

European Polytechnic Institute, Ltd. Kunovice

BACHELOR THESIS

2007

Jan VÁLEK

European Polytechnic Institute, Ltd. Kunovice

Educational branch: Electronic computers

**DIGITAL PROCESS AND TRANSMISSION OF
VIDEO STREAM**

(Bachelor thesis)

Author: Jan VÁLEK

Bachelor thesis leader: RNDr. David SKOUPIL

Olomouc, May 2007

xx Oficiální zadání xx

I affirm that I evolve my bachelor thesis separately under the command of RNDr. David Skoupil and with the list of literature used derived from literal and technical sources.

Olomouc, May 2007

I devote my bachelor thesis to leader RNDr. David Skoupil and to my bachelor thesis opponent Ing. Vladimír Grohmann. The expert control they provided me with while I was working on my bachelor thesis. Furthermore, thanks to RNDr. Stanislav Michal for providing me with useful internet links. Lastly I want to thank my family and friends for their care and assistance.

Olomouc, April 2007

Jan Válek

Content

Introduction.....	7
1 Video compression possibilities review	8
1.1 Lossless codecs	9
1.1.1 RAW	9
1.1.2 HuffYUV	9
1.2 Loss codecs	9
1.2.1 Indeo® Video 5.10.....	9
1.2.2 Microsoft H.261 a H.263	9
1.2.3 Microsoft Video 1	10
1.2.4 MJPEG	10
1.2.5 MPEG-1	11
1.2.6 MPEG-2	12
1.2.7 MPEG-3	13
1.2.8 MPEG-4	13
1.2.9 ASF, WMV	13
1.2.10 Quicktime.....	14
1.2.11 RealVideo	14
1.2.12 DivX 3.11a Alpha	14
1.2.13 DivX 4, DivX 5.....	15
1.2.14 XviD.....	16
1.2.15 DV.....	17
2 Video streaming	18
2.1 Introduction into streaming.....	18
2.2 Streaming options	19
2.3 Coding.....	20
2.4 Streaming parameters.....	21
3 Streaming data employment possibilities	24
3.1 Exploitation in education	24
3.2 Commercial usage.....	25
3.3 Own video stream – demonstration	26
3.3.1 Formats available for streaming.....	26
3.3.2 Playback video stream on www pages.....	28

4	Creating online presentation.....	30
4.1	Macromedia Flash.....	30
4.1.1	Macromedia Flash Media Server	30
4.1.2	Flash application and streaming video.....	31
4.2	Synchronized Multimedia Integration Language.....	31
4.3	JavaScript	33
4.4	SMIL – basic elements and construction	34
4.4.1	SMIL 1.0	34
4.4.2	SMIL 2.0	38
4.5	SMIL – Making a presentation in practice	39
4.6	TIME+ HTML	40
4.6.1	HTML+TIME Specification	41
4.6.2	Examples of HTML+TIME for MSIE 5.5 and 6	44
4.6.3	XHTML+SMIL Profile.....	45
4.6.4	The future of SMIL.....	45
4.7	SMIL – animations and presentation effects	46
4.8	Exploitation and resume	47
5	Description of accessible application for work with streaming data	49
5.1	RealNetworks RealSystem.....	49
5.2	Windows Media	49
5.3	Apple QuickTime.....	50
5.4	MPEG	50
Result	51
Resume	53
Used literature		54
Used internet sources		55

Introduction

Nowadays, multimedia development, computing engineering and internet connectivity are incredibly advanced. That, what a few years ago might have been bid fairly unbelievable is today common affair. Analog video cameras and cameras have already been changed from larger proportion to digital ones. Indeed, development didn't stop and new devices are coming out on the market every single day. Still at a high rate, powerful Digital cameras with higher chip resolution, digital cameras with the support of new video standards such as HDV are filling markets all around the world. This development course is totally correct. You can do almost anything with digital video without losing quality after you download it to the pc. It is possible to modify videos, to process, to convert to other video formats, to burn it on a DVD disk and via modern high speed internet transfer to the whole world. Even transfer videos real-time. Burnt DVD disk can be archived so easily without loss of quality.

Watching multimedia via the internet is still more and more popular but not even like commercial question, but I dare to say, that these branches are shifting up the education department. Students stay at home, while teachers lecture in class. Generally this type of multimedia transfer is called streaming.

In my work, I endeavor to sort out from large volumes of information from this branch only the most important cogitations and construct them into an intelligible form. The result of the work may serve as a direction as to how to stream multimedia data via the internet.

1 Video compression possibilities review

Video is nothing more than sequences of pictures going one by one. If we want to save that video to the computer and process it, modify by cutting programs and convert it into the appropriate formats (for export to another media intended for another forms of playback besides computer eg. DVD or broadcasting on the internet) we can discover the problem within size of an uncompressed video.

For example, we take a video stream in resolution 720x576 pixels (classical DV resolution in PAL¹ norm.) in color format RGB (it is by default 3x8bits, so 3 Bytes per one video pixel) and frame rate 25Hz. Simple calculation gives us the result, that if we want to save one second of video stream we need $720 \times 576 \times 3 \times 25 = 31\,104\,000\text{ B} \approx 30\text{MB}$. Today, current hard drives don't reach this transfer speed. Additionally, if we calculate with a 120GB hard drive, then we are able to save about 63 minutes of record on it.

