

ViSiCAST Deliverable D1-2: Advanced Sign Transmission Demonstrator

Project Number:	IST-1999-10500
Project Title:	ViSiCAST
	Virtual Signing: Capture, Animation, Storage and Transmission
Deliverable Type:	Report

Deliverable Number:	D1-2
Contractual Date of Delivery:	Sept. 2002
Actual Date of Delivery:	Jan. 2003
Title of Deliverable:	Advanced Sign Transmission Demonstrator
Work-Package contributing to the Deliverable:	Workpackage Broadcast Transmission (WP1)
Nature of the Deliverable:	public
Author(s):	Brückner

Abstract:

The deliverable gives a focus on how to transport different formats of ViSiCAST data via the broadcast channels. Furthermore the deliverable will describe different implementations of the Multimedia Home Platform MHP and the implications for ViSiCAST. Some results gained in subjective tests are given in a thesis related to workpackage 1. At the end of the deliverable some impressions of the booths of IRT and EBU demonstrating ViSiCAST applications at IBC-2002 are given.

Key words: MPEG-2, MPEG-4, MHP

Table of Contents

1. INTRODUCTION

2. VISICAST AND MPEG-2

3. VISICAST AND MPEG-4

3.1 Major functionalities in MPEG-4 visual specification

3.2 Sign language in MPEG-4 video elementary streams and MPEG-4 file format

3.3 MPEG-4 A/V Encoder

4. VISICAST AND THE MULTIMEDIA HOME PLATFORM MHP

4.1 Introduction

4.2 MHP compliant STB

4.3 Implications for ViSiCAST

4.4 The SNHC (Synthetic/Natural Hybrid Coding) player using Face & Body Animation

5. SYNCHRONISATION AND COMPRESSION

APPENDICES

Appendix A: Results of a thesis in WP1

Appendix B: Visicast at IBC

LIST OF ABBREVIATIONS

REFERENCES

1 Introduction

The begin of digital TV will produce a lot of new, innovative services. Consequently there will be a great advance in the capabilities of additional services. It is envisaged that new business models and opportunities will be created as a result of these multimedia broadcast services. To be accepted by the user, these new integrated broadcast services need to be designed so that they can be easily managed and readily and comfortably accessed.

Therefore the main objectives of the System for Virtual Signing, Capture, Animation, Storage and Transmission (ViSiCAST) project in the work package one were:

- To develop Technology for real-time design of new services containing MPEG-2 and MPEG-4 elements for closed and open signing.
- To develop Multimedia Terminal Technology for access to these new services. This is done through enhanced reception at the home. New techniques are required to successfully compose different types of content.
- To provide Europe with a lead in technology for creating these services and receiver terminal technology.

The ViSiCAST system will allow to produce and display innovative integrated broadcast content for recorded and on-line programs using MPEG-2 and MPEG-4 multimedia technology as a basis. The combination with other types of multimedia content (e.g. HTML, XML) was also foreseen.

ISO MPEG and DVB technology has been chosen as the cornerstone technology for designing the ViSiCAST broadcast system. It provides an integrated suite of tools for coding, indexing and transmission of natural and synthetic signers.

In order to show the performance of the developed systems, examples of broadcast services have been developed. The services demonstrated new opportunities for the disabled:

- The services are able to access information from the broadcast transport stream
- The services are able to work in a so called open and/or closed signing mode which means the viewer can decide to display or not to display the new service on its TV screen
- The services consists of MPEG-2 and MPEG-4 elements that can be received from broadcast channels
- The services will be displayed on a common terminal platforms like TV sets or PCs

2 ViSiCAST and MPEG-2

Figure 1 shows the basic structure of a ViSiCAST data broadcast model within the DVB Multiplex streams [1]. The underlying MPEG transport stream (TS) either carries the Paketized Elementary Stream (PES) and/or the information will be carried in sections. Beside the MPEG2 Video/Audio the ViSiCAST content will be transported in MPEG-4 format or, alternatively, in a proprietary format. ViSiCAST have chosen the so called mask-vr as a proprietary format [12].

Mask-vr is the pre-existing motion capture, recording and playback software developed by Televirtual Ltd. The motion capture or recording routines of mask-vr trap and process data at a rate of down to 50 kb/sec when compression algorithm are used. In a later stage of the project the mask-vr format was renamed to BAF (Body Animation File).

