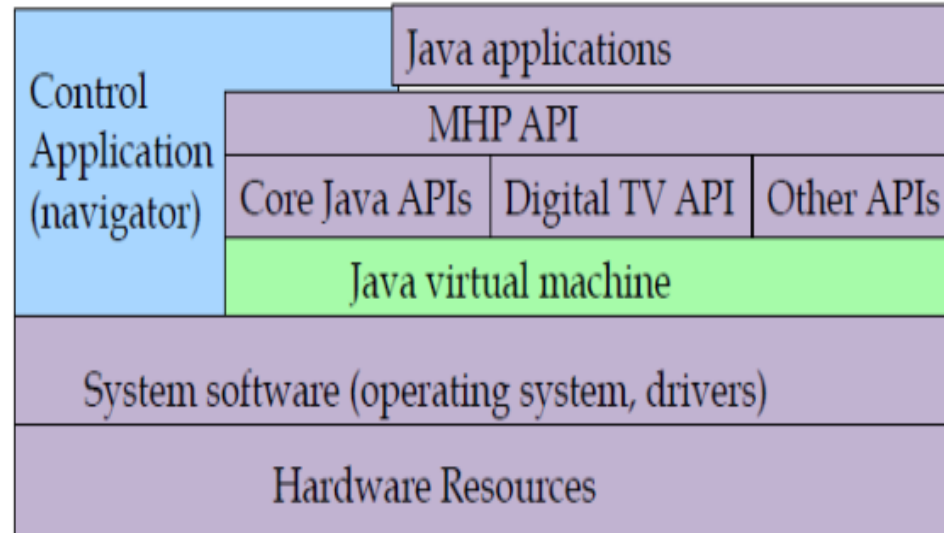
DVB-MHP

- Multimedia Home Platform (MHP) is the middleware standard developed by the DVB group
 - This is a public (open) standard (as is MHEG-5)
 - Based on a Java virtual machine but not all of Java is supported
 - The structure is typical of other middleware but MHP itself has not been successful commercially
- Next generation is Hybrid Broadcast Broadband TV (see later connected TV lecture)

MHP 1.0

- Version 1.0 was finalised by DVB project in 2000 and subsequently published by the ETSI later in the same year.
- Cleaner version (MHP 1.0.3) was finalised in 2003.
- Java based middleware
- To enable different terminal classes, it defines three profiles:
 - Enhanced Television
 - Interactive Services
 - Internet Profile (MHP 1.1)
- MHP-compliant applications run on all kind of devices
 - requires full implementation of MHP specification