However, even if could save a video stream in this resolution, we probably would not be able to process it or try to convert it to another, more useful, format. That's the reason why we would convert the video stream before usage. That's what codecs were made for (shortcut for COde/DECode). The Codec is a powerful tool for changing the video stream size (coding/compressing) and playing it, using the compressed file, in real time close to the uncompressed format quality. We will discuss them in the next chapter.

Different codecs, of course, have a different quality. We can see these differences clearly in the final size of the video stream and also in the quality of played video file (the compress ratio is usually 1:4 to 1:100). We can also divide a codecs into the two groups, loss and lossless.

¹ PAL (Phase Alternation Line) is one of color signal coding standards for television broadcast. Another are for example SECAM or NTSC.

1.1 Lossless codecs

1.1.1 RAW

RAW means uncompressed format of video stream or pictures. Full PAL means 720x576 and bitrate approximately 29, 7 MBps and half PAL means 352x288 and bitrate 7, 2 MBps.

1.1.2 HuffYUV

This codec compress video stream using Huffman's coding. In the best result the final size of a compressed file is up to 40% of the RAW file. This codec can compress the color format RGB² and YUV³. This codec is fast and it is for free.

1.2 Loss codecs

1.2.1 Indeo® Video 5.10

This codec was developed by Intel Corporation. It always keeps a good picture quality. You can use the setting "every picture is the key picture⁴". Using the setting 100% the quality is the final quality pretty close to the raw format.

1.2.2 Microsoft H.261 a H.263

H.261 is the standard codec for videoconferences and for video telephony via ISDN. It can change the actual data flow (compression rate of the video stream) according to the ability of the network to transfer the data. This codec chooses between 64 and 128kbps. It's usually used for video conferences and security systems.

² Color model RGB means Red-Green-Blue and it is adaptive way to mix colors used in all monitors and data projectors.

³ YUV is color model used in television broadcast in norm PAL. Model is using element vector [Y,U,V] for describe specific color. Y is brightness factor, U and V are color factors.

⁴ Key picture means that the others to the next key picture are following the setting of the first key picture.

1.2.3 Microsoft Video 1

This codec has been a standard component of all operating systems from Microsoft since version Windows 95. The quality of the final video is extremely poor. Even if we set the quality up to 100% we can see squares and other annoying defects in video pictures. The codec is slow and the video is coded by this codec is even larger than the same video coded by the lossless HuffYUV codec.

1.2.4 MJPEG

MJPEG (Motion JPEG) is built on compress individual frames using JPEG compression. In this codec, we can choose a compress rate between 1:6 and 1:16. If we use the compress format 1:8, the video quality is still very good and the data flow is about 4MBps. It is good rate between the quality and the size of the video file. The advantage of this codec is that each frame is separately compressed. It means that each frame is the key frame. Therefore, this codec is very appropriate for video trimming on a computer. This codec is immediately built in many half-professional video capture cards as hardware. Those cards are able to capture the video without any problems even on quite slow computers (computer with 300MHz is adequate). A common disadvantage of those implemented codecs is the impossibility to play the video on another computer without this type of video card (without implemented codec). Software codec compress with MJPEG is, for example, PIC-Video MJPEG Codec.

Advantages:

- Each frame is a key frame, ideal for trimming
- Hardware should be implemented
- Support interlaced frames
- Relatively high video quality

Disadvantages:

- High requirements for CPU
- High data flow

1.2.5 MPEG-1

MPEG is an abbreviation for Motion Pictures Experts Group. The aim of this group was to standardize the methods of compression video stream and build open and effective compression. Format MPEG-1 was accomplished in 1991 and in 1992 accepted as a norm – ISO/IEC/11172. It was a draft for work with video stream on resolution 352x288 pixels, 25 frames/s and data flow 1500kbps. The parameters of MPEG-1 compression are compared with the analog format VHS. Format MPEG-1 became a component of the “White Book”, which is defined as a norm for record moving pictures on CD (74 minutes of video).

MPEG compression is using I, P, B frames. I frames (Intra Pictures) are key pictures and they are compressed almost like MJPEG but with the additional option that different parts of a picture are compressed with different compress rates. P frames (Predicted Pictures) are coded with allowance to the nearest previous I or P frame. B frames (Bidirectional Pictures) are calculated as difference pictures between the nearest previous I or P frame and nearest following I or P frame. All frame sequences (from one I frame to the next I frame) we call GOP (Group of Pictures) and standard MPEG stream VCD, SVCD and DVD used the sequence IBBPBBPBBPBBPBBPBB. After all, MPEG standards do not determine rules and limitations to distances of I and P frames. In addition, the compression allowed finishes at GOP at any time and so we use the next sequence of GOP starting with the I frame. It results in a better quality of video. Compress videos which contain changeable distances between key frames we referred to as VKI (Variable Keyframe Interval). Count of I, P, B frames we should mostly set up on the basis of compressor implementation. From the view of space occupation, the I frames occupy the most space. The others are P frames and B frames occupy the least amount of space. MPEG-1 compression is not appropriate for trimming due to the distances of key frames. Almost all trimming, editing and processing programs are able to export video stream to MPEG-1 format. This codec is one of the most rampant and it is able to play it almost on every computer and on about 95% of DVD players. We can stream this format. However, this codec is outworn nowadays, yet is still the most compact format. If we compare it with other codecs, its quality is worse than others because if we want to reach a high quality of picture, we need many more bites for compression than with another codecs (DivX, XviD).