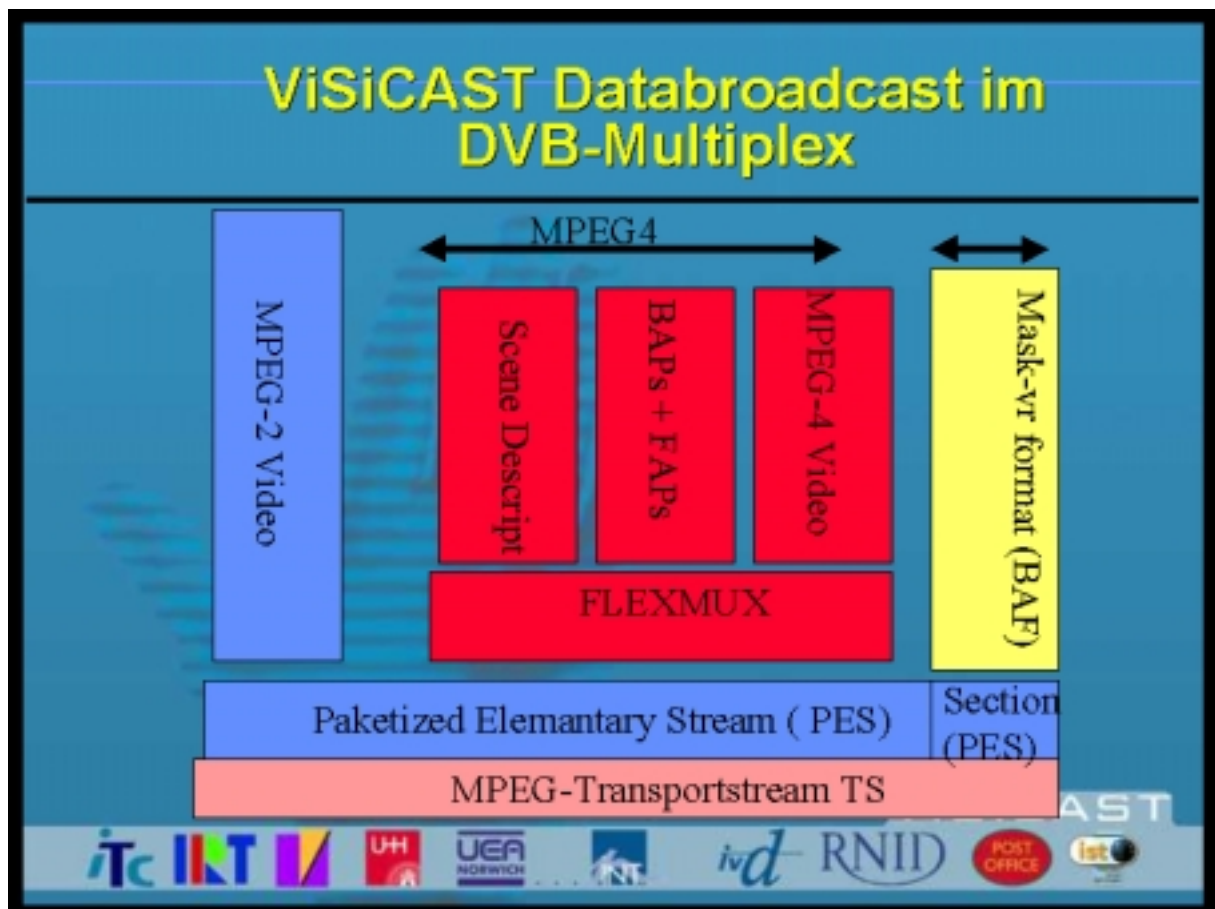


Figure 1: Basic structure of a MPEG multiplex

In the ViSiCAST project the following mechanisms of the MPEG-2 transport stream syntax are utilised:

- **PES packets:** Containing both MPEG-4 and MPEG-2 visual material encapsulated as described in MPEG-2 systems [2] and amendment 7 to MPEG-2 systems [5].
- **Sections:** Containing PSI and SI tables (one section syntax defined for each type of table). The tables contain descriptors as defined in several specifications from both MPEG (MPEG-2 systems [2] and amendment 7 to MPEG-2 systems [5]) and DVB (DVB SI [8], DVB data broadcasting [6] and DVB MHP [7]).
- **DSM_CC sections:** Containing the data in the object carousel as described in [17].

More details on the usage of the different mechanisms and their use are given in section 5 under "Synchronisation".

For the purpose of over-all synchronisation it was necessary to have access to the MPEG-2 Transport Stream for recovering the Object_Clock_Reference. Only with this information an over-all synchronisation could be realised.

Due to the fact, that the actually available DVB-board from Fujitsu-Siemens delivered not the TS but only MPEG-2 video and audio (PES will be accessible in an updated version) as well as private sections, only the particular synchronisation layers of the different engines are available. Therefore at the moment it is only possible that each engine must achieve synchronisation by itself.

3 ViSiCAST and MPEG-4

3.1 Major functionalities in MPEG-4 visual specification

The visual part of MPEG-4 (ISO/IEC 14496 part 2) [3] comprises algorithms supporting coding of natural still images and video sequences as well as tools supporting compression of synthetic 2D and 3D graphic geometry parameters. The functionality provided is summarised below.

Formats and compression: When encoding natural video material using MPEG-4, several combinations of bit rates (5 kbit/s – 10 Mbit/s), formats (interlace and progressive scan) and resolutions (from sub-QCIF to higher than standard television) can be used. Efficient compression is provided at all bit rates addressed, and the quality is adjustable between “acceptable” at very high compression ratios up to “nearly loss-less” using little compression.

Improved coding efficiency: MPEG-4 uses the same basic technology as MPEG-1 and MPEG-2 for encoding of video, that is a hybrid DCT (Discrete Cosine Transform) coding scheme using motion estimation to exploit temporal redundancy and DCT transformation to exploit spatial redundancy. Through development of completely new tools and fine-tuning of others, the coding efficiency is improved, particularly at lower bit rates. The coding scheme is designed to have nearly immediate access to streaming video.

Objects and coding of shape: MPEG-4 allows encoding of objects with arbitrary shape. This means that information about the height and width of the picture is not sufficient anymore. In addition to the luminance and chrominance components, also information about the shape

must be transmitted (alpha channel). MPEG-4 defines efficient techniques for encoding of two types of shape information:

- *Binary shape coding*: A bit map defines whether a pixel belongs to an object or not (on/off).
- *Grey-scale shape coding*: The grey scale map defines the transparency of each pixel. Multi level maps can be used to blend different layers of image sequences.

The advantage of object-based coding is twofold: The pixels outside the area of interest need not to be encoded, and object-based manipulation is facilitated. The drawback is that bits must be spent on encoding the shape information itself.

Scalability: MPEG-4 video offers several kinds of scalability. This means that the bit stream is hierarchically split into layers. Each layer can be decoded without reference to the layers above. In this way meaningful information can be retrieved without decoding the complete bit stream. The reconstructed quality is better the more layers are utilised for decoding. In this way one single bit stream can be used to feed decoders of different complexities and capacities, and if peeling off layers before the actual transmission, the same bit stream can also be used as basis for transmission on networks with different bandwidths available.

