



# Fachprüfung: Hyper- und Multimedia

## Hypermedia- und Multimediasysteme (1,2)

31.01.2002

<b>Name</b> (bitte in Blockschrift)	
<b>Matrikelnummer</b>	
<b>Unterschrift</b>	

**Hinweise:**

- Überprüfen Sie Ihr Klausur-Exemplar bitte vor Beginn der Klausur auf Vollständigkeit.
- Bitte halten Sie Ihren Lichtbildausweis sowie den Studentenausweis zur Kontrolle bereit.
- Dauer der Klausur: 120 min.
- maximal erreichbare Punktzahl: 200
- Bitte füllen Sie das Deckblatt vollständig aus, beschriften jedes Blatt mit Ihrer Matrikelnummer und unterschreiben Sie dieses Klausur-Exemplar.
- Jedes Verlassen des Prüfungsraums muss ausdrücklich mit der Aufsicht vereinbart werden.
- Zugelassene Hilfsmittel sind ausschließlich Schreibutensilien, nicht-programmierbare Taschenrechner und das eigene(!) Gedächtnis.
- Bitte vermeiden Sie die Verwendung von roter Farbe.
- Die nach jeder Frage eingeklammerte Zahl ist die bei dieser Frage maximal erreichbare Punktzahl.
- Beachten Sie die in vielen Fragen enthaltenen Teilfragen!
- Falls der Platz für die Beantwortung einer Frage nicht ausreichen sollte, verwenden Sie bitte die Rückseite.
- Nutzen Sie im Falle von Unklarheiten hinsichtlich der Fragestellung die Möglichkeit zu Rückfragen!

**Viel Erfolg!**

Punkte Teil 1	
Punkte Teil 2	
Punkte gesamt	

1. Prüfer .....	..... Note
2. Prüfer .....	



**- I. Prüfungsfragen zur Veranstaltung im SS 2000 (HMS-1) -**

1. Mit welchen Zielsetzungen wurden die unterschiedlichen MPEG-Standards entwickelt? Mit welchen neuen Konzepten wurde dies jeweils technisch umgesetzt? (16)

- MPEG-1:  
Ziel: Speicherung und Wiedergabe von bewegten Bildern und Audio-Speichermedien (CD-I, DAB, ISDN=64 kbps) (++)  
Konzept: optimierte Kompression von Audio und Video (++)
- MPEG-2:  
Ziel: digitales Fernsehen (DVD, DVB) (++)  
Konzept: verschiedene Qualitäts- und Kompressionsstufen, Interlace (++)
- MPEG-4:  
Ziel: Multimedia-Anwendungen (++)  
Konzept: Interaktivität, unabhängige Medienobjekte, Kombination realer und synthetischer Objekte (++)
- MPEG-7:  
Ziel: Inhalts-basierte Informations-Suche, Archiv-Funktionalität (++)  
Konzept: Meta-Daten zur Inhaltsbeschreibung (++)

2. Wodurch erreicht MP3 einen besonders hohen Kompressionsgrad von Audio-Dateien? (3)

- verlustbehaftete Verfahren: physiologische Optimierung: (+)
  - z.B. Verzicht auf unhörbare Signale (Hörschwelle, Maskierung) (++)



3. Erläutern Sie das Konzept von *Profiles* und *Levels* in MPEG-2. Wo findet sich HDTV in diesem Konzept wieder? Was bedeutet SP@ML? (12)

- MPEG-2 beinhaltet verschiedene AV-Protokolle unterschiedlicher Qualität, die sich durch Kombination je eines Profiles und eines Levels definieren. (++)
- Level gibt die Auflösung in Bildpunkten (+) und Frames/s (+) an.
- Profile gibt die Bandbreite bzw. Kompressionsrate an, sowie Farbsampling (4:2:2, etc.) und B-Frame-Rate. (++)
- HDTV: High Level (1920\*1152\*60) (++)
- SP@ML = Simple Profile (<15 Mbps) (++) bei Main Level (720\*576\*30) (++)

