

# HOOFDSTUK 1: Inleiding

## §1. Wat is multimedia?

### *definitie*

veelzijdig gebruikt + contextafhankelijk

computerwetenschappen:

“computertoepassingen waarin verschillende ‘media’ gebruikt worden

media = spraak, geluid, stilstaand beeld, bewegend beeld, animatie, tekst, ...

+ **interactiviteit**

### *onderzoeksgebieden*

multimediaverwerking en –codering

encoderen en decoderen van audio/beeld/video

compressie van audio/beeld/video

analyse van audio/beeld/video

multimediasystemen en –netwerken

netwerk protocols, internet, ... als én met multimedia

afspeelsystemen (hardware/OS/specs)

client/server en streamen

databanken en multimedia

multimediasystemen en –software

vb. user interface, ...

### *productie*

diverse jobs en functies

technische aspecten

vb. compressie, interactie, implementatie, ...

artistieke aspecten

vb. artiest, GUI designer, schrijver, concept art, ...

### *lineaire en niet-lineaire multimedia*

lineair: geen navigatie, interactiviteit (vaak wel, maar verborgen en niet standaard)

vb. presentatie, film, ...

niet-lineair: aan de hand van interactie met de gebruiker

vb. video games, hypermedia, ...

## *presentatie*

stijlen zeer belangrijk

bepaald door onderzoek naar menselijk zicht  
veel onderzoek naar gedaan (vb. marketing)

kleur-principes

niet te veel kleuren

niet als versiering

consistent gebruik van kleuren

contrast

vb. geen roze tekst op rode achtergrond

geen donkere achtergrond met lichte letters, maar omgekeerd

inverse kleuren (voor contrast)

$K = (R, G, B)$

$K' = (255-R, 255-G, 255-B)$

lettertype

groot genoeg lettertype + weinig tekst tegelijkertijd

zeker bij vb. presentatie

schreefhebbend vs. schreefloos

schreefhebbend: lange tekst, ...

schreefloos: presentatie, website, ...

## §2. Geschiedenis van multimedia

1945 – Artikel ‘*Memex*’

door Vannevar Bush  
al enkele basisconcepten van hypertext  
rechtstreekse inspiratiebron voor latere ontwikkeling

1960 – Introductie term ‘*hypertext*’

door Ted Nelson

1967 – Oprichting *Architecture Machine Group* aan MIT

door Nicholas Negroponte

1968 – Voorstelling *On-line System (NLS)*

door Douglas Engelbart  
primitieve hypertext-applicatie

1969 – *FRESS*

door Ted Nelson en Andries van Dam (Brown Universiteit)  
primitief hypertext systeem en –editor

1985 – Oprichting *MIT Media Lab*

door Nicholas Negroponte en Jerome Wiesner  
opvolger van de Architecture Machine Group

1989 – Voorstelling *World Wide Web*

door Tim Berners-Lee en Robert Cailliau (CERN, Geneve)

1990 – Successen *Apple Multimedia Lab*

door Kristina Hooper Woolsey  
van Apple en later terug gestopt

1991 – Aanvaarding MPEG-1 als digitaal beeld standaard

voorloper van MPEG-2, MPEG-4,... in jaren ‘90

1991 – Opkomst PDA’s

wijzigend computergebruik t.o.v. MM (mobiel, klein scherm)  
geschiedenis:

Palm vs. Compaq

Compaq alliantie met Microsoft => Palm weg

Compaq overgenomen door HP

1992 – Aanvaarding JPEG als standaard voor digitale beeldcompressie

voorloper van JPEG2000

1993 – Productie *NCSA Mosaic*

door National Center for Supercomputing Applications (University of Illinois)  
eerste volledige browser  
MM in al zijn vormen over hele wereld sturen!

1994 – Productie *Netscape*

door Jim Clarck en Marc Andreessen  
geschiedenis:

concurrentie met IE

proces tegen Microsoft <-> laten rekken en iedereen overgestapt

neergang

Mozilla project (Open Source)

1996 – Introductie DVD Video

1998 – XML 1.0 als W3C Recommendation

1998 – Opkomst MP3-spelers

## §3. Standaardisatie

### *W3C-standaarden*

standaard: open en gratis beschikbaar

<-> gebruik: NIET altijd gratis (wegens patenten)  
verantwoordelijkheid bij gebruiker (disclaimer)

procedure

1. standaardisatiecomité geeft requirements uit voor bepaalde toekomstige standaard
2. markt reageert + dienen elk hun 'oplossing' in markt = bedrijven, universiteiten, onderzoeksgroepen competitieve fase
3. standaardisatiecomité vergelijkt en test ze a.d.h.v. *core-experiments* vaak vele verschillende beter op andere deelaspecten
4. standaardisatiecomité kiest die beste delen + brengt partners samen (proberen)
5. partners werken samen om tot geheel te komen  
<-> nog steeds veel (eigen)belangen en discussies  
coöperatieve face
6. testen + publiceren van standaard  
<-> onderliggend vele patenten