Error robustness: Several tools for improved error resilience are included. Most of them are optional, and aimed for use in wireless networks. The tools are therefore most powerful in error-prone environments at low bit rates.

Face and body animation: The face and body animation parts of the standard allow sending parameters that calibrate and animate synthetic faces and bodies. Two sets of parameters are defined: the face/body definition parameters, which transform a default face/body into a customised one, and face/body animation parameters, which produce the position and movement of the face/body. The models themselves are not standardised, only the parameter sets.

Coding of 2D and 3D meshes: MPEG-4 provides coding tools for both 2D and 3D meshes. The 3D polygonal meshes are widely used as a generic representation of 3D objects. Corresponding error resilience tools are also provided in order to facilitate partial recovery of a mesh when parts of the bit stream is corrupted. The bit stream is also scalable, to enable decoders to reconstruct a simplified version of the object by decoding only a part of the bit stream.

This is all described in the systems part of MPEG-4 (ISO/IEC 14496 part 1) [4].

MPEG-4 BIFS : MPEG-4 scenes are created according to the MPEG-4 standard [16]. As far as possible all nodes are implemented (both 2D and 3D) in ViSiCAST.

MPEG-4 scene composition: The different MPEG-4 audio and video objects are synchronised through referencing the same clock in the media streams [16].

MPEG-4 FlexMux is **not** used.

3.2 Sign language

The principal aim is to provide a sign language interpretation of the television programme; this is applicable to all scenarios. ViSiCAST provides two ways to transport signing in MPEG-4: broadcasting of a human signer and methods based on 3D virtual characters. The human signer is recorded against a chroma-key background, from which a key signal is derived. The video and key signal are encoded as an arbitrary-shaped MPEG-4 object and transmitted synchronously with the main programme via DVB.

For good results the video and signer should be synchronous to within a few frames. The displayed image of the signer should be at least as good as that of the main programme it accompanies as the viewer is continually watching the face and hands of the signer for the entire length of the programme.

Two types of MPEG-4 data are provided:

1. *MPEG-4 video elementary streams*. These are neither part of any compound MPEG-4 scene nor SL packetised. The streams shall comply with the MPEG-4 visual specification, and must contain all information required to decode it (including visual object sequence header, visual object headers, and video object layer headers) [3].
2. Files conformant to the *MPEG-4 file format* as described in version 2 of the MPEG-4 systems specification [4]. The files contain scene and object descriptions along with all the accompanying media data elementary streams.

Their inclusion in the MPEG-2 transport stream should comply with the specification in amendment 7 to MPEG-2 systems [5]. For the second item above, this amendment allows several options. When working with MPEG-4 scenes (item 2), amendment 7 requires access to the individual elementary streams. This information must therefore be extracted from the compound MPEG-4 file. Additionally two types of private descriptors must be made based on the information found in the MPEG-4 file: one IOD descriptor for the complete scene, and one SL descriptor for each SL packetised elementary stream. Both descriptors are to be included in the descriptor loops in the PMT.

When working with video elementary streams (item 1), information to create an MPEG-4 video descriptor must be retrieved from the video stream itself. This descriptor is also to be included in the PMT.

3.3 MPEG-4 A/V Encoder

There are various MPEG-4 players currently available: inter alia the so-called SoNG and the so-called SNHC player [14,15]. It has been established by our partner INT to use the SNHC player for the ViSiCAST project.

Extraction of MPEG-2 content

MPEG-2 A/V content is extracted from the MPEG-2 TS by hardware and fed directly into the MPEG-2 hardware decoder of the DVB SAT PCI card. For time-shifted viewing, the data is retrieved from local storage and fed into the HW decoder [9+10].

Extraction of MPEG-4 content

The ViSiCAST terminal should support the delivery of streaming MPEG-4 content as defined in ISO/IEC 13818:1996 Amendment 7 [5]. MPEG-4 A/V information as well as MPEG-4 OD and scene descriptions will be transmitted in the MPEG-2 PES. MPEG-4 files can be transmitted in parallel in so-called 14496_sections. The DVB SAT PCI card API allows access to the 14496_sections, which are utilised by the section filtering modules. In fact the extended APIs are provided by the system level DVB card driver.

4 ViSiCAST and the Multimedia Home Platform MHP

4.1 Introduction

The Multimedia Home Platform (MHP) encompasses the peripherals and the interconnection of multimedia equipment via the in-home digital network [7]. The MHP solution covers the whole set of technologies that are necessary to implement digital interactive multimedia in the home - including protocols, common API languages, interfaces and recommendations. The reference model shown in Figure 2 allows the development of high-level APIs and applications, independent of the MHP system infrastructure. This model offers system modularity through the use of key interfaces. These interfaces are able to maintain the stability of MHP systems as they evolve - both in terms of hardware and software. Backward compatibility will be supported to the largest possible extent, e.g. by using scalable applications. Each application is developed to comply sufficiently with the reference model to ensure cross-platform interoperability in a competitive environment. This should result in host platforms where the integrity of the application is protected, and its behaviour is stable and predictable (thus resulting in high quality of the service). The reference model must also define modes for data delivery, memory handling and instruction execution.