4. Wie arbeitet der CELP-Encoder in MPEG-4? Für welche Art von Audiosignalen wird er verwendet? (7)

- Auswahl desjenigen Codeworts mit minimalem Prädiktionsfehler (++):
  - Aus einer Sammlung von Codeworten wird ein zum Eingangssignal passendes ausgesucht. (+)
  - Nach Verstärkung (Gain) des Codeworts durchläuft das Signal Kurzzeit- und Langzeit-Prädiktor. (+)
  - Nach Durchlaufen eines perzeptuellen Gewichtungsfilters wird ein gewichteter Prädiktionsfehler festgestellt. (+)
- CELP wird verwendet für Sprache und Breitbandsprache / Schmalbandaudio. (++)



5. Beschreiben Sie das 3-Schichten-Modell für MPEG-4 Audio. (12)

- Demux/Decoder-Schicht → AudioBIFS-Schicht → BIFS-Schicht (+++)
- Demux/Decoder-Schicht:
  - Decoder (Para-Decoder, CELP-Decoder, T/F-Decoder) nehmen Multiplex-Ströme von Servern auf. (+++)
- AudioBIFS-Schicht:
  - Audio-Subgraph (Szene Graph):  
Audioquellen werden aufgeteilt (Switch), gemischt (Audio Mix) und/oder mit Effekten überlagert (Audio FX) (+++)
- BIFS-Schicht:
  - logische Verschaltung (Kombination) der Audio-Elemente mit weiteren BIFS-Elementen (z.B. Video) (+++)

6. Unterscheiden Sie grundsätzlich Push- und Pull-Applikationen für MPEG-7. Nennen und beschreiben Sie je zwei Beispiele? (12)

- Push-Applikationen: Ohne Anwender-Initiative werden Informationen vom Anbieter (aus dem Medienarchiv) individuell angepasst verschickt. (++)  
Bsp.:
  - Medienauswahl (Filterung) durch User-Agenten (++)
  - personalisierte TV-Angebote (++)
- Pull-Applikationen: Anwender beschaffen sich aktiv durch gezielte Suche Informationen beim Anbieter. (++)  
Bsp.:
  - Sound-Effekte für speziellen Bedarf (++)
  - Film- oder Nachrichtenszenen auf Beschreibung oder anhand bestimmter Textpassagen (++)



7. Beschreiben Sie das Prinzip des Multimedi-Streaming. Worin bestehen Vor- und Nachteile im Vergleich zu herkömmlichen HTML-Dokumenten? (9)

- Streaming: Verschicken multimedialer Daten in Form von Datenströmen, die permanent ausgelesen und sofort angezeigt werden. (+++)
- Vorteile: keine Wartezeiten, Live-Sendung möglich, kein großer Zwischenspeicher erforderlich (+++)
- Nachteile: keine Qualitätsgarantie; Bandbreiten-abhängig; Spezialsoftware (Encoder/Decoder) auf Server- und auf Clientseite erforderlich. (+++)

8. Worin besteht das grundsätzliche Prinzip der Huffman-Kodierung? Auf welche Weise (und unter welchen Voraussetzungen) trägt diese zur Datenreduktion bei der JPEG-Bildkompression bei? Wie sähe eine solche Codierung konkret für die Zahlenfolge [ 0 2 2 2 2 5 1 1 ] aus? (14)

- Huffman-Kodierung: redundanzfreie präfixfreie Codes variabler Wortlänge in Abhängigkeit von Auftrenenshäufigkeit. (++++)
- Vorgehensweise: Wahl der 2 Werte geringster Wahrscheinlichkeit und Zuordnung je eines Wertes 0/1; Zusammenfassung dieser beiden Werte zu neuem Hilfssymbol und analoges Vorgehen mit nächstwahrscheinlichem Wert, bis alle Werte erfasst; dabei erhält der häufigste Wert nur ein einstelliges Symbol.
- Datenreduktion durch kurze Wortlängen für häufig auftretende Werte; (+++) besonders effizient bei hoher Wiederholungszahl einzelner Werte. (+)
- [ 0 2 2 2 2 5 1 1 ] → [ 11000001111010 ] (++)  
2 → 0 ; 1 → 10 ; 0 → 110 ; 5 → 111 (++++)