Advantages:

- High allowance in software players, compressor and decompressor is free
- Using for Video CD
- Advisable for streaming

Disadvantages

- Do not support interlaces frames
- Low video quality at a low bit rate
- One constant bit rate
- Not advisable for trimming

1.2.6 MPEG-2

After accomplishing MPEG-1, standard people started to use it and they tried to use it for high video resolution. They encountered some problems due to the MPEG-1 being unusable. The MPEG-1 supports only the compress of whole frames. It doesn't support the compression of interlaced frames. The format MPEG-2 was accomplished in 1994 and became the standard of compression digital video stream. It was drafted in in order to reach the quality of broadcasting video stream. It brings the support of interlaced frames compression (half frames). Furthermore, it supports the variable bit rate which means that in more demanding scenes, it uses up more bites for compression and in less demanding scenes it uses up fewer bites. It also supports constant bitrate.

It acquired a higher quality of picture than in the MPEG-1 in the same bit rate and full resolution 720x576. On the other hand, the main disadvantage of the MPEG-2 is the very high charge of the processor upon playback and practically no difference in quality at low resolutions compared to the MPEG-1. For streaming at low quality, it is more advisable to employ the MPEG-1 and in full resolution and high bit rates it is more advisable to employ the MPEG-2.

Advantages:

- Used for SVCD, DVD
- Used for digital broadcasting (DVB – Digital Video Broadcast)

- High quality at high bit rate (6MBps and more)
- Variable bit rate support

Disadvantages:

- Hardware or software player support is needed in computers. Software players insist on processor charge.
- Low quality at low bit rate
- Not useful for trimming

1.2.7 MPEG-3

It had to be used for HDTV (High Definition TV) but the development of the MPEG-3 was terminating because the MPEG-2 fully matches the requirements of HDTV.

1.2.8 MPEG-4

MPEG-4 was developed by Motion Experts Group. It is not a strict definition of compression and compress algorithms. It is only a group of parameters and properties, which the compressor must execute for the compatibility of the MPEG-4. We know many MPEG-4 implementations, which we have chosen from definition of the MPEG-4 only for things, which are for the best for the given format. Codecs use compress forms of MPEG-4 are for example Microsoft MPEG-4 v1, v2 and v3, DivX 4, DivX 5, XviD and others.

1.2.9 ASF, WMV

Microsoft Corporation had streaming video coverage in mind, which companies like Apple and RealTime reached with their QuickTime, MOV and RM formats. Microsoft developed their own format ASF (Advanced Streaming Format), used above all to video stream. ASF is similarly a video format and compression which comes from AVI and allows only Microsoft MPEG-4 compression. Microsoft introduced the WMV format too. It is the newest ASF format. ASF only partly implement MPEG-4 compression because they don't support B frames.

Advantages:

- Advisable for streaming

Disadvantages:

- Format finality. Impossible for using in other programs except in those from Microsoft. It prohibits converting ASF to other formats
- The highest resolution is 352x288. Compressor discards frames to keep data flow.
- ASF format is filled in with redundant data for keep data flow and the final file is larger even by up to 25% than direct copy of video stream

1.2.10 Quicktime

Quicktime is a format from Apple, which was very expanded and interesting because it competes against the MPEG. It is compatible with PC and MAC platform and uses compression from between 1:5 and 1:25. Today, it is used on presentation CDs and for video streaming. Although this format isn't the best today and won't be advised it for use. We need a special player, which isn't a part of operating systems and we must download it from the Apple internet pages. Apple charge that Quicktime is fully compatible with MPEG-4 but quicktime 6.0 doesn't support the real MPEG-4 stream.

1.2.11 RealVideo

Real Video and Real System G2 are compress formats developed by the Real Networks Company. It has conformable properties to Quicktime but it is more suited to video stream compression.

1.2.12 DivX 3.11a Alpha

DivX 3.11a Alpha is not a legally modified version of the ASF MS-MPEG-4v3 codec. Microsoft in beta version of this codec allows saving it in AVI video format, but in final version this option becomes disabled. Though one computer pirate changed the final codec and that it was possible to save the video in AVI format. He also changed some bad properties and the new codec, whereby the maximum resolution 352x288 was not restricted, has been developed.

This non-legal codec changed the whole digital video world. You can save up to one hour of video stream on to one CD disk at a very comfortable quality. If you reduce the bit rate you should save much more on CD but you lose quality. Today, this codec has been achieved and provides no reason to stop its legal application.