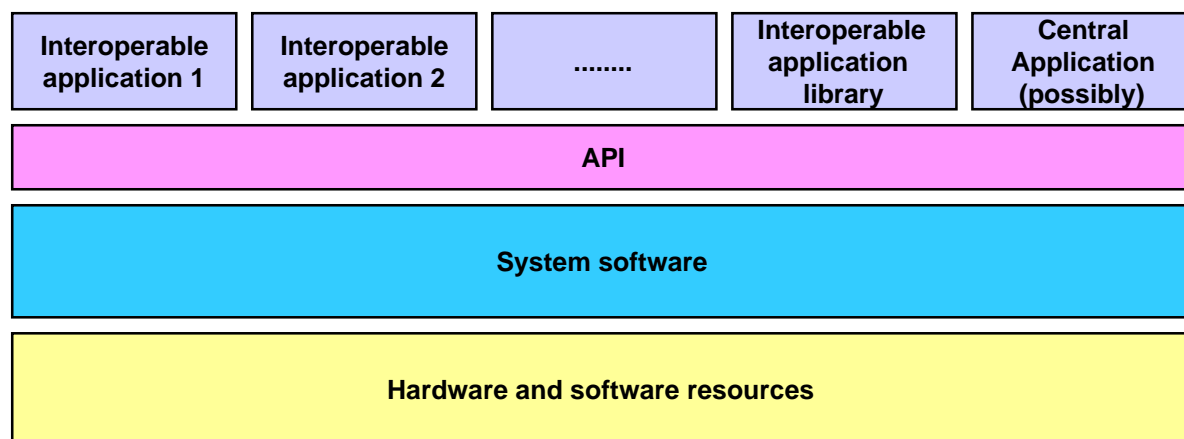


Figure 2: MHP reference model

The model consists of four layers:

- application (content, script) and media (audio, video, subtitle) components;
- the API and native navigation/selection functions;
- platform/system software or middleware, including the interactive engine, the run-time engine or virtual machine, the application manager, etc.;
- hardware and software resources and associated software.

The main system functions are:

- application launch and control, session/event management;
- security and access;
- content loading;
- navigation and selection;
- declarative content and streams presentation control;
- communication and I/O control;
- signalling, bit transport, driver and management functions.

4.2 MHP compliant STB

Basically, MHP is built around the DVB-J platform, which includes a “Java Virtual Machine” and provides a wide range of generic APIs. MHP applications access the platform only through these APIs.

Typical examples for MHP applications are

- electronic programme guide (EPG),
- information services (“super teletext”, news tickers, stock tickers),
- applications synchronised to TV content like ViSiCAST,
- e-commerce and secure transactions.

The main elements of the first release 1.0 of the MHP specification are:

- MHP architecture
- Detailed definition of enhanced broadcasting and interactive broadcasting profiles,
- Content formats including PNG, JPEG, MPEG-2 Video/Audio, subtitles and resident and downloadable fonts,
- Mandatory transport protocols including DSM-CC object carousel (broadcast) and IP (return channel),
- DVB-J application model and signalling,
- Hooks for HTML content formats (DVB-HTML application model and signalling),
- DVB-J platform with DVB defined APIs and selected parts from existing Java APIs, JavaTV, HAVi (user interface) and DAVIC APIs,
- Security framework for broadcast application or data authentication (signatures, certificates) and return channel encryption (TLS),
- Graphics reference model,
- Annexes with DSM-CC object carousel profile, text presentation, minimum platform capabilities, various APIs.

In order to simplify the development, but also to demonstrate the potential of MHP and drive future standardisation activities, ViSiCAST chose the Siemens-Fujitsu Set Top Box "Activy" (see Fig.3).

to build the ViSiCAST terminal on top of MHP. IRT made their MHP reference implementation available to ViSiCAST and ported it to the aforementioned platform.



Figure 3: Fujitsu-Siemens ACTIVY300 STB

4.3 Implications for ViSiCAST:

If ViSiCAST is regarded as an additional service it has to be stated that such a service is not an interoperable application which is based on JAVA (so called DVB-J application). Such a service has more similarities to services like DVB-HTML. This is an application which can be accessed through MHP by a mechanism called "interoperable plug-in" and defined in MHP 1.0 [7]. Figure 4 depicts the schematic scheme of the generalised MHP architecture in digital TV.

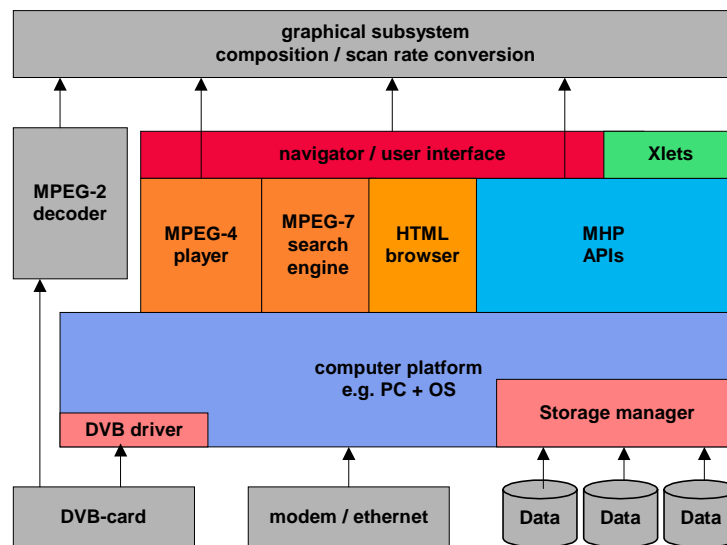


Figure 4: The MHP engine in the generalized digital TV receiver architecture

The precondition to address ViSiCAST by MHP is an extension to the existing MHP standard. Up to now no decision was taken by the ViSiCAST consortium to give a definitive priority to any of the 4 methods for delivery investigated: the proprietary BAF format, the MPEG-4 compliant BAP/FAP format, the MPEG-4 compliant video format or the transport via XML syntax, so called SiGML format. The last one can be regarded as a first-grade relative to DVB-HTML, because XML is supported in MHP 1.1. Therefore the choice of this format as a broadcast format would be the easiest way for an implementation. Nevertheless it has to be considered that the presentation engine of the existing ViSiCAST SiGML format [15] requires a 3D rendering capability on the receiver side which is not yet implemented in MHP 1.1. This applies to the proprietary BAF format and the MPEG-4 compliant BAP/FAP format, too. On the other hand, the implementation of a 3D rendering engine would require an extension of the existing MHP 1.1 standard. For reasons of high implementation costs it is, however, very unlikely that the MHP expert group will put it as an issue on the list of future extensions. We have to be aware that a future service like ViSiCAST will be for a minority of viewers only.