9. Welche Randbedingungen bzw. Sonderfälle (nennen Sie mindestens drei) müssen bei der JPEG-ähnlichen Bildkompressions-Implementierung in einem (C/C++ od. Java- od. Perl- etc.) Programm besonders berücksichtigt werden (unabhängig von der jeweiligen Programmiersprache)? Machen Sie jeweils Lösungsvorschläge. (15)

- Rundungsfehler (++) → Verwendung von double precision bei Zwischenrechnungen od. int zu float zu int bei RGB-Umwandlung in YUV (+++)
- Bildränder passen nicht zur 8x8-Matrix (DCT u.a.) (++) → Verwendung kleinerer Matrizen oder Auffüllen mit Nullwerten oder Anschluss von Bildkopien an den Rändern (+++)
- Chrominanz-Reduktion: Wahl der kombinierten Werte (++) → Mittelwert oder Auswahl eines der zusammen gezogenen Werte (+++)
- ...

Punkte Teil 1

/ 100

## - II. Prüfungsfragen zur Veranstaltung im SS 2001 (HMS-2) -

1. Was sind „DataIslands“, wie und in welchem Kontext werden sie verwendet? Wie kann auf „DataIslands“ zugegriffen werden? (15)

Mit Hilfe von DataIslands können XML-Daten direkt innerhalb einer HTML-Seite enthalten sein. Der Vorteil ist dabei, dass die Daten dem Client zur ständigen Verfügung stehen, sobald die betreffende Seite im Browser geladen ist. Eine mögliche Anwendung ist zum Beispiel das Verwalten von Tabelleninhalten, die auf Anforderung hin umsortiert werden sollen ohne dass die Seite oder die Informationen in der Tabelle erneut vom Server geladen werden sollen.

Eingebunden werden können DataIslands u.a. wie folgt:

```
<XML ID="XMLID">  
  <XMLDATA>  
    <DATA>TEXT</DATA>  
  </XMLDATA>  
</XML>
```

oder

```
<XML SRC="http://localhost/xmlFile.xml"></XML>
```

oder

```
<SCRIPT LANGUAGE="XML" SRC="http://localhost/xmlFile.xml"></SCRIPT>
```

oder

```
<SCRIPT ID="XMLID" LANGUAGE="XML">  
  <XMLDATA>  
    <DATA>TEXT</DATA>  
  </XMLDATA>  
</SCRIPT>
```

Der Zugriff geschieht dann ganz normal mit JavaScript, z.B.:

```
function returnXMLData()  
{ return document.all("XMLID").XMLDocument.nodeValue; }  
  
function returnXMLData()  
{ return XMLID.documentElement.text; }
```



2. Wie lassen sich innerhalb von JavaScript eigene Objekte erstellen? Erstellen Sie ein Beispiel. (10)

Beispiel:

```
function Farbe(Farbwert_R, Farbwert_G, Farbwert_B)
{
  this.Farbwert_R = Farbwert_R;
  this.Farbwert_G = Farbwert_G;
  this.Farbwert_B = Farbwert_B;
}

function TestObjekt()
{
  Test = new Farbe("33","99","C0");
  alert("Der Rotwert meiner Farbe ist Hexadezimal " +
  Test.Farbwert_R);
}
```

3. Wie lassen sich innerhalb von JavaScript globale Variablen deklarieren? Wo sind deren Grenzen? (10)

Globale Variablen werden außerhalb von Funktionen deklariert. Sie gelten nur innerhalb der Seite, in der sie in einem Scriptabschnitt deklariert sind, d.h. z.B. für benachbarte Frameseiten sind sie nicht unmittelbar gültig.





4. Was verstehen Sie im Kontext der Webtechnologie unter „DOM“? Welche Bestandteile gehören zum DOM? (5)

DOM ist die Abkürzung für Dokumenten-Objekt-Modell, das eine HTML-Seite als in Objekthierarchien angeordnete Elemente versteht. Es umfasst alle herkömmlichen Tags sowie Mengen und Aufzählungstypen, sogenannte Collections.