1.2.13 DivX 4, DivX 5

The group that has fingers in modified DivX 3.11a Alpha codec decided to make their own codec. From the base project called OpenDivX come the first version called DivX 4, which was available even with source codes but the quality was not as good as in the DivX 3.11a. DivX 4 codec. It allowed a few compress variants. One passed with a specific data flow, one passed with a specific quality and two others passed. The first of the above-mentioned compress variants tries to adhere to specific data flow. This variant was found many times not to adhere to and make coded video larger than the supposed one. Variant with specific video quality try to code video and adhere laid video quality. The extempory size of the final file is found to be disadvantageous.. The last variant also passes through the compress video. In the first pass, the video file is analyzed and the results are recorded in the log file. Upon the second pass, information from the first pass is used and data flow is used. In scenes with a higher bit rate compression DivX 4 use I and P frames. It supports variable distance I frames (VKI) too.

Version DivX 5 codec

- DivX Standard – standard version which is free. It does not contain all tools available in full version
- DivX Pro GAIN – full version which installs software to the system for displaying internet advertisements
- DivX Pro – full version without advertisements

Since the version DivX 5 is this codec closed without source codes. The codec is compatible with the MPEG-4, compressed into MPEG-4 Simple Profile format and it can play older versions of DivX, MPEG-4 Simple Profile, MPEG-4 Advanced Simple Profile a H.263 (videoconferences). DivX 5 uses advanced techniques in compressing and it has a 25% improvement on the version DivX 4 in the same file size.

DivX 5 has integrated some tools/filters and it is able to change video size directly upon coding. It is able to apply filter, crop video and other elements. Furthermore, it implements algorithms for raising compression using a psychovisual model. Using this model results in better compression without lost of quality because it uses knowledge of the imperfection of the human visual system. It implements a two-pass compression, as in B

frames. Moreover, it applies motion global compensation, which is an algorithm which optimizes compression for pan, fade, zoom, blaze changes (explosions), stagnancy fields (water) and others. This codec can export strictly into MPEG-4 format too and the conversion between it and AVI format. As we see, codec DivX 5 brings many things but the main disadvantage lies in the paid/advertisement full version.

Advantages:

- High quality of video stream in low data flow
- We can use arbitrary resolution divided by 4 up to 1920x1088
- Advanced compress techniques
- Color formats YUV and RGB supported
- MPEG-4 compatible

Disadvantages:

- High CPU requirements
- Playback almost on computers

1.2.14 XviD

In the moment when OpenDivX becomes closed, this situation is unsuitable for some programmers working on OpenDivX. They took codes from open source OpenDivX and started to develop their own codec called XviD. XviD is MPEG-4 compatible again and implements many properties of MPEG-4. However, it doesn't support two-pass coding (B frames) yet. XviD codec contains many settings and for reaching high quality of output it is necessary to know about this codec a lot and the correct setup of codec is very difficult. It can produce MPEG-4 compatible data flows.

Advantages:

- Codec is free
- High video quality at low data flow
- We can use arbitrary video resolution divided 4 up to 1920x1088
- Many codec settings
- YUV and RGB color formats support

- MPEG-4 compatible

Disadvantages:

- High CPU charge
- Playback almost on all computers

1.2.15 DV

I speculated where to insert this codec for a long time. In many articles it is written, that this is a lossless codec but the reality is different. So if it reaches the compressed rate 1:5 it results in information loss. After all, this compress algorithm is very accomplished and the final picture is comparable with lossless compressions.

Advantages:

- Each frame is the key frame
- High quality of video comparable with professional devices

Disadvantages:

- Large data flow

2 Video streaming

In simple terms, we can say that streaming is real-time one way multimedia data transfer via digital networks. 'In real-time' means that on the receiver's side we are able to display incoming data directly without waiting for the whole file to download.

2.1 Introduction into streaming

Upon streaming, the data flow isn't saved on the receiver's hard drive but it is played directly. It means that this method of multimedia data transfer brings some advantages and disadvantages:

- We needn't wait for the whole multimedia file to download, which is often quite large. Multimedia content is available quite directly. In use, the receiver of multimedia content needn't wait for complete downloading, for example, in the case of long recording, yet it may be play directly. We can possibly move on a time line. It is possible to glance over the record and make out the thing we are finding are contained in the record.
- Upon streaming, data is not saved on the hard drive, so it isn't possible to play records from local hard drives without the possibility of connecting to the internet. But applications, which can capture streaming data and save it to the local hard drive for later usage without connectivity to internet, exist. Those programs are for example CSoft Stream Down or MPlayer.
- For receiving streaming multimedia data, a specific internet connection is necessary. For receiving sound in-voice quality connectivity, 40kbps is adequate. If we want to transfer quality stereo sound (music) we need an internet speed of about 140kbps and the highest demands for internet transfer capacity of line are necessary for transferring video data where we need up to 200kbps. In whole streaming we finally chose the composition between quality and accessibility. At sound records we can choose a number of channels (mono – voice, stereo – music) and a bit rate (for sound record fully equal 32 kbps to channel and for music minimally 64kbps to channel). We can choose the frame rate in video record (in presentations fully equal 10-15 frames per second) also the size of the image (we choose that in reference to

readability and sound articulation) and compression. If we want to reach the best results, we have to use the most modern compress formats and settings in accordance to the concrete case.

