

Ohjelmistoarkkitehtuuri

- Johdanto
- Ohjelmointi
 - + Selain-ohjelmistoarkkitehtuuri
 - + Hajautetut ohjelmistot
- Ohjelmistoarkkitehtuurin toteutus
 - + Palvelimet
 - + Verkko
 - + Päätelaitteet
- Java
 - + JMF
 - + MIDP
 - + MHP
- XML
 - + X-Smiles
- Käyttöjärjestelmät

Petri Vuorimaa

1

Verkotettu multimedia

Käyttöliittymä

Middleware

Tietoliikenne

- Verkotetulla multimedialla tarkoitetaan multimedian sovelluksia ja palveluja, jotka on hajautettu verkkoon
- Esim. www-palvelut, digitaalinen televisio, WAP

Petri Vuorimaa

2

Multimedian vaikutukset

- Multimedia sisältää jatkuva-aikaisia mediaelementtejä, jotka on synkronoitu keskenään
- Media siirretään yleensä mediavirtoina (stream)
- Virrassa mediaelementin yksiköt (ääninäyte, videokuva ym.) pysyvät järjestyksessä
- Mediavirrat voidaan synkronoida
 - + Sisäinen synkronointi: isochronous
 - + Mediavirtojen välinen synkronointi: synchronous

Petri Vuorimaa

3

Mediavirtojen puskurointi

- Mediavirtoja ei voi koskaan täydellisesti synkronoida
- Tiedonsiirron viive vaihtelee eli esiintyy huojuntaa
- Ihminen havaitsee vähäisenkin huojunnan
- Huojuntaa voidaan vähentää puskuroinnilla
- Puskurointia tarvitaan useassa eri vaiheessa

Petri Vuorimaa

4

Ohjelmointi

- Hajautettu multimediajärjestelmä voidaan toteuttaa kahdella eri tavalla:
 - + Asiakas / palvelin
 - + Hajautettu ohjelmisto
- Selain-ohjelmistoarkkitehtuuri on tyypillinen esimerkki asiakas/palvelin -arkkitehtuurista
- Hajautettu ohjelmisto tarkoittaa tyypillisesti olio-ohjelman hajauttamista verkkoon

Petri Vuorimaa

5

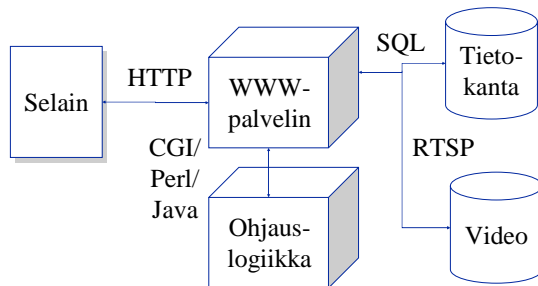
Selain-ohjelmistoarkkitehtuuri

- Internet ohjelmistot perustuvat asiakas/palvelin -arkkitehtuuriin
- Tyypillinen asiakasohjelma on www-selain
- Tyypillinen palvelin on www-palvelin
- Lisäksi on myös muita palvelimia (tietokanta, video on demand, videoneuvottelu jne.)
- Lisäksi tarvitaan apupalvelimia (proxyt, hakemistot jne.)

Petri Vuorimaa

6

WWW-arkkitehtuuri



Petri Vuorimaa

7

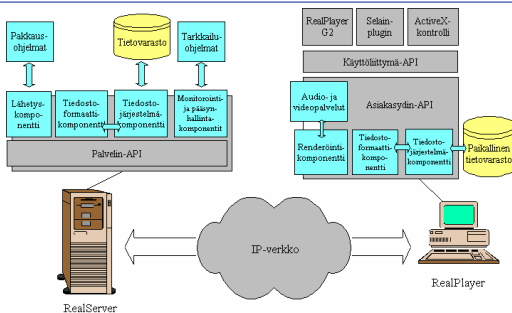
Realmedia

- Realmedia on tyypillinen esimerkki asiakas / palvelin -arkkitehtuurista
- Järjestelmän avulla voi streamata eri medioita (ääni, video, grafiikka jne.)
- Järjestelmä koostuu tuotantotyökaluista, palvelimista, välipalvelimista ja asiakasohjelmista
- Asiakasohjelman perusversiot ovat ilmaisia, mutta muut maksullisia tuotteita