### *MPEG (ISO)-standaarden*

MPEG verplicht patenten van meewerkende entiteiten

standaard: open en tegen (kleine) betaling

<-> gebruik: NIET altijd gratis (wegens patenten)

procedure

5. <analoog W3C>
6. testen + klaar voor publicatie
7. standaardisatiecomité vraagt naar patenten  
verplicht op ALLE geïntegreerde technologieën  
patent = teken van engagement + geloof in eigen technologie  
want toen shitload aan nutteloze + ongebruikte standaarden
8. publicatie

## *video-standaarden*

### huidige standaarden

analoge TV: NTSC (VS) en PAL (EU) en SECAM (Fr.) + niet compatibel!  
digitale TV: MPEG-2  
DVD: MPEG-2  
Blu-ray: MPEG-2 of H.264/MPEG-4 ACV

### geschiedenis

MPEG-2 = enorm succes  
standaardiseren van decoder  
= vanuit consument standpunt (vb. mp3 speler)  
<-> encoder niet gestandaardiseerd!!  
=> concurrentie blijft mogelijk  
testen v/e device = bitstream door decoders laten lopen

### MPEG-4 AVC

door MPEG (ISO) en ITU (VN)  
opvolger MPEG-2 en backwards compatibel  
=> geen onmiddellijke overstap voor content providers nodig  
tot 40% bitreductie mogelijk  
maar niet altijd, want afhankelijk van encoder!

## *bemerkingen*

### open vs. gratis

standaarden vaak open = vrij te bekijken (evt. tegen vergoeding)  
toch gepatenteerd => voor gebruik misschien betalen!

### gepatenteerd vs. te betalen

vaak onderliggende patenten als verdedigingsmechanisme  
want alle rechten bij maker (vb. latere versies)  
maar toch gratis voor normale gebruikers

## §4. Metadata

### *wat*

= 'data over de (multimedia)data'

vaak XML

vb. maker, regisseur, componist, keywords, rating, GPS locatie, ...

### *nut*

zeer belangrijk voor latere archivering/opslag

geldt als fiche voor MM-bestand

gebruikt voor opzoekingen in MM-databanken

als textuele data/info 'op' audio, video en beeld

makkelijker te lezen en te interpreteren

<-> afleiden uit groot, lang en zwaar bestand

### *aanmaken*

bij creatie van het MM-project!!

#### reden 1:

later moeilijker te achterhalen

dan uit bestand halen of door onderzoek

= arbeidsintensief + kostelijk

#### reden 2:

later hele archieven tegelijk voorzien

dan zeer dure operatie, dus zelden rendabel

## §5. XML technologieën

### *XML*

= 'Extensible Markup Language'

standaard voor definiëren van formele mark-up talen  
representatie van gestructureerde gegevens  
platte tekst

meta-taal

uitbreidbaar

zelf 'standaarden' maken, afspreken, en vastleggen via DTD en XSD

vereenvoudiging zeer complexe SGML  
specifiek voor WWW

te herkennen door computers (parsen) én door mens (leesbaar)

strikte scheiding inhoud <-> opmaak

geen opmaak in XML bestand

opmaak door externe stylesheet (XSL, CSS)

opmaak door transformeren naar XHTML, PDF, WordML, ... (XSLT)

### *SMIL*

'Synchronized Multimedia Integration Language'

doel: MM-presentaties definiëren in XML

verschillende vorm

temporeel: wanneer wat tonen?

spatiaal: waar wat tonen?

nog weinig ondersteund

niet door vb. browsers

wel al door media players



## §6. VRML

### VRML

= Virtual Reality Modelling Language (“Virmel”)

doel: objecten beschrijven in 3D wereld (VRML world)  
bevat enkele elementaire bouwstenen om mee te bouwen  
laat deze objecten (met elkaar) interageren

standaard

vrij beschikbaar

<-> gepatenteerd

beschrijft elementaire blokjes + hoe ermee te bouwen + hoe ermee te interageren

succes vanaf MPEG-4

daar objecten apart uit film halen + apart coderen

nood aan “scenebeschrijving”

= hoe staan objecten t.o.v. elkaar

beetje gebaseerd op SMIL

niet enkel wat zit waar en wanneer

ook samenstelling door bouwmogelijkheden

### VRML-shapes

soort klassestructuur van type nodes (= hiërarchie)

*shape*-node

generische node van alle objecten in VRML

vb. Box, Sphere, zijn geometry nodes

*material*-node

bevat eigenschappen van oppervlakte v/e object

kleur v/h object

transparantie v/h object

*texture*-node!

*texture*-node

doel: textures “mappen” op een 3D-object

ImageTexture / MovieTexture

PixelTexture (ook soort figuur, maar pixel per pixel doorsturen)

## *VRML-world*

bevat algemene opties/eigenschappen

camera/point of view

instelbaar

positie, rotatie, hoek

belichting

3 soorten

DirectionalLight

bepaalde richting (evenwijdige lichtstralen)

gehele scene belicht

PointLight

lichtbron 'in' de scene (klein object)

in alle richtingen (diffuus)

SpotLight

lichtbron 'in' de scene (klein object)

in bepaalde richting + breedte

achtergrond

instelbaar

panorama

texture plakken op randen v/d wereld

grote kubus rond de VRML-wereld

## *interactie en animatie*

a.d.h.v. geïnterpreteerde taal bij VRML

=> zeer traag

=> nooit gebruikt

vaak apart via externe toepassing/taal implementeren