- In the case of slow or fluctuate connection, the playback can result in a multimedia stream drop-out or does not allow us to play stream. One of the eventual resolutions is to offer a video stream in more transfer speeds where lower transfer speeds mean reduced content quality.

2.2 Streaming options

- **On-demand** mode – multimedia content is located on the server and the computer sends only requested pieces of content to the client
- **On-line** mode – multimedia content is just in real time forming on the streaming server and transfers via the transferring server to clients. It means live broadcasting for example television or radio via internet.

If we want to stream data using a server we need to recode them to format available for streaming – into format accepted by stream server. Most employed are RealMedia, WindowsMedia and QuickTime. Those formats are qualitatively comparable and most expanded.

The main advantage of WindowsMedia formats are their native support in Microsoft operating systems but we cannot playback them on other operating systems. The main advantage of RealMedia formats is their non-dependence on the operating system but it is always necessary on the client side to install support software for playback. QuickTime is less expanded then other two and depends on some operating systems (Windows, Mac). We can place some common multimedia formats like MPEG-1, MPEG-2, Flash or MP3 on a server. The disadvantage of the MPEG is almost 5x higher data flow then Flash or WindowsMedia namely in the same output picture quality.

2.3 Coding

Producer⁵ is using for coding. Recoding data may be directly located on the broadcasting server (mode On-demand) or they may be decoded in real time and they are located on the server in sequence (mode On-line). An On-demand mode is situated simpler because for video recoding we aren't under time press and we can allow for example multi pass video processing – longer time of processing. At On-line mode data, the data must be offered constantly with minimal delay which may be often in rank minutes.

For coding multimedia content into specific stream format we use a program generally called encoder. For coding audio/video into format WindowsMedia we use program Windows Media Encoder, for coding into format RealMedia is used Real Producer Basic. Both are available free to download on their official pages.

After installation and run concrete encoder we can change:

- *Multimedia data source* – video/audio file is the most used source. Other options are devices such as microphone or camera.
- *Coding parameters* – above all we can choose the size of output data flow (in kbps) and the type of data flow (CBR/VBR it means constant/variable), coding quality (low/middle/high, one pass/multi pass). Similarly, we can change properties of audio/video stream (sampling frequency, mono/stereo sound, resolution and frames per second).
- *Target of recoded data* – most used is saving into the file and at On-line streaming mode are recoded data real time constantly sent to the streaming server

After recode, it is possible to replay the streaming content from the local hard drive and right away review a final quality of the stream. If the final quality isn't adequate it is necessary to change the coding parameters.

After that, it lacks the recoded file and locates it to the chosen directory on the streaming server. Multimedia content will be accessible via the URL link from the web.

⁵ It is powerful computer.

Your URL prefix with instructions for connecting your directory will be sent after registration.

2.4 Streaming parameters

The effect of optional parameters of coding for final multimedia content quality:

Size of data flow – if the data flow is larger than the output quality, is approaching input the quality. The tendency is to reduce this data flow for the possibility of playback stream on the client side with slower connection to the internet. The size of the data flow is chosen separately for audio and video. Acceptable quality of audio is 32kbps and more and at video in resolution 320x240 and 30 fps may be acceptable at 200kbps.

Type of data flow – at constant flow is established like size without the view to how many motion scenes are in the video. At variable flow, the size of flow is dynamically accommodated to the amount of motion in the scenes so is it more effective form of coding. Temporary peaks in the variable flow may at slow connected clients raise drop-outs at playback. If variable flow settings are allowed then it is better at to set up average size plus it's maximal and minimal size.

Quality of coding – in multi pass video coding, much better output quality is attained but coding is running much longer. At On-line coding, it is possible to only use one pass coding. If we set low/middle/high quality of coding, we choose different algorithms for one coding pass. Larger demand quality means more time for coding.

Video resolution – The higher resolution means more captured details in picture but a higher necessary bit rate. It is unreasonable to set up a higher output resolution than input resolution. In lower input resolution, it is better to set up lower output resolution. At On-line streaming mode, such a powerful computer is often unavailable which is real time and is able to code in high resolution.

Sampling frequency of audio – higher sampling frequency means a higher quality of output sound and we need a higher data flow. It is unreasonable to set up higher sampling frequency of output than sampling input. For capture speak equal sampling 8-16 kHz for CD quality 44 kHz.

Number of frames – a higher number of frames means that final picture is more fluent but demand on the data flow is higher. For the highest fluent equal to a set up between 24-30 frames per second at a very small dynamic or if we don't want high fluent we can reduce the number of frames to 1-10 per second.

Accessing multimedia streaming content to the user is often realized via classic hyper-link to subdirectory of streaming server where the multimedia content is located. For function indemnity is mostly demanded on the client side upon install support of the streaming format.

Multimedia stream exploitation is mostly used in a section of presentations, live broadcasts, education and e-learning which mostly benefit the main advantage of streaming and it is immediately accessible via multimedia content via network by means of a web link.

The streaming system consists of several parts. Data is streamed by means of the streaming server and via network is transmitted to client computers. In the case of on-demand transfer, each client may have different requests and the server generates him his own data flow.