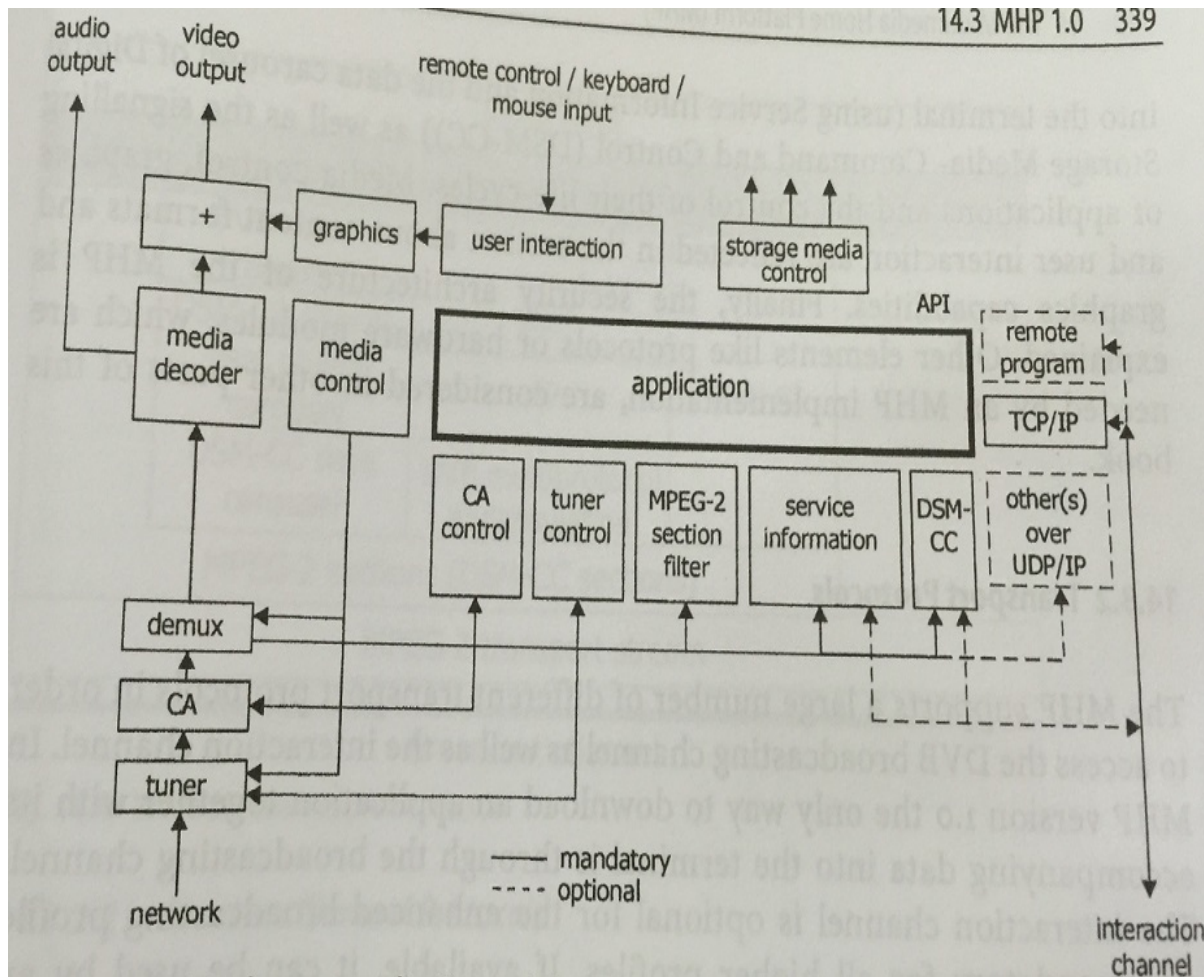
Basic Architecture



Basic MHP /DVB-J architecture

- OS integrates all HW drivers.
- All implementation specific software components are also realised within OS.
- JVM is responsible for hardware and OS abstraction
- MHP API runs on top of JVM and includes DTV API, Core JAVA API, User I/O API etc.
- The sum of all MHP Java functions is referred to as DVB-J.
- DVB-HTML applications are also defined in MHP.
- A Java application could be an Electronic Programme Guide, a web browser or a game.