If we regard the 4th method to transport ViSiCAST content we are facing likewise the same problems as experienced when trying to implement a 3D rendering engine. A MPEG-4 video

decoding engine is not yet supported in the MHP 1.1 standard and requires an extension of the specification which should be achievable.

Originally, the Fujitsu-Siemens STB does not have 3D HW-support. Consequently the proprietary BAF format could not be displayed using the MHP STB. Alternatively, two independent MPEG-4 decoder were integrated: the so called SoNG player [14] for MPEG-4 Video/Audio and the so called SNHC player [15] able to play MPEG-4 compliant BAP/FAP information.

Both decoders are independently integrated into Java, with a common interface to the “Player Stub” of the user interface. The SoNG player was also used in the related IST project SAMBITS (IST-1999-12605) [13] and is described in [14]. Some further details of the SNHC player which was made available through our project partner Institut National de Télécommunications are given in the next section:

4.4 The SNHC (Synthetic/Natural Hybrid Coding) player using Face & Body Animation

SNHC along with the rest of MPEG-4 is a coding standard supported by the Systems architecture for controlling the communication and decoding of real-time media compositions in terminals. SNHC is thus *NOT* a graphics rendering standard, *NOR* an animation language.

The SNHC group is a subgroup of MPEG-4 and originates requirements and technology for co-operative integration with the other MPEG-4 subgroups within the Audio, Visual, and Systems parts of the MPEG-4 specification. The scope of work for SNHC includes human face and body description and animation, integration of animated text and graphics, coding of scalable and synthesized textures, 2D/3D mesh coding, video planes and shapes as separate scalable objects, hybrid scalable text-to-speech, synthetic audio coding, 2D/3D synthetic graphical constructs (e.g. triangle or box), and the ability to build 2D/3D scene compositions from instances of the elementary streams mentioned above.

In 1999, the version 1 MPEG-4 standard included face animation, 2D mesh coding, and view-dependent coding technologies in SNHC. In 2000, body animation and 3D model coding tools were newly added as version 2 tools in MPEG-4. Currently SNHC is mostly concentrating on producing the specification of MPEG-4 version 5, one of whose essential differential assets, with respect to previous MPEG-4 versions, is the AFX (Animation Framework eXtension) toolset.

The FBA visual tool contains the definition and coding of the Face Animation Parameters (FAPs) and the Body Animation Parameters (BAPs). FAPs and BAPs are defined relative to a neutral face and neutral body pose. FAPs and BAPs are normalised before coding to achieve independence from the face and body models. This approach allows any FBA visual bitstream to animate any MPEG-4 compliant humanoid model without downloading the model used for encoding. Humanoid models may be resident in the terminal, proprietary, or obtained by any means. In order to achieve normalisation, several dimensions of the source Humanoid (e.g. a real person) are measured in the neutral position and then the animation parameters are expressed in proportion to these dimensions. The decoder must also measure the dimensions of the target humanoid model (in neutral position) in order to invert the normalisation of the FAPs before application.

The real-time transmission of compressed Facial Animation Parameters (FAPs) and Body Animation Parameters (BAPs) provides for very low bitrate animation of a synthetic

facial/body model previously stored in or downloaded into a terminal. FAPs are normalised and define motion relative to the neutral face in order to be independent of any particular face model, BAPs work in much the same manner in order to be independent of a particular body model. Control of the face is scalable, where the most robust form of facial control provides for the expression of visemes and emotional states adequate for speech and mood intelligibility, especially if the facial animation is complemented by speech audio or text-to-speech. Set-up of Facial Definition Parameters (FDPs) and Body Definition Parameters (BDPs) provide for the connection between the ensuing FAP streams and the facial mesh/texture that represents the face, thereby offering artistic latitude to develop proprietary face/body models and animation programs while depending on the FAP/FDP + BAP/BDP standard. Time coding is adequate for the single-frame alignment of FAPs and audio/text. The multiplexing provided in Systems can join a multi-user connection with this capability.

An FBA stream can contain information about animation of one single Face or/and one single Body node. To animate several Face or Body nodes, separate streams must be used. The FBA requires some means of linking the FAP/BAP control of an animated mesh (corresponding to key facial or body features) to a specific 2D or 3D geometric model with adequate topological complexity to achieve elasticity and expressiveness. While MPEG-4 SNHC will not standardise facial and body models, a means of representing facial/body meshes and transmitting them efficiently has been needed from the outset. Some applications suited to run in real-time (such as limited synthetic environment traversal, training on 2D/3D models and remote-control animation like FBA) call for geometry compression. 2D/3D maps that augment GIS (Geographic Information Systems) or provide terrain visualisation offer further examples.

MPEG-4 includes 2D and 3D-mesh compression (3D Mesh Coding) and animation. 2D animated Delaunay meshes with implicit topological structure can be used for mesh-augmented video manipulation and dubbing. 3D Mesh Coding is a way to efficiently compress a 3-D model with enhanced network capabilities: computational graceful degradation control, incremental representation, support for non-manifold model, error resilience, and quality scalability via hierarchical transmission of levels of detail. The Topological Surgery (TS) representation is the key element in the coding of a 3D mesh. A 3D mesh can be thought of as IndexedFaceSet in ISO/IEC 14772-1:1997, and consists of connectivity, geometry, and properties (e.g. color, normal, texture coordinates, etc). The connectivity information is encoded losslessly, whereas the other information could be quantized before compression. In order to maintain the congruence of the system, geometry and properties information are encoded in a similar fashion. The output of the decoder can be directly used as an IndexedFaceSet node represented in ISO/IEC 14496-1:1999.