Petri Vuorimaa

8

Realmedia-arkkitehtuuri



Petri Vuorimaa

9

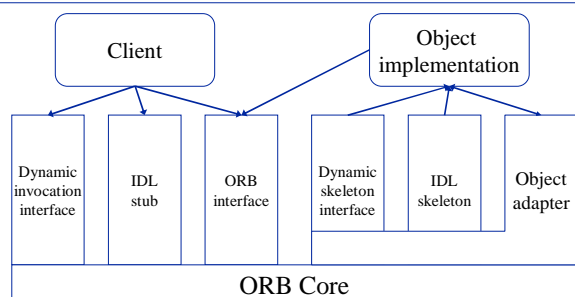
Hajautetut ohjelmistot

- Oliopohjainen ohjelma voidaan hajauttaa usealle eri koneelle
- Olioiden pitää jotenkin löytää toisensa
+ Hakemistopalvelu
- Olioiden metodikutsut ja -vastaukset pitää välittää oikealle koneelle
+ Object Request Broker (ORB)
- Kaupallisia ratkaisuja ovat esim. OMG Corba, Microsoft DCOM & .NET sekä Java RMI & Jini

Petri Vuorimaa

10

Corba-arkkitehtuuri



Petri Vuorimaa

11

Komponentit

- Asiakas (client) ja hajautettu olio (object implementation) keskustelevat ORB:n välityksellä
- Rajapinnat on määritelty IDL-kielellä (Interface Description Language)
- Normaalisti kutsutaan staattista tynkää (stub), joka välittää kutsun rungon (skeleton) kautta oliolle
- Toteutus voi olla myös dynaaminen
- Oliot rekisteröityvät sovittimen (adapter) avulla

Petri Vuorimaa

12

Ohjelmistoarkkitehtuurin toteutus

- Ohjelmistoarkkitehtuurin toteutusta voidaan tarkastella myös järjestelmän komponenttitasolla
- Multimedialla on vaikutusta sekä palvelimiin, verkkoon että päätelaitteisiin
- Tehokas toteutus vaatii lisäominaisuuksia ja jopa em. komponenttien täydellistä uudelleen suunnittelua

Petri Vuorimaa

13

Palvelimet

- Palvelimet voidaan jakaa transaktio- (www, tietokanta) ja stream-palvelimiin (video)
- Transaktio-palvelimen tärkein ominaisuus on skaalattavuus
- Tarvittaessa tehtävä voidaan jakaa useammalle palvelimelle (replikointi)
- Videopalvelimissa korostuu reaaliaikaisuus
- Käytännössä tämä tarkoittaa tehtävien skedulointia

Petri Vuorimaa

14

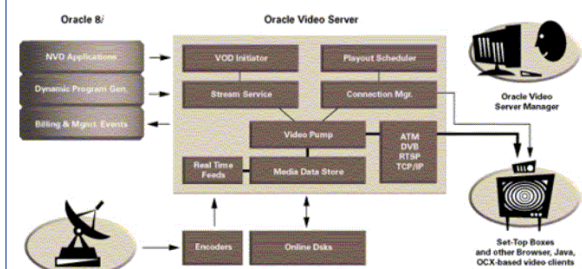
Levyskudulointi

- Video-on-demand -palvelimen kovalevyjen käyttö tulee miettiä tarkkaan
- Tavallinen tiedostopalvelin pyrkii optimoimaan haku-aikaa ja tilan käyttöä
 - + tällöin haut suoritetaan reiluus-periaatteella
- Videopalvelimen on pidettävä puskurit sopivan täynnä
 - + reaaliaikaiset skedulointi-algoritmit
 - + tiedostojen järjestys

Petri Vuorimaa

15

Oracle Video Server



Petri Vuorimaa

16

Verkko

- Verkon pitää tukea multimedian siirtoa:
 - + Quality of Service (QoS)
 - + reaaliaikaiset mediavirrat
 - + skaalattavuus
- Mediavirtoja varten on varattava tietyt resurssit
 - + ATM QoS
 - + IP Intergrated Services
 - + IP Differentiated Services

Petri Vuorimaa

17

Verkko (jatk.)

- Verkkoprotokollien on tuettava reaaliaikaisia mediavirtoja
 - + esim. IP Real-Time Protocol (RTP), Real-Time Control Protocol (RTCP) ja Real-Time Streaming Protocol (RTSP)
- Lisäksi verkon on tuettava useita yhtäaikaista käyttäjiä
 - + esim. IP Multicast

Petri Vuorimaa

18

Päätelaitteet

- Päätelaitteen suurin ongelma on rajalliset resurssit:
 - + Prosessori-aika, muisti ja tietoliikenneyhteydet
- Nykyiset päätelaitteiden käyttöjärjestelmät tukevat vuorovaikutteisia käyttöliittymiä
- Multimedian tuki on heikompi
- Reaaliaikaiset käyttöjärjestelmät sopivat paremmin multimedialle