Interface between MHP and Apps

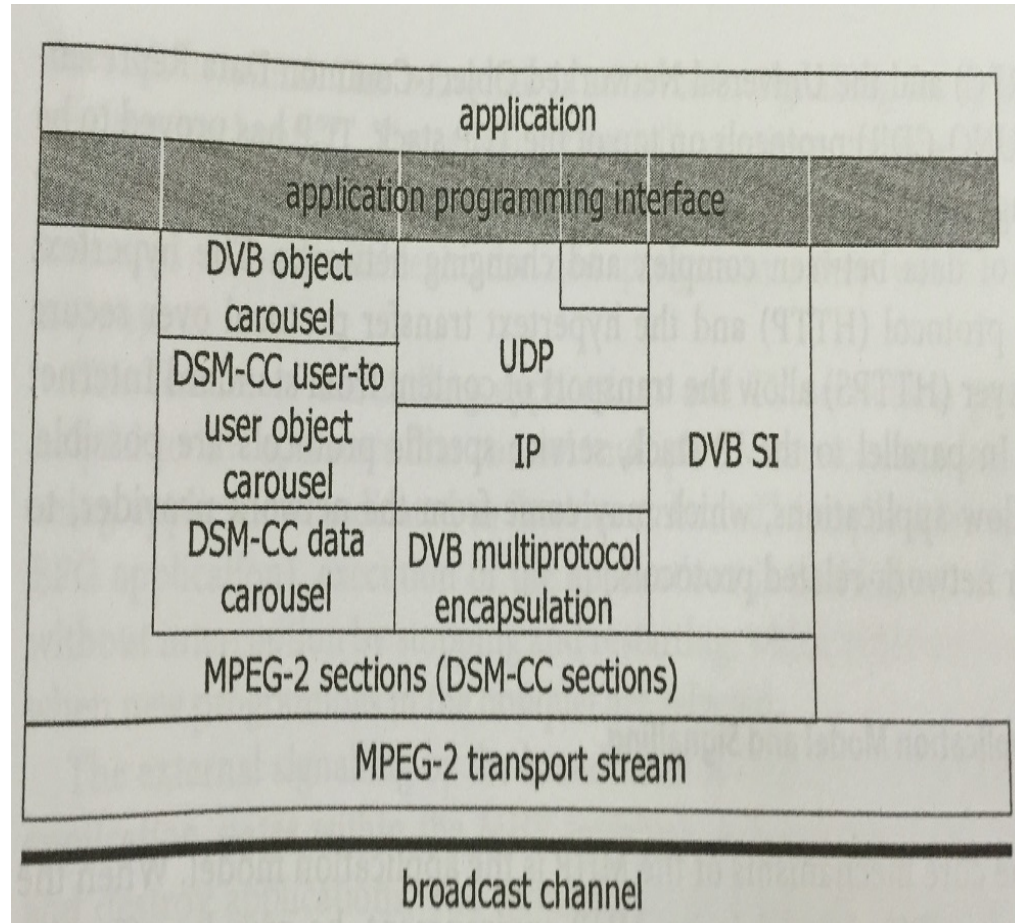


- Control Application/Navigator serves as the basic interface with the user.
 - Pre-installed, vendor specific
 - Enables service selection along with their monitoring and control.
- Even if the hardware resources are physically distributed, MHP logically combines them for the application execution.
- Applications are transported using the service information and data carousel of DSM-CC

Transport protocols

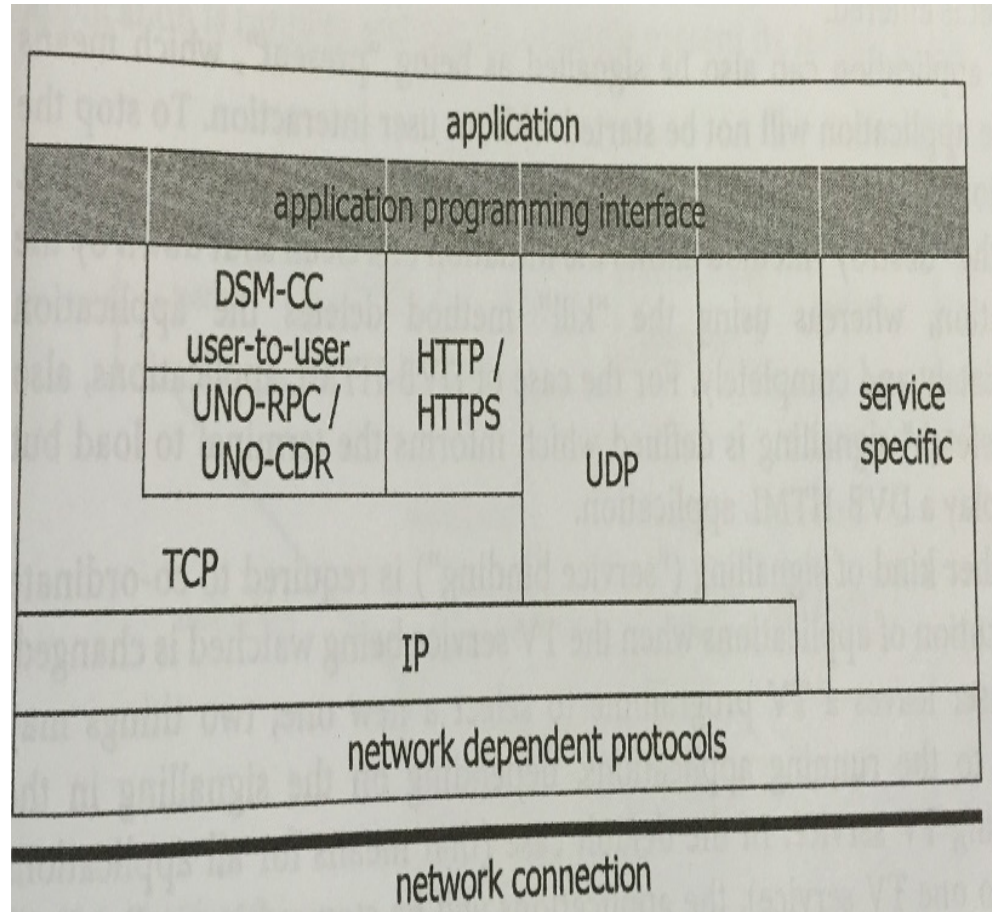
- MHP support large number of broadcast channels in order to access the DVB broadcasting channel as well as interaction channels
- Broadcast channel was the only way to download apps into the terminal in MHP1.0.
- Interaction channel is optional for the enhanced broadcast profile
 - Mandatory for all higher profile
- If available, can be used by an application for all kinds of remote server interaction and internet services.

