

# Software

- Introduction
- Programming
  - + Browser based software architecture
  - + Distributed software
- Components
  - + Servers
  - + Network
  - + Terminals
- Java
  - + JMF
  - + MIDP
  - + MHP
- XML
  - + X-Smiles
- Operating systems

# Networked multimedia

User interface

Middleware

Communications

- Network multimedia can be defined as multimedia applications and services, which are distributed into a network
- For example, www services, digital television, and WAP

# Causes of multimedia

- Multimedia has continuous media elements, which have to be synchronized
- Media is usually transferred as streams
- Inside the stream, the samples (audio sample, video frame, etc.) are in order
- Media streams can be synchronized
  - + Internal synchronization: isochronous
  - + External synchronization: synchronous

# Buffering of media streams

- It is impossible to synchronize media streams exactly
- Data transfer delay fluctuates, which causes jitter
- Human can detect even small fluctuation
- Fluctuation can be reduced with buffering
- Usually, buffering is required in several stages

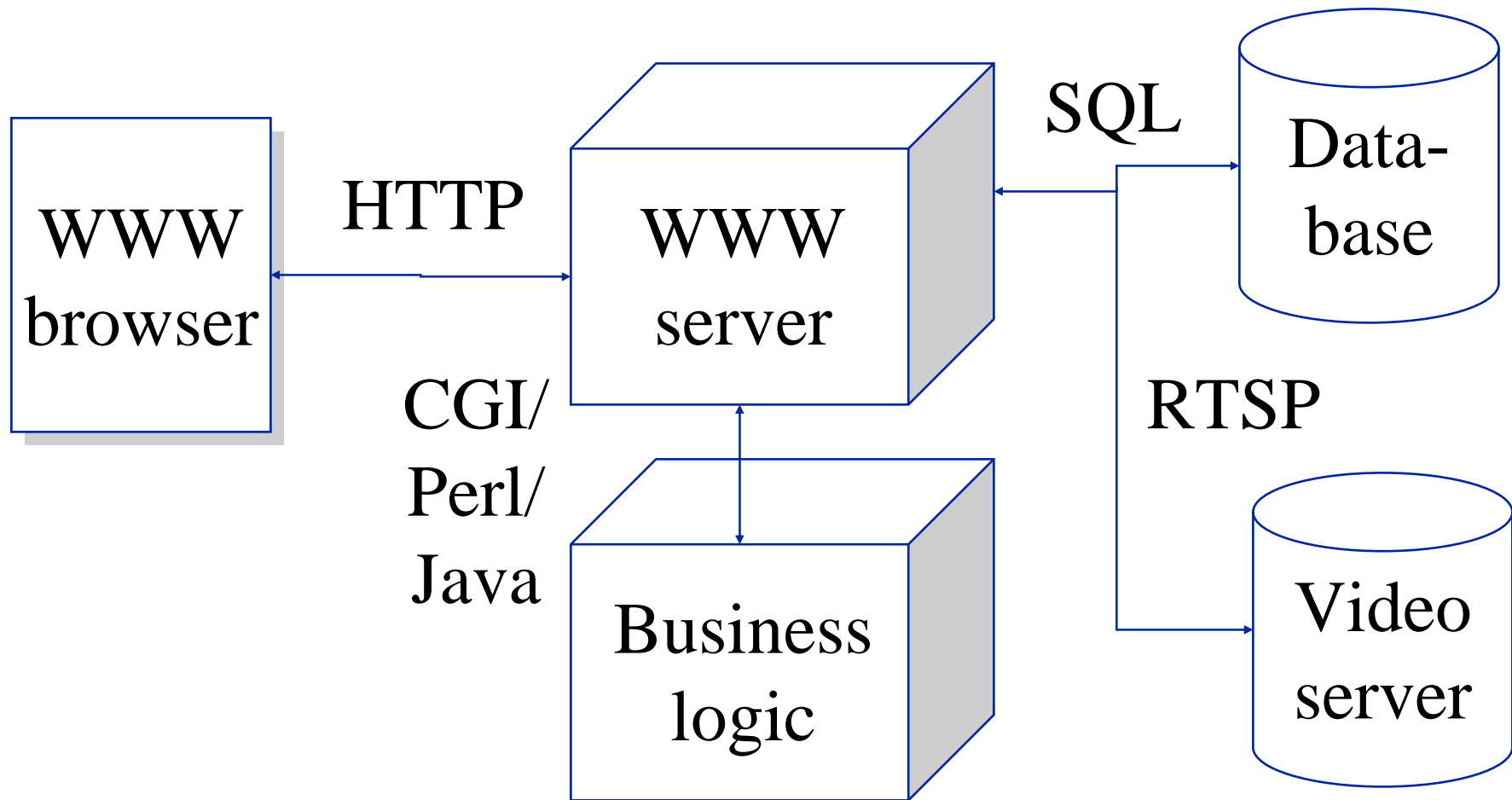
# Programming

- There are two ways to implement distributed multimedia system:
  - + client / server
  - + distributed software
- Browser software architecture is typical example of client / server architecture
- Distributed software means usually distributing a object oriented software into a network

# Browser based architecture

- Internet applications are based on client / server architecture
- WWW browser is a typical client program
- WWW server is a typical server
- There can also be other servers (database, video on demand, video conference, etc.)
- In addition infrastructure servers are needed (proxies, directories, etc.)