5. Wie lassen sich im MS Internet Explorer HTML-Elemente zoomen? Nennen Sie auch mögliche Maßeinheiten. Welche Bestandteile einer Seite können gezoomt werden? (10)

Über die CSS-Eigenschaft "style.zoom=...". Der Wert kann in Prozent oder als Fließkommazahl angegeben werden (100% entsprechen dabei 1.0). Grundsätzlich kann nicht nur die gesamte Seite, sondern auch einzelnen Elemente gezoomt werden (z.B. Bilder).

6. Was sind sog. „Collections“ im Unterschied zu normalen Elementen? Wie greifen Sie auf Bestandteile einer Collection zu? Nennen Sie 5 Beispiele für Collections in DHTML? (10)

Collections beschreiben eine Ansammlung gleichartiger Elemente einer HTML Seite wie z.B. alle Bilder oder alle Hyperlinks. Beispiele hierfür sind all; anchors; applets; areas; cells; children; elements; embeds; filters; forms; frames; images; imports; links; options; plugins; rows; rules; scripts; styleSheets; tbodies. Die Anzahl der einzelnen Elemente einer Collection kann mit collection.length abgefragt werden. Auf ein einzelnes Element kann mit einem Index zugegriffen werden.



7. Nennen Sie typische Bestandteile von DHTML und deren Funktion. (10)

Das Dokumentenobjektmodell (DOM) beschreibt den hierarchischen Aufbau der Elemente einer HTML-Seite, auf die (und deren Eigenschaften und Methoden) mit Hilfe von Scriptsprachen (dynamic content) wie JavaScript und Cascading Style Sheets (CSS) (dynamic styles) zur Laufzeit lesend und schreibend zugegriffen werden kann. Events schließlich dienen der Möglichkeit, auf Benutzeraktionen (button click, scrolling, ...) mit Hilfe des Scriptings reagieren zu können.

8. Was ist der Unterschied zwischen den Properties „innerHTML“ und „innerText“ im Internet Explorer? (5)

„innerText“ umfasst nur den Text eines Elementes, „innerHTML“ zusätzlich die Tags innerhalb eines Elementes inklusive ggf. definierter Attribute. Beide Eigenschaften sind lesend und schreibend.

9. Auf welche (4) verschiedenen Arten können Cascading Stylesheets (CSS) in HTML Seiten verwendet werden? Lassen sich die unterschiedlichen Arbeitsweisen in Bezug auf ein HTML-Element kombinieren? Wie sehen dabei die Prioritäten aus? (10)

Inline über das Attribut Style auf Ebene des einzelnen Elementes, Embedded im Style-Tag im Headbereich, External über das Link-Tag als Verweis auf eine separate CSS-Datei oder per Scripting. Die genannten Formen lassen sich mischen bzw. kombinieren. Dabei gilt folgende Priorität in absteigender Reihenfolge bei sich widersprechenden Anweisungen: Scripting > Inline > Embedded > External.



10. Wann setzen Sie Elemente und Attribute in XML ein? (5)

Metainformationen werden gewöhnlich als Attribute, Inhalte als Elemente eingesetzt. Grundsätzlich sind jedoch beide Varianten mit kleinen Einschränkungen möglich (Textlänge und Sonderzeichenbehandlung bei Attributen...).

11. Wie können Sie im Internet Explorer 5.x eigene Tags definieren? (10)

Über die Definition eines Namespace „xmlns“ im HTML-Tag und der Deklaration entsprechender Styles für die innerhalb des Namespace vorkommenden Tags

```
<HTML xmlns:FH="urn:FH">
  <HEAD>
    <STYLE>
      @media all{FH\;SEMESTER{COLOR:RED;}}
    </STYLE>
  </HEAD>
  <BODY>
    <FH:SEMESTER>6. Semester</FH:SEMESTER>
  </BODY>
</HTML>
```

Punkte Teil 2

/ 100

**Viel Erfolg!**