Broadcast Channels Protocol Stack



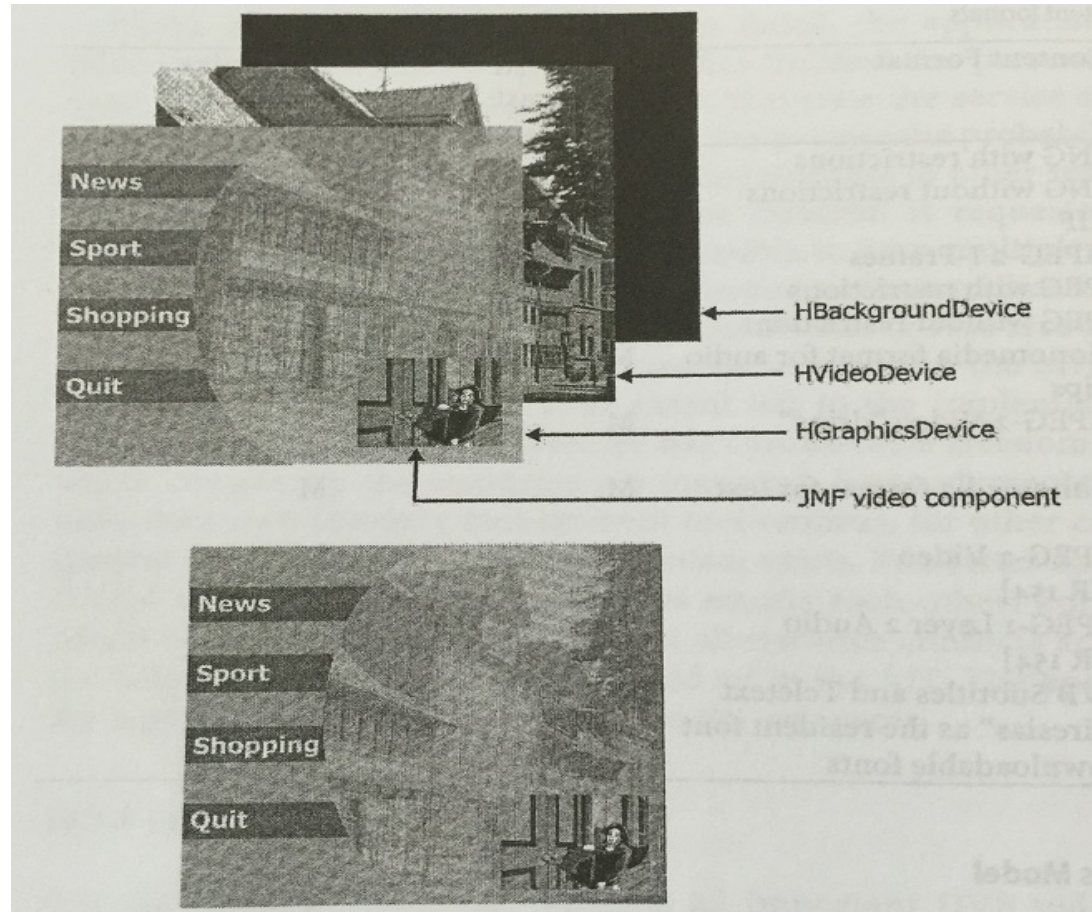
- MPEG-2 TS is used as transport medium for all kinds of digital data (Audio/video/apps)
- Terminal uses SI to distinguish between various types of data
- SI contains individual app details
 - Name & Signalling
- MHP apps transmitted via Object Carousel
- Content of object carousel is cached in the receiver.
- UDP is selected as the upper layer in the stack