Mediavirran käsittelyvaiheet

- Pakettien vastaanotto
- Verkkoajurit
- IP/UDP -protokollat
- Reaaliaikaiset siirtoprotokollat
- Koodekki
- Audio, video ym. ajurit
- Toisto

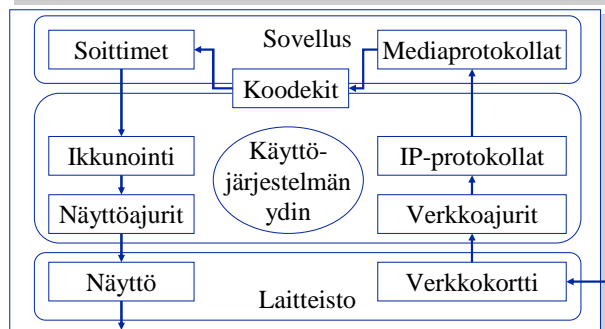
Käyttäjän vuorovaikutus

- Päätelaitteet myös seuraa käyttäjän toimenpiteitä
 - + Näppäimistö
 - + Hiiri
 - + Jne.
- Laitteet aiheuttavat keskeytyksiä
- Keskeytykset voivat helposti ruuhkautua verkkokeskeytysten ym. kanssa

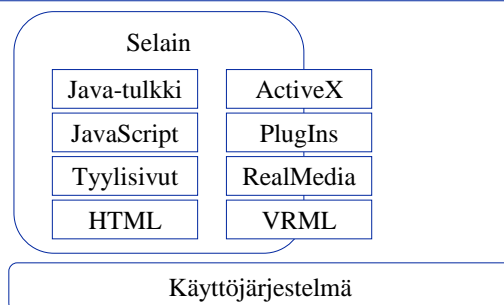
Ohjelmistoarkkitehtuuri

- Päätelaitteen ohjelmistoarkkitehtuuri voidaan toteuttaa monella eri tavalla
 - + Käyttöjärjestelmä + ajurit + ikkunointi
 - + Selain + Plug-In soittimet
 - + Java
 - + XML-selain

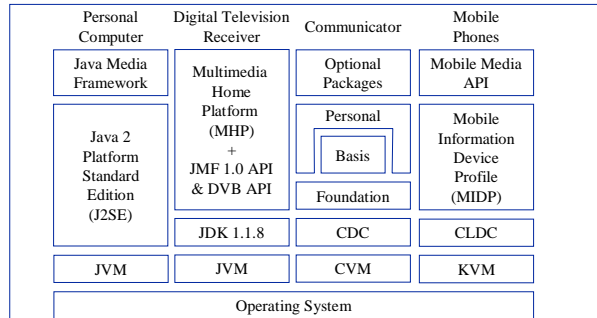
Käyttöjärjestelmä



Selain



Java



Petri Vuorimaa

25

Päätelaitteiden konfiguraatiot

- **Connected Limited Device Configuration (CLDC)**
 - + Hidas prosessori (16/32 bit, 25 MHz)
 - + Vähän muistia (min 192 KB)
- **Connected Device Configuration (CDC)**
 - + Nopeampi prosessori (32 bit)
 - + Enemmän muistia (min 2 MB)

Petri Vuorimaa

26

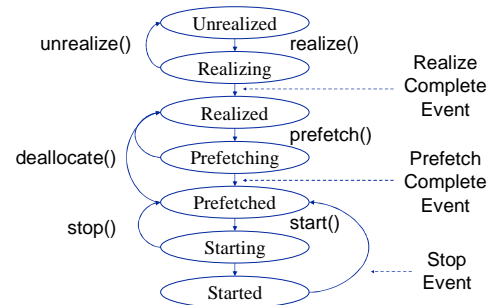
Java Media Framework

- JMF mahdollistaa multimedian Java-sovelluksissa
- Reaaliaikaiset verkkoprotokollat
- Multipleksaus
- Koodekit
- Toistimet (renderer)
- Efektit
- Kaappaus
- Kontrolli

Petri Vuorimaa

27

JMF Tilakaavio



Petri Vuorimaa

28

Java MIDP

- Mobile Information Device Profile (MIDP)
- Tarkoitettu pienitehoisille kulutuslaitteille
 - + matkapuhelimet ym.
- Koostuu Java-virtuaalikoneesta ja joukosta ohjelmointirajapintoja (API)
- Viimeisin versio 2.0

Petri Vuorimaa

29

MIDP vaatimukset

- **Minimivaatimukset ovat:**
 - + Näyttö 96x54, 1 bit (mv)
 - + 256 KB muistia MIDP-toteutukselle
 - + 8 KB sovellusmuistia
 - + Kaksisuuntainen verkkoyhteys
 - + Audio