# WWW architecture

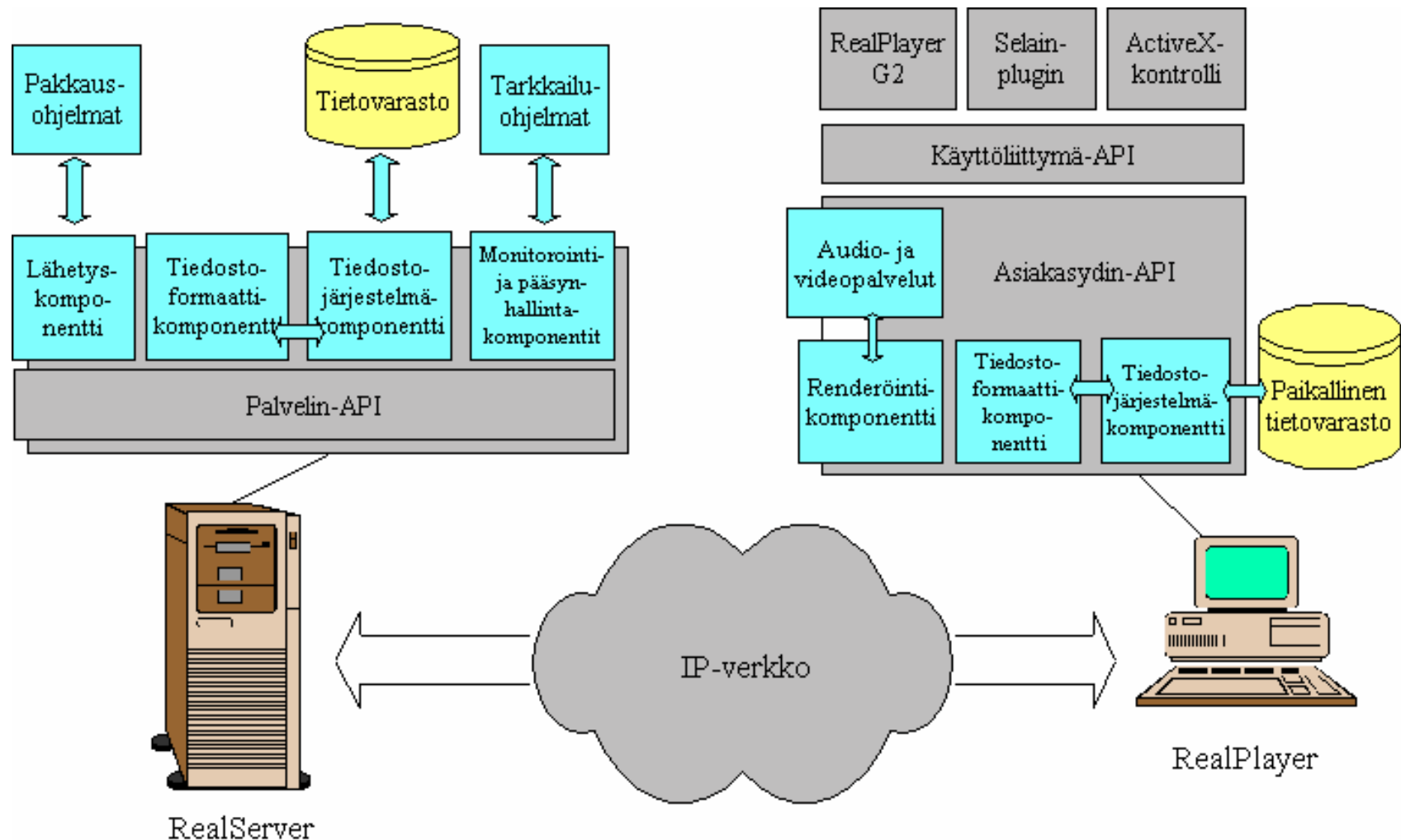


# RealSystem

- RealSystem is a typical example of client / server architecture
- The system can stream different media (audio, video, animations, etc.)
- The system is composed of development tools, intermediate servers, and client programs
- The basic version of the client software is free, but other software products are commercial



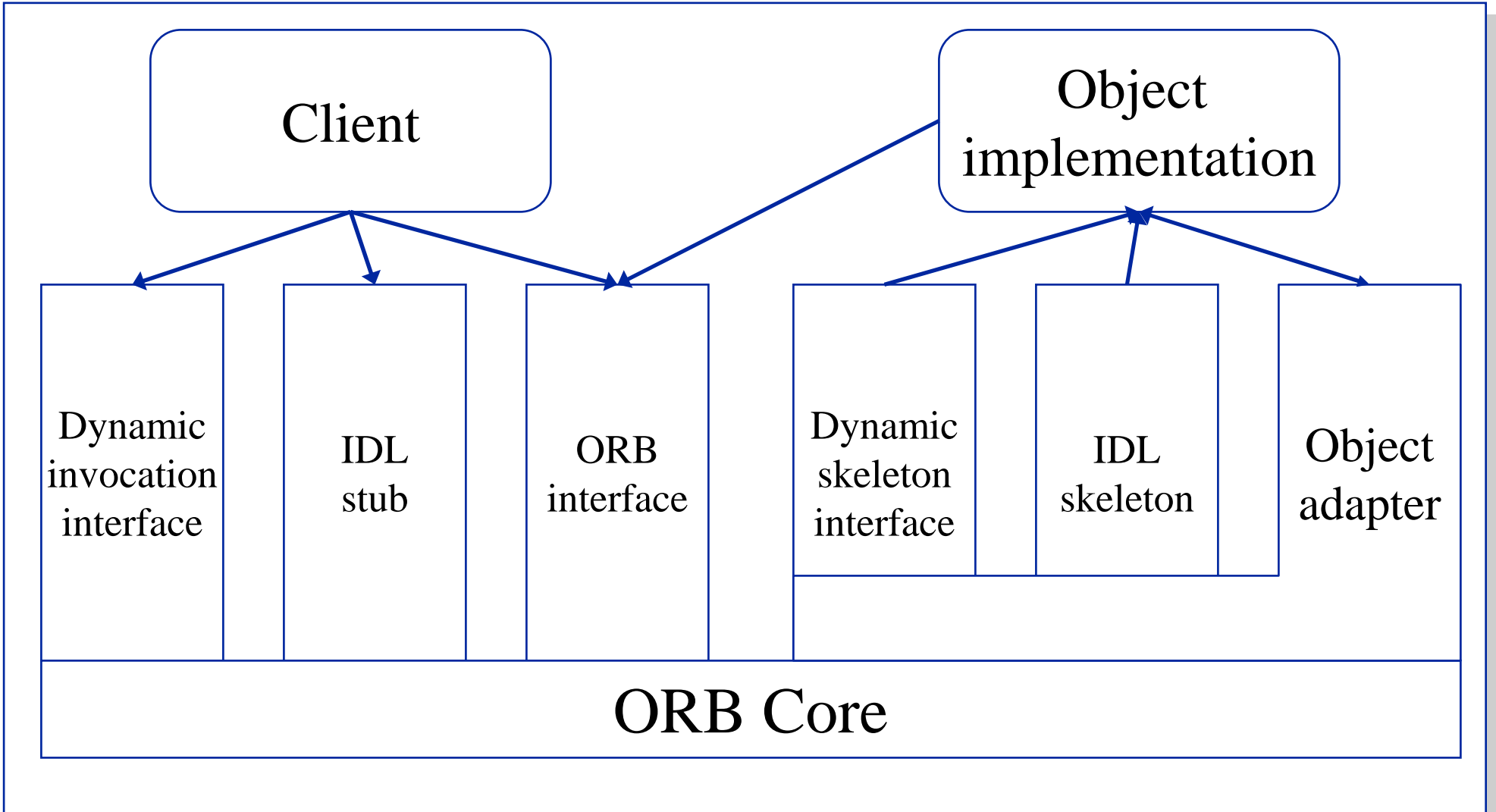
# RealSystem architecture



# Distributed software

- Object oriented software can easily be distributed to several computers
- Objects have to find each other somehow
  - + directory service
- The method calls and replies of the objects have to be forwarded to the right computer
  - + Object Request Broker (ORB)
- Commercial solutions are, e.g., OMG Corba, Microsoft DCOM & .NET, and Java RMI & Jini