MeshGrid is a mesh representation with a regular structure. The MeshGrid representation consists of a uniform or non-uniform distributed 3D reference grid of points, and a mesh description attached to the reference grid. The position of each mesh vertex is obtained by a relative displacement from the reference grid point that the vertex is attached to. The mesh is finally obtained by connecting these vertices in a predefined way.



Figure 5: The ViSiCAST meshgrid

Animating an articulated 3D model requires knowledge of the position of each model vertex for each key-frame; specifying such data is an enormous and expensive task. For this reason the AFX animation system employs the bone-based modeling of articulated models that effectively attaches the model vertices to a bone hierarchy (skeleton). This technique avoids the necessity to specify the position for each vertex, only the local transformation of each bone in the skeleton being animated. The local bone transformation components (translation, rotation, scale, scaleOrientation and center) are specified at each frame and, at the vertex level, the transformation is obtained by using the bone-vertex influence region.

To address streamed animation, the animation data is considered separately (independent of the model definition) and can be specified for each key-frame.

Animating a skinned model is achieved through the updates of the geometric transformation component of the skeleton by transforming the bones and/or to the muscle curve form.

5 Synchronisation and Compression

ViSiCAST is a program-related service. This makes synchronisation a big issue. As synchronisation mechanisms mostly are defined within each standard, the main challenges arise when trying to synchronise elements from different standards. In the ViSiCAST project, need for synchronisation between the following types of elements are foreseen:

- MPEG-2 Video/Audio and MPEG-4 Video
- MPEG-2 Video/Audio and MPEG-4 BAP/FAP format
- MPEG-2 Video/Audio and proprietary motion data transported in private sections
- MPEG-2 Video/Audio and delivery of ViSiCAST content in various formats via DSMCC and Object Carousels

Between standards, very often synchronisation “within a second” is good enough for the ViSiCAST demonstrations. In general though, it should be possible to obtain better synchronisation. Additionally very tight synchronisation is required within MPEG-2 (e.g. lip-sync between the audio and video components), and between the elements in each MPEG-4 scene [11].

It is very important that the synchronisation issues are approached from both the studio and terminal sides:

- In the studio, all information put into a transport stream must eventually reference the same master clock. This is the MPEG-2 system time clock, and information related to this will be held in the headers of both PES packets and transport packets.
- The ViSiCAST terminal should be able to reconstruct the time reference and somehow synchronise its engines to be able to present the compound service properly

Synchronisation of ViSiCAST in a MHP environment is enabled through the use of DSMCC Stream Event Objects as transported in the Object Carousel. They indicate the presence and location of synchronisation events which are themselves transported on dedicated elementary streams and encapsulated in DSMCC stream event descriptors. Those events contain an ID and a time reference which is based on NPT, which is a frame-accurate media time defined by DSMCC and reconstructed from PCR information present in the MPEG-2 stream.

SNHC uses explicit compression technology in face/body animation, video/texture/mesh compression, and Structured Audio - that is, in most areas. In some cases MPEG-4 SNHC uses transformation and Quantisation techniques for coding specific lossy types of A/V objects. In other cases, such as Structured Audio or downloaded scene graphs, MPEG-4 and SNHC use the inherent compression power of transmitting lossless abstract representations of real-world and synthetic scene elements to lower bitrate demands during a session. This is typically accomplished by relying on downloads of models and their subsequent real-time rendering by terminal resources that use MPEG-4 streams. Prior MPEG, ITU, and VRML work have contributed to the technology base; MPEG-4 improves on that heritage in specific areas.

Appendix A: Results of a thesis in WP1

From October until December 2002 Miss Bianca Schröter, a student at the University of Applied Sciences at Mittweida/Germany, did a diploma work in the Institut für Rundfunktechnik in order to finish her thesis on "Tonsubstitution im digitalen Fernsehen - neue Möglichkeiten für die Einblendung von Gebärdensprachdolmetschern durch den Standard MPEG-4" . Her thesis concentrated on the intelligibility of signers (human signers as well as AVATARS) coded and transmitted via broadcast in a MPEG-4 video elementary streams. Miss Schröter discussed these under the following circumstances: A PC-based TV receiver card was used, and the decoded MPEG-4 video signal was transferred and displayed on a standard TV set.

Moreover, the following assumptions were taken as a basis: (a) For the introduction of this new service for the deaf, such as ViSiCAST, only MPEG-4 decoders (decoding MPEG-4 video elementary streams) will be available. (b) The implementation of these decoders may be done in both, software as well as hardware. (c) The avatar is used in a multimedia studio, and before transmission it is coded into a MPEG-4 compliant video elementary stream by the broadcaster.

Thus, two encoders are cascaded: the avatar building software plus the MPEG-4 Video Codec. The advantage for the broadcaster to generate the avatar in the studio rather than in the user terminal lies in the possibility to automate the creation of the sign language with the help of text and/or speech. The disadvantage of this configuration is the higher bitrate required due to the lower data efficiency of the MPEG-4 video standard in comparison to the MPEG-4 BAP/FAP format. Figure 1 shows the data rates as used by the codecs.