Interaction Channel Protocols stack



- It is mainly based on the IP which may be used to transmit data across various kinds of network dependent protocols
- Choice of IP is mainly due to its wide acceptance and its' application is independent of return channel
- On top of it both TCP/UDP can be used
- Apps can also access user to user carousel encapsulated using UNO-RPC/CDR
- TCP is robust against packet loss in bad transmission
- HTTP/HTTPS for internet service.

Graphics Model



- Offers large number of different tools to display video, graphics, buttons and texts.
- Three graphics layer:
 - Background plane: for still images
 - Video plane: for running video sig.
 - Graphics plane (for visualising interactive components)
- Different layers can be combined with a set of different paint rules
- Layers can be transparent or semi transparent.
- MHP supports Java AWT (abstract windowing toolkit) graphics interface.
- Supports a number of different screen modes

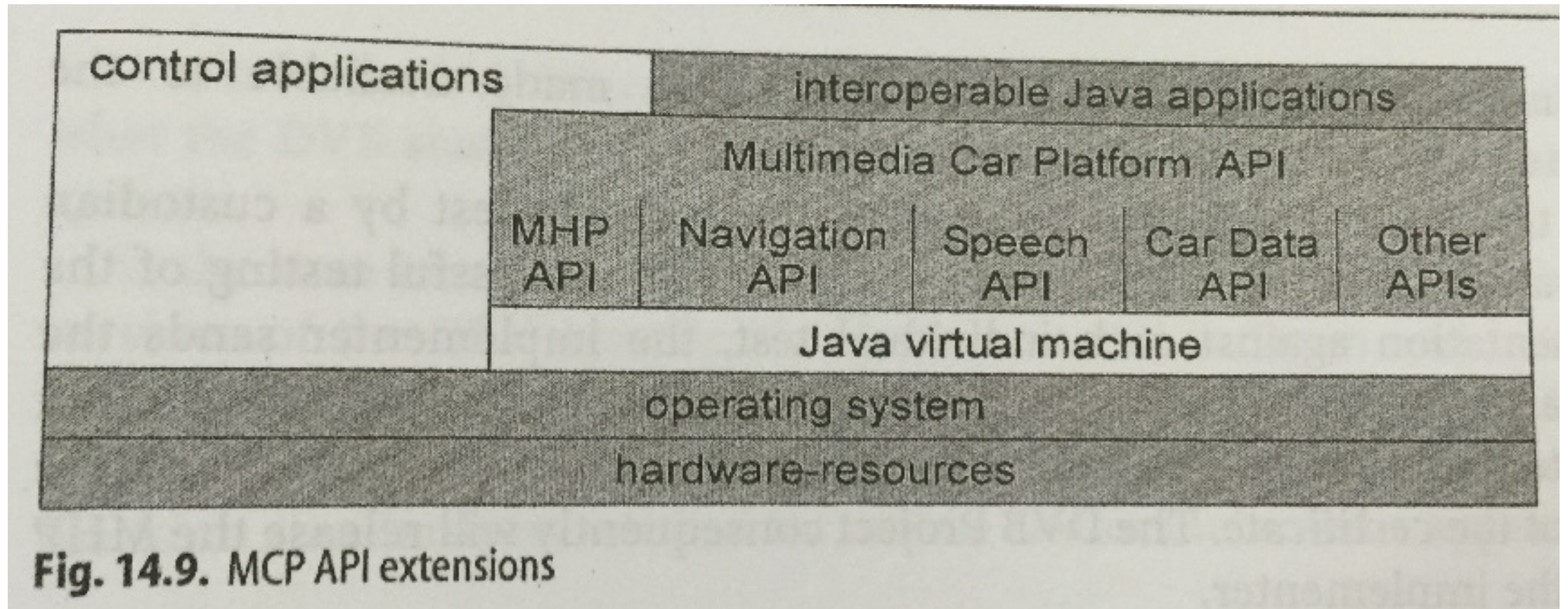
MHP1.1

Table 14.2. Content types accessible in DVB-HTML

MIME media type	common name
text/XML	XML
application/XML	
text/CSS	CSS
text/plain	monomedia format for text
text/dvb.utf8	
audio/mpeg	Monomedia format for audio clips or audio
image/jpeg	JPEG
image/png	PNG
image/gif	GIF
image/mpeg	MPEG-2 I-frames
video/dvb.mpeg.drip	MPEG-2 video drips
video/mpeg	MPEG-2 video
multipart/dvb.service	multipart DVB service
application/dvb.pfr	downloadable fonts
application/dvbj	initial class file of an Xlet
text/ecmascript	ECMAScript

- More advanced details of the internet access profile.
- DVB-HTML definition
- 100% compatible to all important XML languages
- User agents performs 3 types of processing on resources:
 - Decoding
 - Presentation and
 - Interaction
-

Future of MHP: Mobile MHP



Some Non-MHP solutions

- Advanced Television Enhancement Forum (ATVEF)
- A cross-industry group formed to specify standard for delivering:
 - Interactive TV applications that be authored one time using a variety of tools and deployed to a variety of TV sets, STB and PC based receivers.
- The specification basically describes HTML 4.0
 - Delivers TV program on both analogue and digital video systems
- The specification consists of 3 parts:
 - Content specification to establish minimum requirement for receivers
 - Delivery specifications for transport of enhanced TV content
 - A set of specific bindings