# Corba architecture



# Components

- Client and object implementation talk via ORB
- The interfaces are defined with IDL (Interface Description Language)
- Normally, the calls are made to the static stub, which forwards the call via skeleton to the object
- The implementation can also be dynamic
- Object register with the help of the adapter

# Software architecture implementation

- The software architecture can be implemented also on the system component level
- Multimedia affects both servers, network, and terminals
- Efficient implementation requires additional features and even full redesign of some components

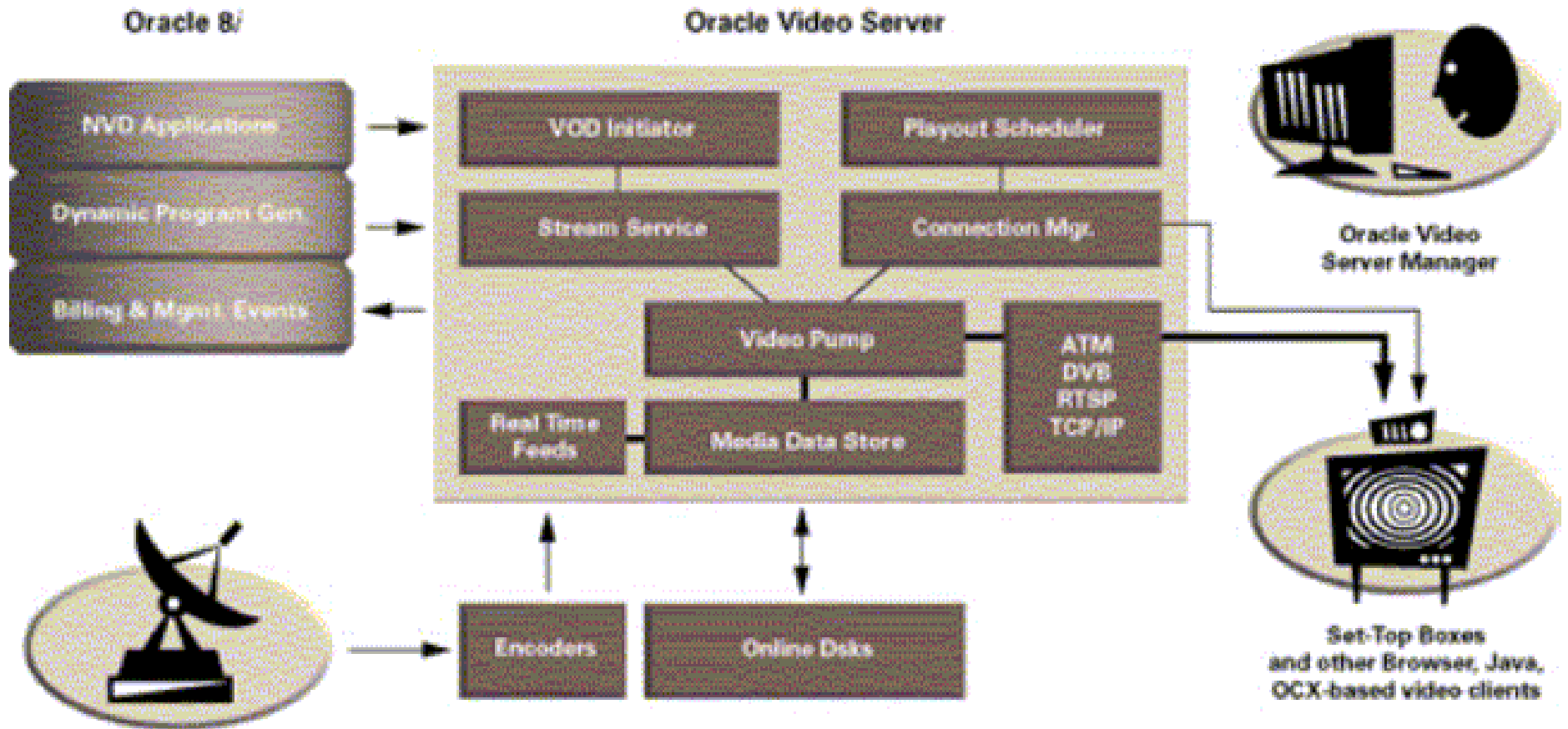
# Servers

- The servers can be classified into transaction (www, data base) and streaming servers (video)
- Scalability is most important feature of transaction servers
  - + if necessary, the task can be divided for several servers (replication)
- Most important feature of streaming servers is real-time support
  - + in practice, this means real-time scheduling

# Disk scheduling

- The hard disk usage of video-on-demand servers have to be designed carefully
- Usually, the disk seek time and space consumption is optimize
  - + thus data retrieval is based on fairness
- Video-on-demand server has to keep buffers full
  - + real-time scheduling algorithms
  - + relative location of files

# Oracle video server





# Network

- Network has to support multimedia transfer:
  - + Quality of Service (QoS)
  - + real-time media streams
  - + scalability
- Resources have to be reserved for multimedia traffic
  - + ATM QoS
  - + IP Integrated Services
  - + IP Differentiated Services

# Network (cont.)

- Network protocols have to support real-time media streams
  - + e.g., IP Real-Time Protocol (RTP), Real-Time Control Protocol (RTCP), and Real-Time Streaming Protocol (RTSP)
- In addition, the network has to support several simultaneous users
  - + e.g., IP Multicast

# Terminals

- The biggest problem of terminals is limited resources:
  - + processing power, memory, and communications
- Current operating systems support primarily graphical user interface
- Support for multimedia is more limited
- Real-time operating system support better multimedia

# Multimedia processing steps

- Receiving of packets
- Network drivers
- IP/UDP protocol
- Real-time transfer protocols
- Codec
- Player
- Windowing system
- Audio, video, etc. drivers