		Sequenz	Datenrate in kbps		
			100	200	300
REALmagic Codec: Variable Bitrate Coding, Bidirectional Prediction, Adaptive Quantization, Quantizer Range 1-31, Quantization Tools H.263, Maximum Keyframe Intervall 25 frames	Process every frame	Life Signer 1	178,73	355,97	512,37
		Life Signer 2	218,28	448,84	635,24
		Avatar	203,88	382,29	495,16
	Process every other frame (decimate by 2)	Life Signer 1	259,16	458,75	601,73
		Life Signer 2	350,73	588,64	718,63
		Avatar	287,63	400,49	447,82
Media 8 Codec: Variable Bitrate Coding, Image Quality 50% (smoother motion sharper image), Keyframe Intervall 1 second	Frames per second 25	Life Signer 1	92,4	194,5	296,5
		Life Signer 2	87,4	188,4	291,3
		Avatar	89,7	190,3	292,5
	Frames per second 13	Life Signer 1	95,2	198,7	301,4
		Life Signer 2	87,8	189,2	291,5
		Avatar	91,1	194,8	298,1

Figure 1: Data-rates preset and applied for various codecs and various application

The thesis focused on the comparison of MPEG-4 video elementary streams displaying the movements of a natural human signer with the movements of an AVATAR converted into the same format. One of the problems Miss Schröter was faced with was: there are considerable differences between the English sign language and the German sign language. Her test persons from Germany turned out to find it difficult to understand gestures performed in the English sign language. Other problems which arose were the deficits of the (so far) existing MPEG-4 software tools and the insufficient quality of nowadays' MPEG-4 video players and encoders. Figure 2 shows the typical appearance of a human signer and an AVATAR on a TV screen.



Figure 2: Appearance of an avatar and of a human signer on monitor or TV screen

Some essential results of the thesis in question are given in the next figures. The EBU "double-stimulus impairment scale method" was used with a five-grade impairment scale:

- 5 = Very annoying
- 4 = Annoying
- 3 = Slightly annoying
- 2 = Perceptible
- 1 = Imperceptible

Figure 3 shows the superiority of a human signer (sequence 1 and 2) with average rankings of 3.2 compared to an AVATAR (yellow) with a ranking of 4.6, using a Media 8 Video Codec.

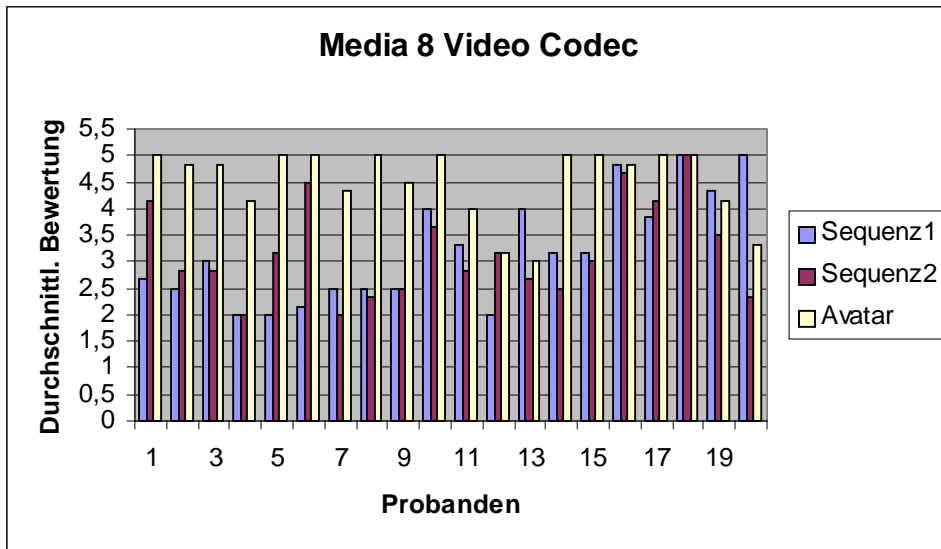


Figure 3: Average ranking ("Durchschnittliche Bewertung") for 20 German test persons ("Probanden") of avatar vs. video images of a signer

Figure 4 shows a similar result gained with a REALmagic MPEG-4 Video Codec: Here again, the average rankings of a human signer (sequence 1 and 2) were 3.1, and that for the AVATAR was 4.6.

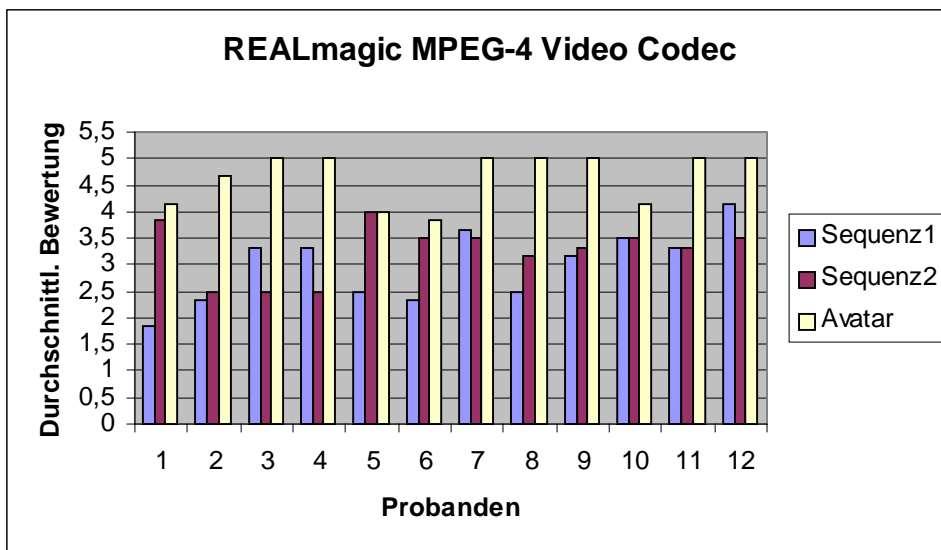


Figure 4: Average ranking ("Durchschnittliche Bewertung") for 20 German test persons ("Probanden") of avatar vs. video images of a signer

The poor results concerning the AVATAR sequence can partly be explained by the fact that gestures of the English sign language were used rather than of the German sign language. German sign language also makes more use of facial expressions than the English sign language. So, under these test conditions, the German test persons clearly preferred a human signer to a 'synthetical' person, i.e. an AVATAR.