In the case of on-line transfer abstractedly possible to generate one stream which will be functional on the network and all clients download the same stream. This type of transfer is called multicast. It is not possible to use it on the internet where we can use only unicast where data flow is generated for each client on the streaming server. The network and streaming server is overflowing in dependence on a number of clients. For example, the bit rate of one flow is 300 kbps (lower video quality) and 30 clients are connected we will get a final out-coming data flow from server 9 Mbps.

From preview information results demanding quality and speed of connectivity aren't just on the client side but also on the streaming server side. The server should be located in a place with a superior internet connection with a high speed connection. More effective would be sharing one infrastructure and to fulfill the task in this manner. Indeed, it will be necessary to have a more efficient server and a faster connectivity but final costs will be much lower even if each company buys their own techniques.

Upon on-demand streaming there are no problems with the transfer of data to the streaming server. Data is modified and saved on the hard drive via the internet on streaming server (in other case we can send data to the streaming server on CD, DVD or another medium). A worse situation begins in the case of an on-line transfer when it requires transfer data to the streaming server in real time. One of the available solutions is to separate the process of coding from the streaming server – principally it could be one computer. In some less demanding applications, it is possible to work like that. The producer captures data from input sources; in real time converts them to accessible format with competent parameters and via network transfer to the streaming server. The streaming server serves as an on-demand transfer distribution to clients. The producer can generate several qualities and then many speeds of data flow. Multimedia data processing on-line (mostly video) is very demanding on the part of the producer exactly in cases when we need to generate more than one quality of output. Requirements on connection speed between the producer and the streaming server aren't high because only one copy of each stream quality is transferred by connection.

For coding multimedia content into concrete stream format, we use a program called encoder. For coding audio/video into formats Windows Media we use the program Microsoft Windows encoder and for re-coding into format Real Media we use Real Producer Basic.

3 Streaming data employment possibilities

Video streaming exploitation is very wide. We can share Multimedia content not even via the internet but also via local computer networks and we can use a higher quality because of higher data throughput of the network.

3.1 Exploitation in education

Electronic support in education is exploited more today than ever before. There are many advantages for students and teachers (for students it is e.g. study anywhere and any-time with their own study rate and for teacher it is e.g. after put forth their initial effort they make education easier e.g. if they check home work). We can divide electronic support in education into two parts – online and offline education. Offline access is based on study materials distribution on specific data media (today usually on CD), however online education is using functions implementing in Learning Management Systems (LMS). These systems are applications intended for managing and controlling all activities related to education process organizations. The main attributes of LMS are: user and class administration, educational materials distribution, test evaluation, communication tools support (e.g. discussion forum, e-mail, and video conference) etc.

The basics of education electronic support are naturally the electronic education courses. If we want an electronic course to be effectual it is necessary to detain the student; for the student not to be allowed to only be subject to passive information but he must be continuously actuate and it must be introduced to a system of feedback. We may achieve it through interactive elements in education course (e.g. blind maps) or other elements like discussion forum or chat.

This work is aimed at the possibility of video streaming exploitation in education support especially in cooperation with other media types (pictures, animations, text) because the whole video streaming playback used to be without problems. For example, we can playback a video record of a lecture which is attendance with presentation and animations. All methods listed below are rated in reference to electronic education support and with options to distribute multimedia using LMS because these systems have limitations which can disfunctionalize some methods.

I mention some criteria for evaluating the characteristics of using each one of these methods (I proceed from actual terms and e.g. on CESNET is already running streaming server and it is not necessary to purchase any other).

Software and hardware demands - In these criteria are demands to software and hardware accessories necessary for video streaming synchronization availability. Next, we have to think about software on the client side for using that method.

Course creation – methods for cooperation video with other media that are so difficult for users who are not able to use it are not predestined to success. It is necessary to create education materials as easy as possible after we install the necessary software.

Browsers support – educating materials are distributed to students via video stream on the internet where the student is able to watch it. Students can watch video stream on many operating systems. It is needed for support of used method in most expanded browsers at least.

3.2 Commercial usage

Today, many commercial servers exist, where anybody can publicize their own videos. This situation is caused by the advanced technology in digital video processing and computer network branches where many people keep adequate hardware for digital video processing and have good connectivity to the internet for publishing and watching streaming data.

The most expanded and best-known servers for video sharing are e.g.:

- youtube.com
- metacafe.com
- video.google.com
- ifilm.com
- putfile.com

Many TV stations have started in the last few years with broadcasting their programs via the internet like stream too. Not only that some TV stations support watching their program in real time but some stations are making archives of their programs which we can watch along its actual time.

Some non-commercial TV stations are broadcasting only via the internet. In the Czech Republic, TV stations broadcasting on the internet are e.g. Óčko or ČT24.

3.3 Own video stream – demonstration

In previous chapters I mentioned digital video problems and its usage for streaming only abstractedly. In this chapter I intend to inspect practical demonstration of video stream realization and process of its presentation on WWW pages.

3.3.1 Formats available for streaming

If we want to stream videos, we need streaming servers with appropriate application which can produce video stream. In my view, the best for streaming is format WMV. Not in light of video quality or size of final video file but clearly in light of accessibility. It is a period when almost all users of the internet use the Microsoft Windows operating system as their main operating system. Microsoft Windows is supplied with WMV video file player – with Windows Media Player.