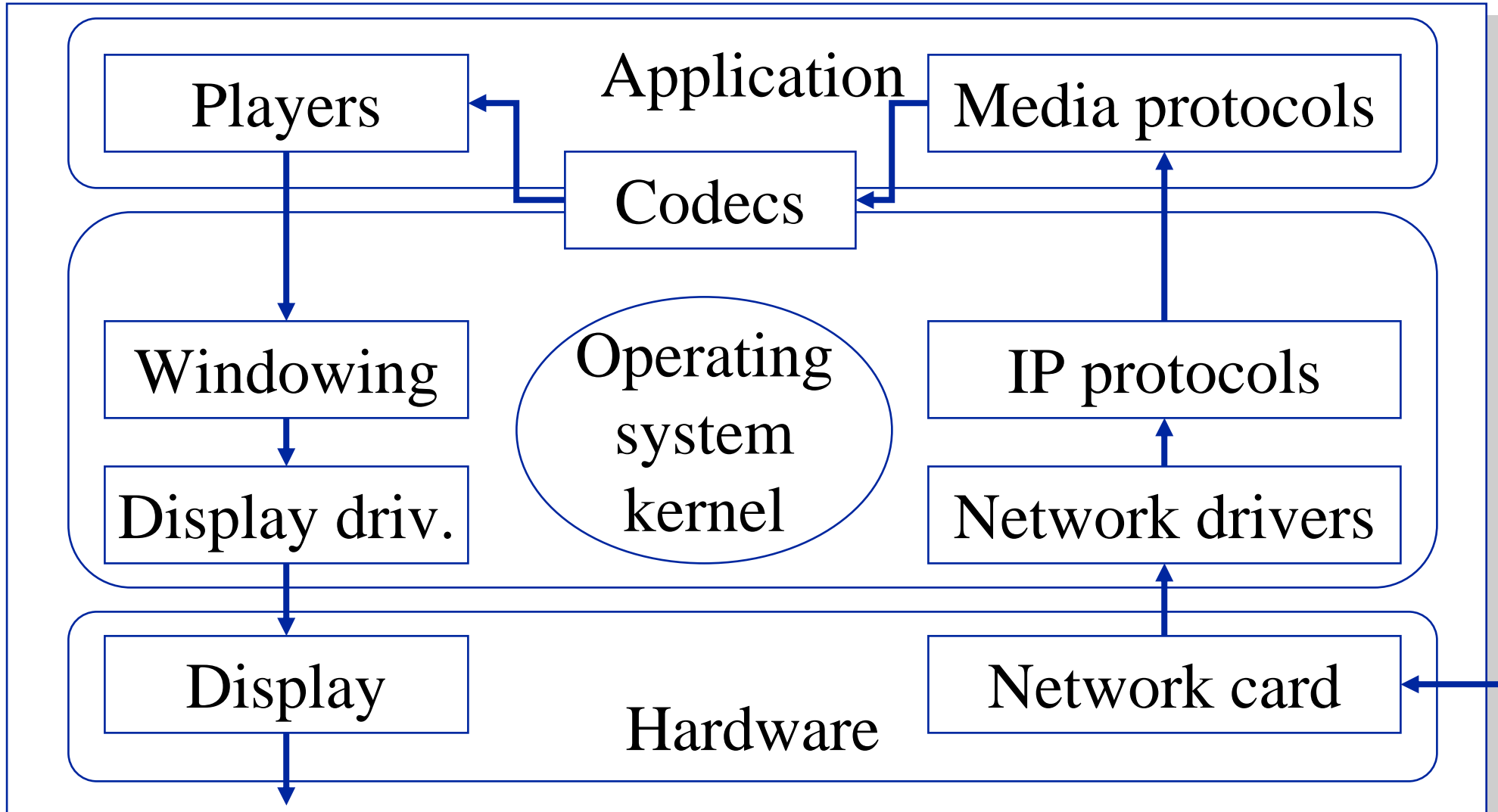
# User interaction

- The terminal also has to track the user
  - + keyboard
  - + mouse
  - + etc.
- The devices create interrupts
- The interrupts can easily jam with the network, etc. interrupts

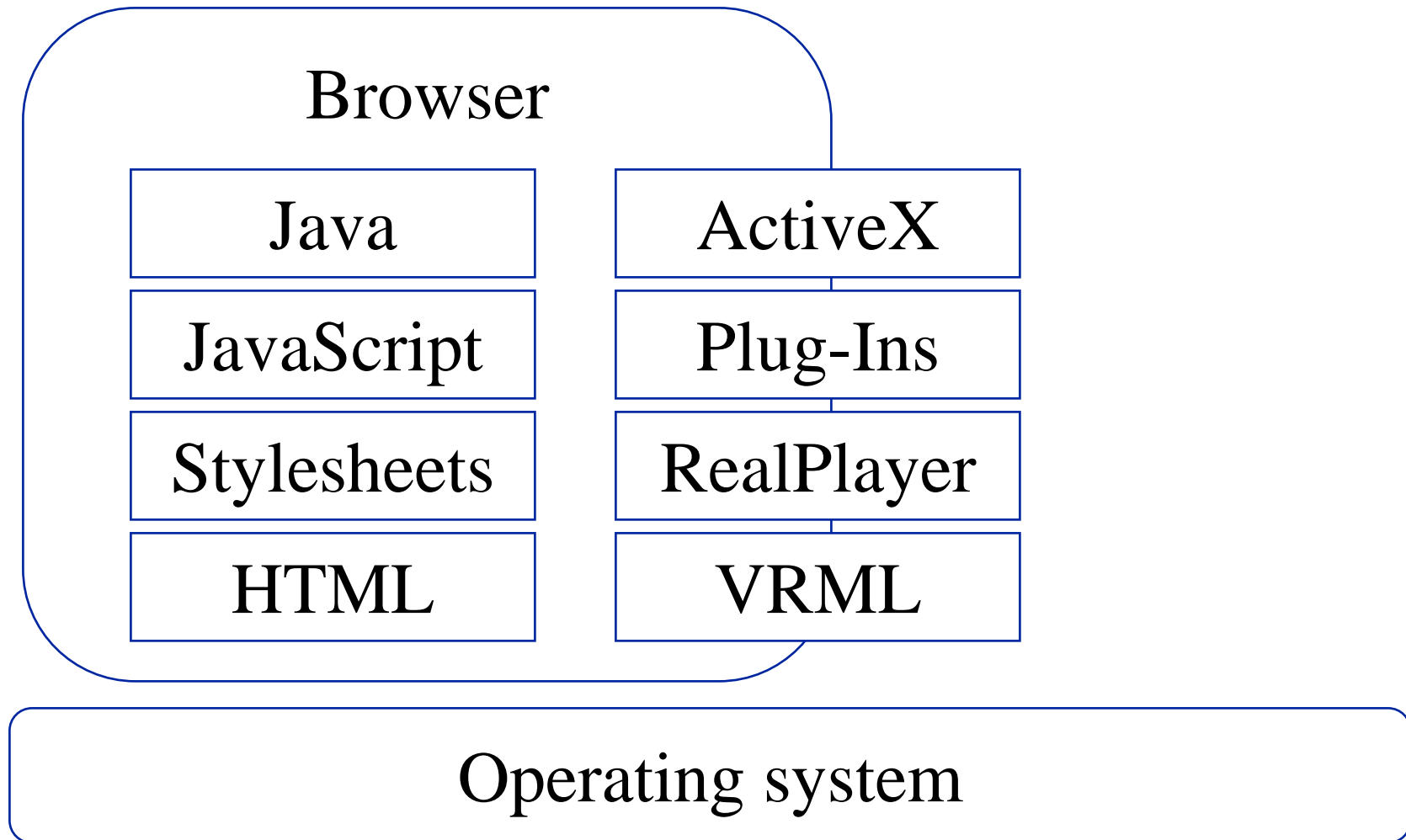
# Terminal software architecture

- The terminal software architecture can be implemented with several ways
  - + operating system + drivers + windowing
  - + browser + plug-in players
  - + Java
  - + XML browser