Petri Vuorimaa

30

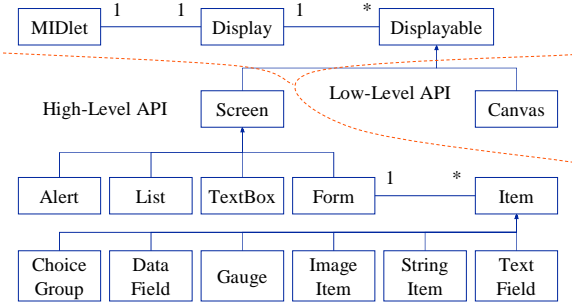
MIDP-ohjelmointirajapinnat

- Peruspaketit (java.lang & java.util)
- Verkko (javax.microedition.io)
- Sovellusten elinkaari (javax.microedition.midlet)
- Datan tallennus (javax.microedition.rms)
- Käyttöliittymä (javax.microedition.lcdui & javax.microedition.lcdui.game)
- Sertifikaatit (javax.microedition.pki)
- Multimedia (java.microedition.media & javax.microedition.media.control)

Petri Vuorimaa

31

MIDP-käyttöliittymät



Petri Vuorimaa

32

Laajennukset

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

Petri Vuorimaa

33

Mobile Multimedia API

- Mobile Media (MMAPI) version 1.0
+ java.sun.com/products/mmapi
- Audion, videon ym. jatkuva-aikaisen median tutkia MIDP:hen
- Alusta riippumaton ohut Java kerros
- Eri asia kuin JMF
- Valinnainen paketti
- MIDP 2.0 sisältää vain audio-osuuden MMAPI:sta

Petri Vuorimaa

34

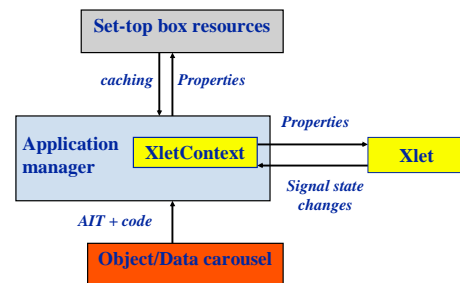
Multimedia Home Platform

- Yksi esimerkki Java-ympäristöstä on digitaalisen television Multimedia Home Platform (MHP)
- Ympäristö koostuu joukosta Java, JMF, TV, verkko, ym. ohjelmointirajapintoja (API)
- Sovelluksia kutsutaan nimellä Xlet
- Xlettejä lähetetään lähetyksen kautta
- Data välitetään ns. data- ja objektiokarusellin kautta
- Application Manager ohjaa sovelluksia

Petri Vuorimaa

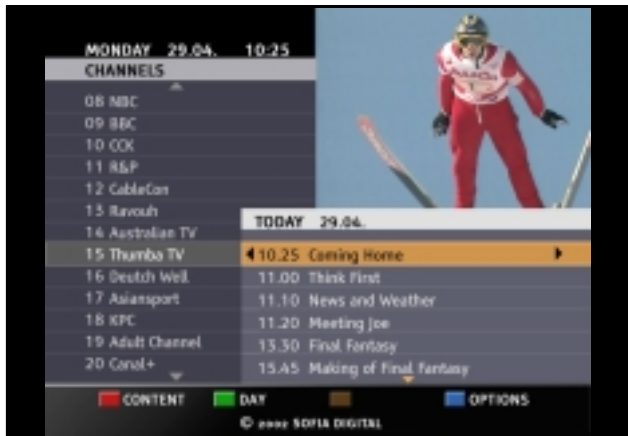
35

Application Manager



Petri Vuorimaa

36



XML

- Merkitäkielten määrittelyyn tarkoitettu metakieli
+ XML-kieltä voidaan käyttää joko sisällön prosessointiin tai esittämiseen
- Prosessoinnilla tarkoitetaan esim. XML-muotoisen sisällön hakua tai suodatusta
- Esittämisellä tarkoitetaan XML-muotoisen sisällön esittämistä päätelaitteessa
- Usein käytetään myös Cascading Style Sheets (CSS) -tyylisivuja ja ECMAScript-kieltä

Petri Vuorimaa

39

Tärkeimmät prosessointikieliet

- eXtensible Stylesheet Language Transformations (XSLT) – XML datan transformaatiot
- XML Schema - Datamalli
- XPath – Elementteihin viittaus