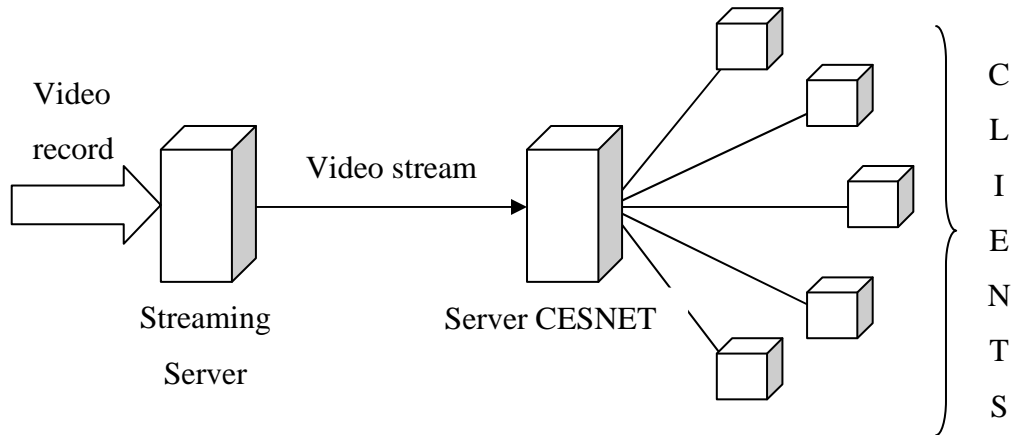
So, the video that I had chosen for streaming I converted into WMV format. I chose two different qualities. First data flow 500kbps and second 100kbps.

Now I am standing ahead of the problem where to run the streaming server. For a trial run, the connectivity speed of server is irrelevant. Nevertheless, I thought about a better connection for school if it wants to offer for students video streams of lectures in future. Therefore, I addressed the CESNET⁶ with eventuality of exploitation of their video server⁷. Their answer was so helpful and they put my video demonstration streams on the server. Next the CESNET is able to cooperate with the school and publish for EPI s.r.o. their video streams of lectures. Except for the possibility of using their video server for streaming mode On-demand (streaming video file is save on video server) they maintain live stream too. Streaming data is sent to the server and they are published. Two possibilities of how to stream videos exist. It depends on the application of use and server possibility.

⁶ Main internet provider in Czech Republic (www.cesnet.cz)

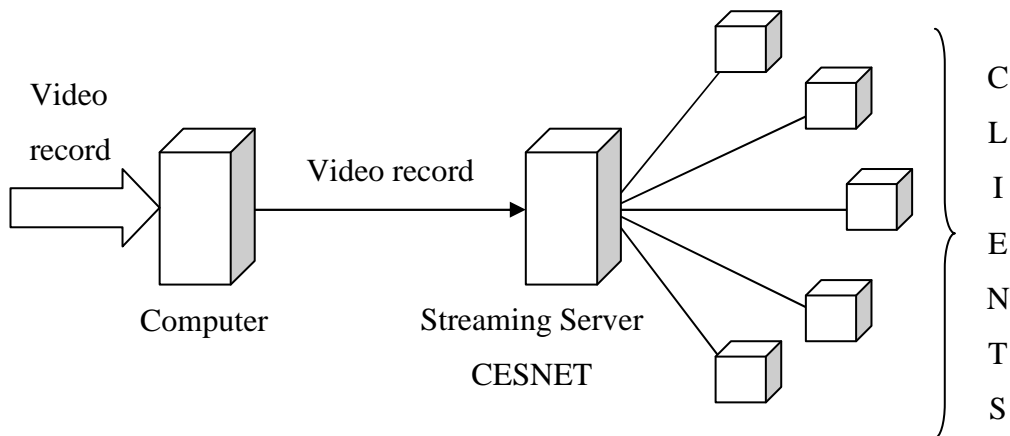
⁷ Streaming server on CESNET ([videoserver](#))

First option (Picture 1.) is to create on the broadcasting side (higher hardware demands) and then to create stream via CESNET just forwarding (concatenate) to clients. This option we can use e.g. at WMV format.



Picture 1. Stream Concatenate

Second option (Picture 2.) is to send whole data of video records to the CESNET server. Data is coded on the server and published for clients like stream. In this option the computer serves on the broadcasting side just like conciliator and there are no high requirements for hardware.