# Operating system

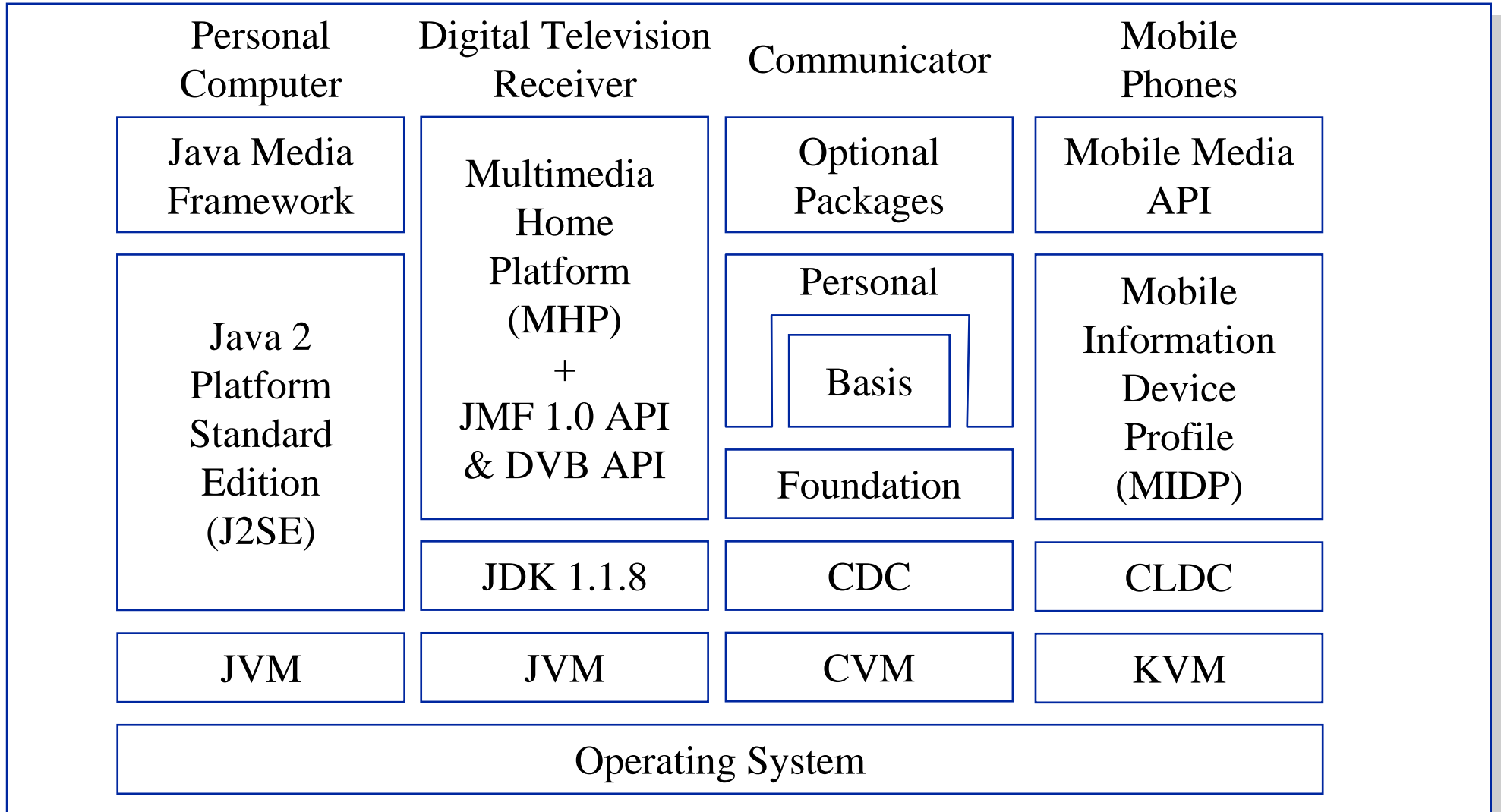


# Browser





# Java



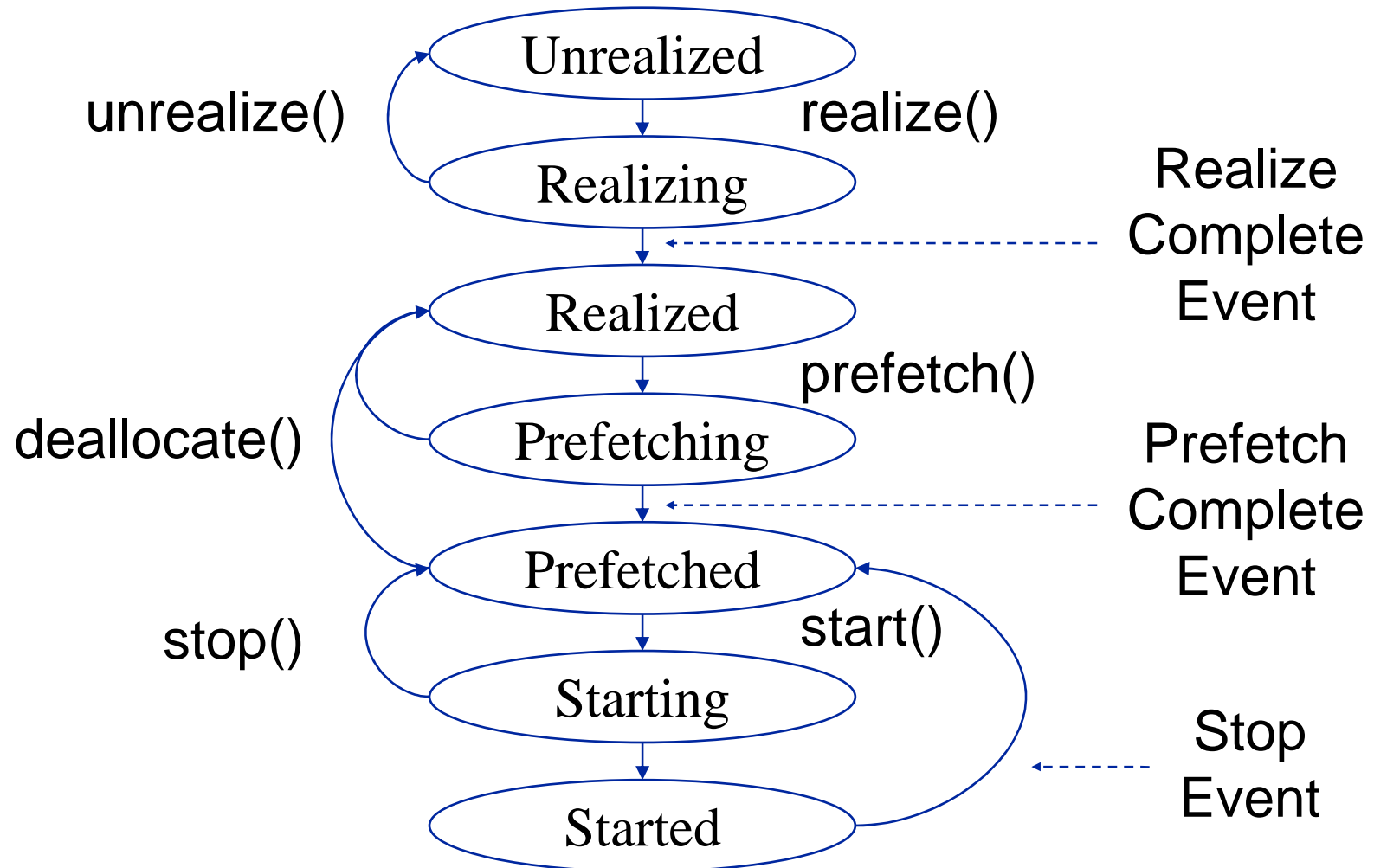
# Device Configurations

- **Connected Limited Device Configuration (CLDC)**
  - + Slow processor (16/32 bit, 25 MHz)
  - + Little memory (min 192 KB)
- **Connected Device Configuration (CDC)**
  - + Faster processor (32 bit)
  - + More memory (min 2 MB)

# Java Media Framework

- JMF allows use of multimedia in Java applications
- Real-time network protocols
- Multiplexing
- Codecs