Appendix B: Demonstration of the ViSiCAST project at IBC2002, Amsterdam, September 2002

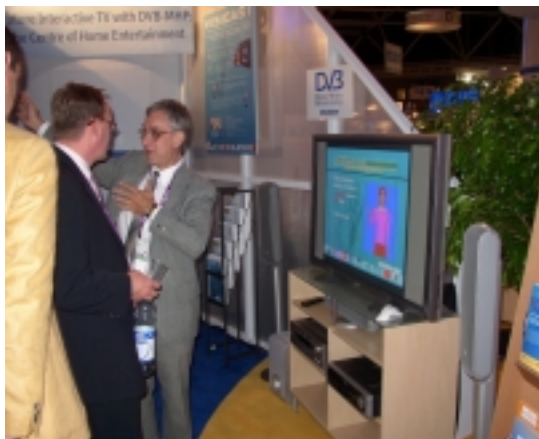
At IBC 2002 in Amsterdam the complete ViSiCAST scenario was demonstrated except broadcasting via SiGML. The required SiGML software of the partners was not available up to this date.

The demonstrations have been shown on the booth of the IRT as well as on the booth of the European Broadcast Union (EBU). It was possible to demonstrate all functionality and to operate the terminal just as a consumer STB from a remote control.

Demonstration Scenarios:

- MPEG-4 compliant BAP/FAP format displayed by the SNHC player provided by the project partner Institut National des Télécommunications and displayed on the MHP Siemens-Fujitsu STB
- MPEG-2 compliant BAF format displayed by a proprietary player (provided by the project partners University of East Anglia and Televirtual Ltd.) using a PC
- MPEG-4 compliant Video streaming format displayed by the SoNG player (provided by our partners in the accompanying IST-project "SAMBITS") of the MHP Siemens-Fujitsu STB

The demonstration system lacks performance when decoding an MPEG-4 scene in parallel to basic operations. As MHP is based on JAVA, this needs already significant resources, while an MPEG-4 SNHC decoder itself asks for substantial computational power. Consequently, the MPEG-4 compliant BAP/FAP format displayed on the SNHC player was demonstrated as a stand-alone solution.



ViSiCAST at the booth of EBU (left) and at the booth of IRT (right)

Remarkable attention was paid to ViSiCAST by the public. Several Radio and TV broadcasting stations were aired concerning this program related service in digital TV.

List of ABBREVIATIONS

AFX: Animation Framework eXtension
API: Application Programming Interface
BAP: Body Animation Parameters
BDP: Body Definition Parameters
DSM-CC: Digital Storage Media - Command & Control (MPEG)
FAP: Face Animation Parameters
FBA: Face and Body Animation
FDP: Facial Definition Parameters
ID: Identifier
IOD: Initial Object Descriptor (MPEG-4)
MHP: Multi Media Home Platform (DVB)
MPEG: Moving Picture Experts Group
OD: Object Descriptor (MPEG-4)
PES: Packetized Elementary Stream
PMT: Program Map Table (DVB)
PSI: Program Specific Information (MPEG)
SI: Service Information (DVB)
SL: System Layer
SoNGH: portals Of Next Generation
STB: Set Top Box
SNHC: Synthetic/Natural Hybrid Coding
TS: Transport Stream (MPEG)

REFERENCES

- [1] ViSiCAST, "Project description and plan", IST1999-10500, October 1999
- [2] ISO/IEC 13818-1: 1996, "Information technology - Generic coding of moving pictures and associated audio information: Systems"
- [3] ISO/IEC 14496-2: 1999, "Information technology - Coding of audio-visual objects - Part 2: Visual"
- [4] ISO/IEC 14496-1: 1999, "Information technology - Coding of 2D and 3D meshes - Part 1: systems specifications"
- [5] ISO/IEC JTC 1/SC29/WG11: 2000, N3050, "ISO/IEC 13818-1 FDAM 7: Transport of ISO/IEC 14496 data over ISO/IEC 13818-1"
- [6] ETSI EN 301 192: 1999, "Digital Video Broadcasting; DVB specification for data broadcasting"
- [7] ETSI TS 101 812: 2000, "Digital Video Broadcasting; Multimedia Home Platform (MHP) Specification 1.0"
- [8] ETSI ETS 300 468: 1997, "Digital Video Broadcasting; Specification for Service Information (SI) in DVB systems"
- [9] Specification of PCI-DVB Sat SetTopBox PC Card, Fujitsu Siemens Computers 06/2000
- [10] Interface description for DVB-C and DVB-S PCI receiver boards, Fujitsu Siemens Computers, 19-05-00
- [11] ViSiCAST Deliverable D1-1: "Direct Sign Transmission", "Solutions for the ViSiCAST Broadcast Synchronisation Problem"
- [12] ViSiCAST Deliverable D4-1: "Prototype animation System for Direct TV Transmission"
- [13] IST project SAMBITS, IST-1999-12605
- [14] IST project SoNG , IST-1999-10192, see <http://www.octaga.com/SoNG-Web/>
- [15] ViSiCAST Deliverable D4-2: SiGML Notation-Avatar Software Driver, IST-1999-10500, December 2001
- [16] Overview of the MPEG-4 Standard. ISO/IEC JTC1/SC29/WG11 N3444, May 2000 (www.cselt.it/mpeg/standards/mpeg-4/mpeg-4.htm).
- [17] [17]ISO/IEC JTC1/SC29/WG11 MPEG, International Standard IS 13818-6 "Information technology -- Generic coding of moving pictures and associated audio information -- Part 6: Extensions for DSM-CC", 1998