Some Non-MHP solutions

- Betanova
 - Developed by German company Beta Research.
 - Opened the software environment of d-box to C/C++ programmers
 - Later versions (Betanova 2.xx) is Java-compliant platform
 - It simplifies the adaptation of existing application to d-box.
- Liberate
 - Platform for delivering enhanced content and services to TV viewers
 - VoD/TV chat/
 - Java based software (also called TV navigator standard)

Some Non-MHP solutions

- Mediahighway
 - Java based
 - Compatible with Canal + technologies
 - Can incorporate apps created in different languages (HTML/MHEG-5)
- OpenTV
 - American company
 - Hardware independent, modular and extensible
 - Compatible with application developed in C/HTML/Java
 - C-code execution engine is provided as part of the basic package
 - Java/HTML execution engine is optional

Extension of MHEG-5

- MHEG-6
 - Additional Java environment to allow efficient processing of complex multimedia/hypermedia information
 - Near video on demand offering
 - Distributed network games etc.
- MHEG-7
 - Defines a test suite that can be used to test MHEG-5 engine's conformance to a specific application domain
 - Format for test cases that can be used to extend the test suite for more detailed testing and for the extension defined by the application domain
- MHEG-8: provides XML-encoding for MHEG5

Migration Concept

- To allow non-MHP based services to migrate to MHP
 - Needs to solve the problems of different transport protocols, content formats, application code and receiver capabilities
- Four possible options:
 - Simulcast: service providers broadcast legacy as well as MHP
 - Allows for a smooth transition with a variety of receivers (+)
 - Design and broadcast the same application for two systems in parallel (-)
 - Shared broadcast using MHP content
 - Similar to Simulcast but the content is shared
 - Legacy system has to be able to understand the MHP content
 - Has to have access to the transport protocol

Migration Concept

- Plug-in for the legacy system operating in a MHP receiver
 - A software running on top of MHP API capable of understanding interpreting legacy system
 - Adds complexity to the receiver (-)
 - Not sustainable option, needs to switch to MHP
- Clean cut:
 - hard switch from legacy system to MHP system at a determined date
 - Legacy API boxes will be unable to execute after the deadline as there will no longer be any application on the air