- Players
- Effects
- Capture
- Control

# JMF State Diagram



# Java MIDP

- Mobile Information Device Profile (MIDP)
- Intended for small devices
  - + Mobile phones etc.
- Consists of Java Virtual Machine (JVM) and Application Programming Interfaces (API)
- Latest version 2.0

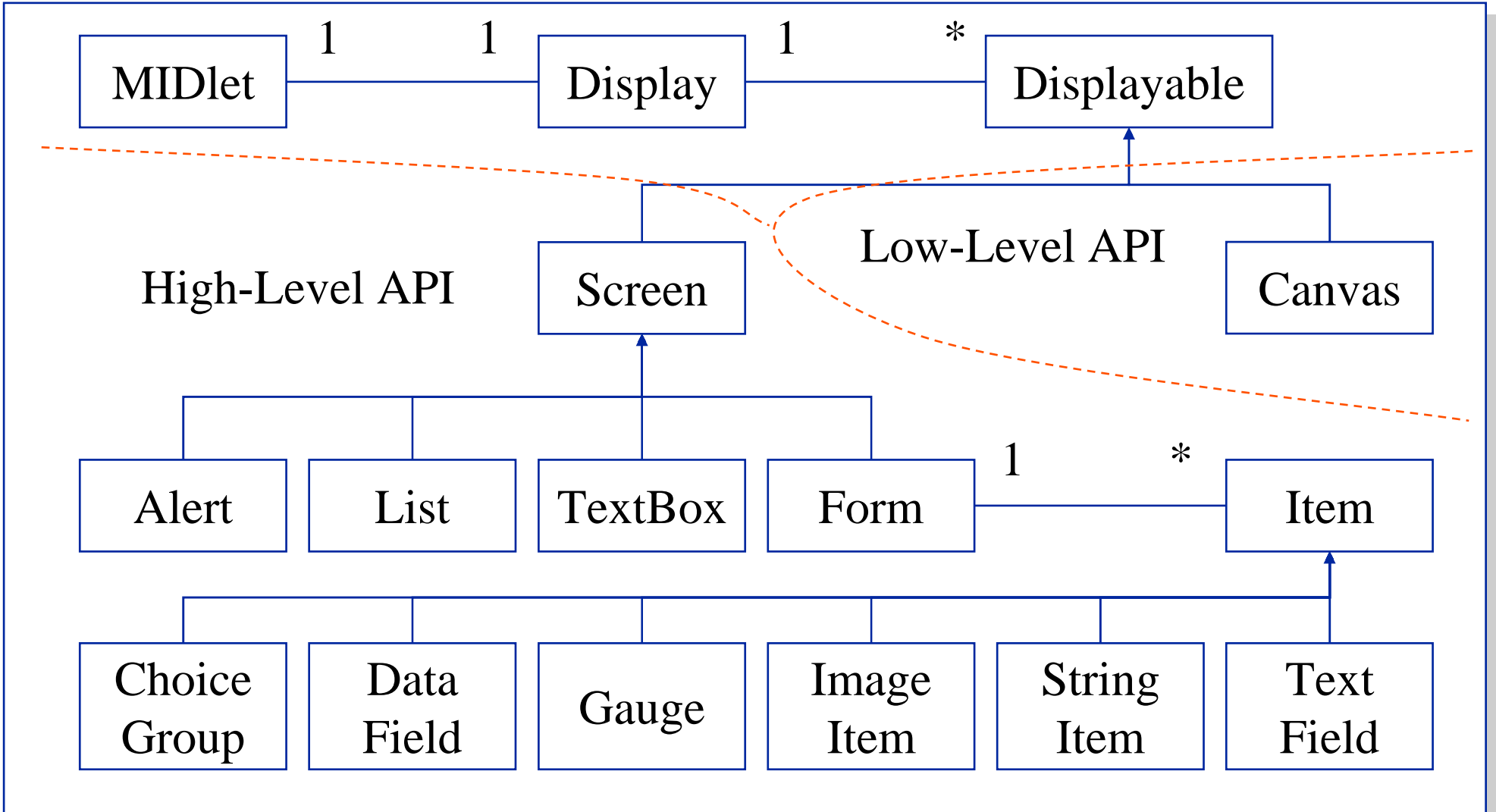
# MIDP Requirements

- Minimum requirements are:
  - + Display 96x54, 1 bit (b/w)
  - + 256 KB memory for MIDP implementation
  - + 8 KB application memory
  - + Two directional network
  - + Audio