Petri Vuorimaa

40

Tärkeimmät esityskielet

- XSL Formatting Objects (XSL FO) – Formatointi
- eXtensible HyperText Markup Language (XHTML) – HTML 4.01 XML muodossa
- Synchronized Multimedia Integration Language (SMIL) – Synkronoitu multimediakieli
- Scalable Vector Graphics (SVG) - Vektorigrafiikka
- X3D – 3D grafiikka
- VoiceXML – Puheohjaus

Petri Vuorimaa

41

XML-datan käsittely

- XML-tiedoston parsiminen
- XSL-tyylisivun parsiminen
- XML-datan muunnos XSL-tyylisivun mukaisesti
- Document Object Model (DOM) -tietorakenne
- Käynnistysaikaisten skriptien suorittaminen
- Esityskielen asemointimallin rakentaminen
- Esityskielen renderöinti
- Tapahtumankäsittely-skriptien suoritus

Petri Vuorimaa

42

Tärkeimmät XML-komponentit

- XML-parseri
- XSL-muunnokset
- DOM-rajapinta
- ECMAScript-tulkki
- XPath-prosessori
- XML Schema -prosessori
- CSS asemointi

Petri Vuorimaa

43

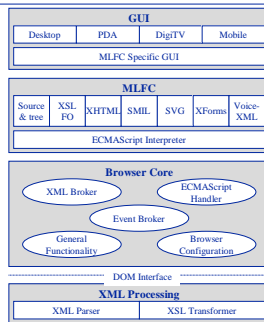
X-Smiles historia

- X-Smilesin kehitys aloitettiin opiskelijoiden ohjelmistoprojektina 1998
- Myöhemmin kehitystyötä on jatkettu tutkimushankkeissa
 - + GO
 - + XML Devices
- Julkaistu avoimena ohjelmakoodina 2001
 - + www.x-smiles.org

Petri Vuorimaa

44

X-Smiles arkkitehtuuri



Petri Vuorimaa

45

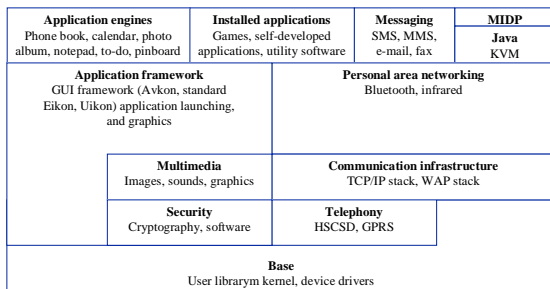
Käyttöjärjestelmät

- Nykyisissä ohjelmistoarkkitehtuureissa käyttöjärjestelmällä on tärkeä rooli
- Suuri osa median käsittelystä tehdään käyttöjärjestelmäkutsuilla
- Sovellus toimii koordinaattorilla
- Datan kopiointi sovelluksen ja käyttöjärjestelmän välillä on ongelma
- Käyttöjärjestelmä hallitsee resursseja huonosti

Petri Vuorimaa

46

Symbian OS



Petri Vuorimaa

47

Reaaliaikaiset käyttöjärjestelmät

- **Pieni koko:** ylimääräiset ominaisuudet on karsittu
- **Keskeytysten käsittely:** ruuhkautuminen estetty
- **Reaaliaikainen skedulointi:** suorituksen ajoitus
- **Tehokas muistinhallinta:** jaettu muistialue
- **Nopea viestinvälitys:** eri tasojen välinen kommunikointi on optimoitu
- **Resurssien varaus:** tehtävää ei oteta suoritettavaksi, jos sitä ei pystytä tekemään

Petri Vuorimaa

48

Microkernel

- Käyttöjärjestelmästä saadaan kompaktimpi karsimalla siitä kaikki "turhat" ominaisuudet
 - + ikkunointi, hajautetut tiedostojärjestelmät ym.
- Yksi ratkaisu on ns. Microkernel
 - + käyttöjärjestelmän ydin (kernel) on tehty mahdollisimman pieneksi ja reaaliaikaiseksi
 - + muut ominaisuudet toteutetaan käyttäjäprosesseina
 - + haluttu kokoonpano voidaan kasata eri moduleista

Sulautettu Linux

- Linux on kiinnostava vaihtoehto myös sulautetuissa päätelaitteissa
- Ylimääräiset ominaisuudet (esim. X-ikkunointi voidaan karsia pois) ja koko saadaan pieneksi
- Reaaliaikaisuuden toteuttaminen ytimeen (kernel) on erittäin vaikeaa, mutta ei aina välttämätöntä
- Esim. digiboksissa laitteisto vastaa videon ja audion käsittelyssä - vuorovaikutteisuus hoidetaan softalla