# MIDP basic APIs

- Basic packets (java.lang & java.util)
- Network (javax.microedition.io)
- Application lifecycle (javax.microedition.midlet)
- Data storage (javax.microedition.rms)
- User interface (javax.microedition.lcdui & javax.microedition.lcdui.game)
- Certificates (javax.microedition.pki)
- Multimedia (java.microedition.media & javax.microedition.media.control)

# MIDP User Interfaces





# Extensions

- Java Bluetooth API (JSR-82)
- Java Wireless Messaging API (JSR-120)
- Java Mobile Media API (JSR-135)
- Nokia UI API

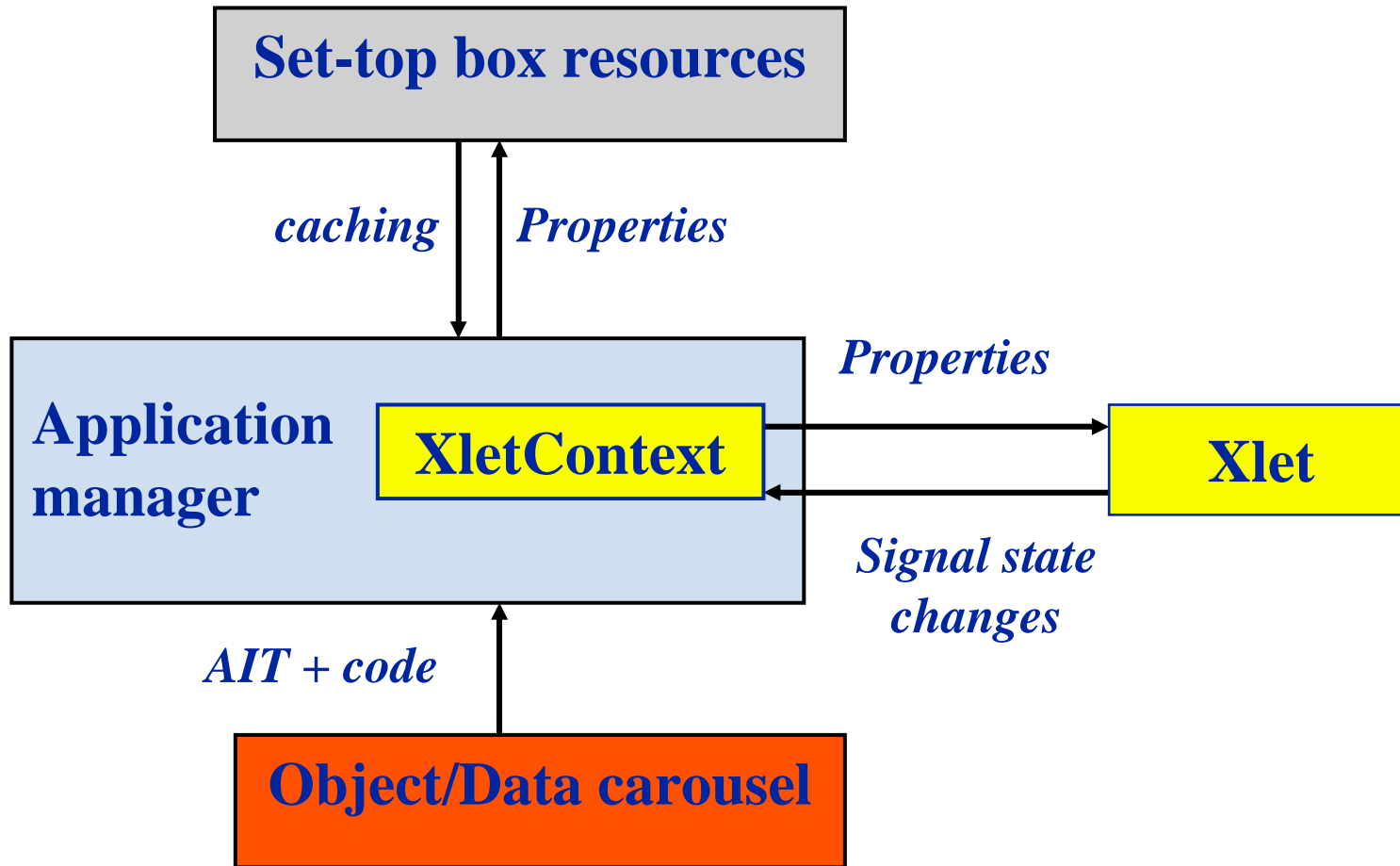
# Mobile Multimedia API

- Mobile Media (MMAPI) version 1.0  
+ [java.sun.com/products/mmapi](http://java.sun.com/products/mmapi)
- Support for continuous media (e.g., audio, video) in MIDP
- Platform independent thin Java layer
- Not the same thing as JMF
- Optional package
- MIDP 2.0 includes only audio part of MMAPPI

# Multimedia Home Platform

- One example of Java environment is the Multimedia Home Platform (MHP) of digital television
- The environment is composed of Java, JMF, TV, net, etc. APIs
- Applications are called Xlets
- The Xlets are transmitted via broadcast network
- Data is transferred through so called data- and object carousels

# Application Manager



MONDAY 29.04. 10:25

CHANNELS

- 08 NBC
- 09 BBC
- 10 CCK
- 11 R&P
- 12 CableCon
- 13 Ravouh
- 14 Australian TV
- 15 Thumba TV
- 16 Deutch Well
- 17 Asiansport
- 18 KPC
- 19 Adult Channel
- 20 Canal+



TODAY 29.04.

- 10.25 Coming Home
- 11.00 Think First
- 11.10 News and Weather
- 11.20 Meeting Joe
- 13.30 Final Fantasy
- 15.45 Making of Final Fantasy

CONTENT

DAY

OPTIONS



▶ UUTISET

**Suomen Kuvalahti korvauksiin Kiteen Pal..**

Irakissa kovia maataisteluita

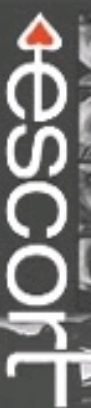
Novo lomauttaa n. 2000 henkilöä

ZZ Top mukaan The Rolling Stonesin ko..

▶ SUBTV

▶ URHEILUKANAVA





▶ URHEILU

Dallas Stars taistelee lännen kärkipaikasta

Lehtitieto: Real suunnittelee Beckharnille..

▶ AJANKOHTAISTA

Enemmän Hockey Nightiä MTV3+ :lta

Andyn värikäs elämä ruutuun

Lisätietoa digitaalisista palveluista

VEIKKAUS



# XML

- Metalanguage for defining markup languages
  - + XML languages can be used either for presenting or processing of content
- Processing means, e.g., retrieval or filtering of XML data
- Presenting means presenting of XML content in different kinds of terminals
- Often, Cascading Style Sheets (CSS) and ECMAScript language are used in addition

# Processing Languages

- eXtensible Stylesheet Language Transformations (XSLT) – XML data transformations
- XML Schema – Data model
- XPath – Element references



# Presentation Languages

- XSL Formatting Objects (XSL FO) – Formatting
- eXtensible HyperText Markup Language (XHTML) – HTML 4.01 in XML format
- Synchronized Multimedia Integration Language (SMIL) – Synchronized multimedia language
- Scalable Vector Graphics (SVG) – Vector graphics
- X3D – 3D graphics
- VoiceXML – Voice control

# Processing of XML data

- Parsing of XML file
- Parsing of XSL style sheet
- Transformation of XML according to XSL style sheet
- Document Object Model (DOM) data structure
- Execution of start-up scripts
- Building of presentation language layout model
- Rendering of presentation language
- Execution of event handler scripts

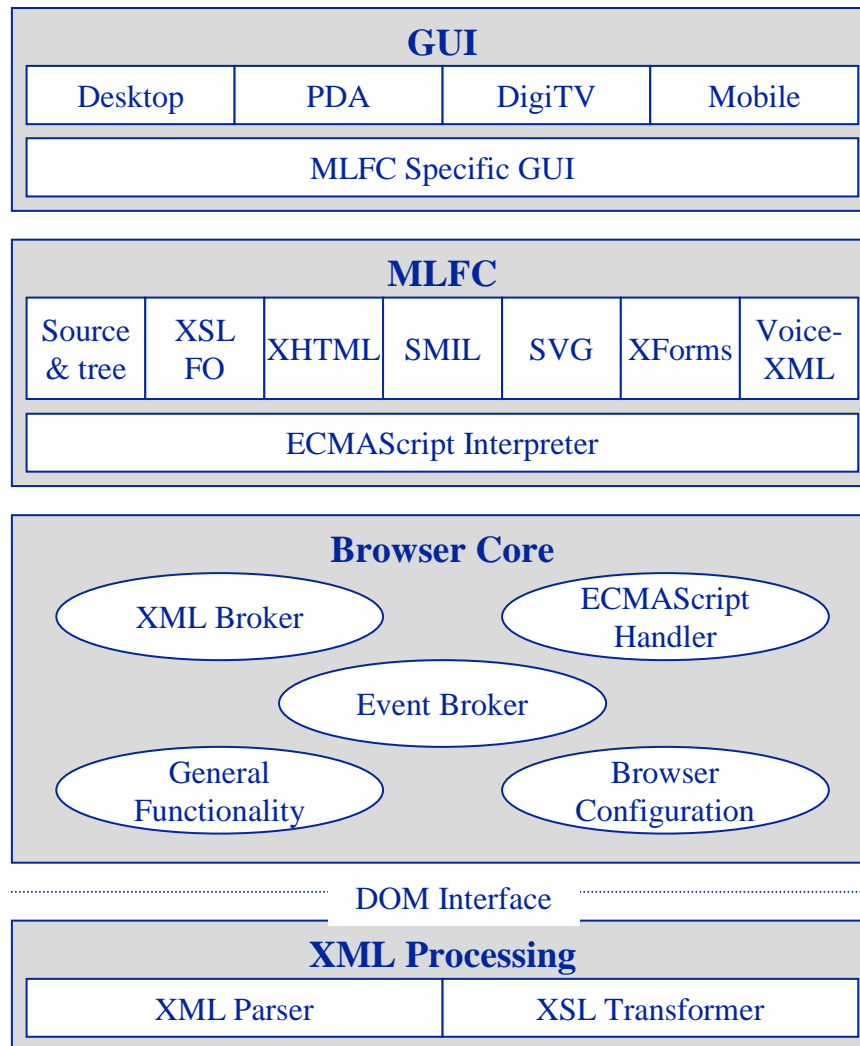
# XML Components

- XML parser
- XSL transformer
- DOM interface
- ECMAScript interpreter
- XPath engine
- XML Schema engine
- CSS layout

# X-Smiles History

- Development of X-Smiles started as student software project in 1998
- Later development work has continued in various research projects
  - + GO
  - + XML Devices
- Published as open source code in 2001
  - + [www.x-smiles.org](http://www.x-smiles.org)

# X-Smiles Architecture



# Operating systems

- In current software architectures, operating system has very important role
- Most of the media processing is done by calling system software
- Application acts as coordinator
- Data copying between application and operating systems causes problems
- Operating systems are poor resource managers

# Symbian OS

<p><b>Application engines</b> Phone book, calendar, photo album, notepad, to-do, pinboard</p>	<p><b>Installed applications</b> Games, self-developed applications, utility software</p>	<p><b>Messaging</b> SMS, MMS, e-mail, fax</p>	<p><b>MIDP</b> Java KVM</p>
<p><b>Application framework</b> GUI framework (Avkon, standard Eikon, Uikon) application launching, and graphics</p>		<p><b>Personal area networking</b> Bluetooth, infrared</p>	
<p><b>Multimedia</b> Images, sounds, graphics</p>		<p><b>Communication infrastructure</b> TCP/IP stack, WAP stack</p>	
<p><b>Security</b> Cryptography, software</p>		<p><b>Telephony</b> HSCSD, GPRS</p>	
<p><b>Base</b> User library kernel, device drivers</p>			

# Real-time operating system

- **Small size:** extra features have been removed
- **Interrupt processing:** jamming prevented
- **Real-time scheduling:** time of task processing
- **Memory management:** shared memory
- **Message forwarding:** communication between different levels is fast
- **Resource reservation:** task will not be accepted, if resources are not available



# Microkernel

- Operating system becomes more compact, if all “unnecessary” features are removed
  - + windowing, share file systems, etc.
- One solution is use so called Microkernel
  - + the kernel of the operating system is as small as possible and real-time
  - + other features are implemented as user processes
  - + the required system can be composed from different components

# Embedded Linux

- Linux is an interesting alternative also in embedded devices
- Unnecessary features (e.g., X-Windows) can be removed so that memory consumption is reduced
- Implementation of real-time features in kernel is very difficult, but not always necessary
- For example, in set-top box hardware takes care of video and audio processing