Picture 2. CESNET like streaming server

3.3.2 Playback video stream on www pages

We play video stream on www pages via inserted object in our case – media player. We use (X)HTML tag *OBJECT*, inserted into the code of the page between *BODY* or *HEAD* tags. *OBJECT* is a geminate tag and has many optional parameters which work directly with inserted objects and affect not only its visual form on www pages but even its behavior and function. Into main options of tag *OBJECT* belong *TYPE* and *DATA* which are the defining format of streaming data and path to streaming data file. An example of basic code for inserting video stream:

```
<OBJECT type="video/x-ms-wmv"  
DATA ="http://server/stream.wmv">  
</OBJECT>
```

To *OBJECT* tag are applied to other important attributes such as *WIDTH* and *HEIGHT*, which define video size.

```
<OBJECT type="video/x-ms-wmv"  
WIDTH="320" HEIGHT="240"  
DATA ="http://server/stream.wmv">  
</OBJECT>
```

Between *OBJECT* tags we can insert tags *PARAM* which aren't geminate. This tag can directly affect properties of inserted object. In media player we can use this code:

```
<OBJECT type="video/x-ms-wmv"  
WIDTH="320" HEIGHT="240"  
DATA ="http://server/stream.wmv">  
<PARAM name="AUTOPLAY" VALUE="false">  
<PARAM name="CONTROLLER" VALUE="true">  
</OBJECT>
```

The parameter *AUTOPLAY* means that video can be played immediately after the object is loaded or the playback will be waiting to the user to play it. The parameter *CONTROLLER* affect that an object will appear with a control panel with active components or not.

4 Creating online presentation

4.1 Macromedia Flash

Exploitation of applications created in Macromedia Flash, streaming video and other media from the view of electronic education support very good because we can make very effective multimedia and interactive education materials and the Macromedia Flash Player is able to be installed into almost all browsers. Thus, final presentations can be watched almost everywhere. Video stream is possible in Flash applications to be employed by relatively new products of Macromedia Company – Flash Media Server.

4.1.1 Macromedia Flash Media Server

Macromedia Flash Media Server (FMS) combines many features which allow making many different interactive multimedia applications working in real time (e.g. video conferences, video on demand) the advantage of which is that they are creating compact output and situations like e.g. open video in new window or separately in extended player do not happen. Considering the theme of this work is constituent to streaming server function support. For streaming video FMS uses own codec (files with extension FLV) which is comparable with common using video codecs. For re-coding video to this source close format we can use coder supply with Flash development kit. In transmission FMS detects quality of client connectivity and from result it comfort quality of video and transmission speed.

Flash Media Server can be run on these operating systems:

- Windows 2000 Server
- Windows 2003 Server Standard Edition
- Linux Red Hat Enterprise v. 3.0
- Linux Red Hat Enterprise v. 4.0

Server hardware demands:

- Pentium 4, 3.2 GHz or higher
- 1 GB RAM

- 1 Gbps Ethernet card

4.1.2 Flash application and streaming video

For creating an education course which will contain Flash animation or applications is important have Macromedia Flash software installed. Using functions of Flash Media Server is possible in these applications located on the FMS run streaming data very simply. Continuous with this are the existing components and it is possible to control the whole video (play, stop, switch to another frame, etc.). It is possible to join all animation and application with specific video and react to the specific position of our video or change the video position according to client reaction.

Using this method as electronic education support has one big advantage: we can run whole Flash application on almost all browsers. Another advantage is that the whole application and all synchronization is transferred into one object (Flash application) and it means that the Flash player does everything by itself. The disadvantage is that for creation of self application demand installed Macromedia Flash program and for fast creating education materials we have to have very good knowledge of this environment. Another disadvantage is that the streaming video must be in FLV file format. However, this disadvantage isn't radical because Macromedia Flash is supplied with a coder into FLV format which supports the following formats: asf, avi, dv, mov, mpeg and wmv. Thus, it isn't possible to use a common streaming server but we have to use FMS (Flash Media Server).

4.2 Synchronized Multimedia Integration Language

Synchronized Multimedia Integration Language (SMIL) is a language released by W3C partnership, which is specializes in multimedia. The first specification SMIL 1.0 was released in 1999. SMIL tries to eliminate absences of common HTML pages (their statics) and allows the development of interactive applications and animations. SMIL can integrate multimedia objects into HTML pages. It works with existing multimedia objects and it only describes its content, time their visibility and its behavior. We can resume the basic functions of language into a few points:

- Strict description of presentation appearance and location of individual objects
- Description of changing presentation in time line
- Description of object interactivity

- Parameterization of presentation

SMIL is markup language and all is based on XML specification. It means that:

- Case sensitive
- All tags must be close correctly (`<tag>...</tag>` or `<tag />`)

SMIL accommodates with a relatively small count of different XML tags. It is possible to create plentiful interactive presentations. Root bundle is tag `<smil>`. It contains two parts: `<head>` and `<body>`.

Section `<head>` describe appearance of presentation; defines eventual object position and many options of displaying and hiding media objects. Meta data (information about author, valid period, key words etc.) can be inserted here too.

In the section `<body>` objects and their mutual synchronization are used – objects can be displayed parallel or sequential (tags `<par>` and `<seq>`). These options can be combined.

SMIL discriminate some media types. Media type used tag and used formats for specific media (using depends on used player) are mentioned in these points:

- Text (`<text>`) ... txt, html;
- Stream text (`<textstream>`) ... only for Real Player;
- Picture (``) ... gif, jpg, png;
- Animation (`<animation>`);
- Audio (`<audio>`) ... wav, au, mp3;
- Video (`<video>`) ... mpg, avi, mov, rm.

SMIL allows for manage presentation by events. A specific object can be run (end) on an event when another object has been ended, run or if a client clicks on it. Events are defined by attributes *begin* and *end* in objects.

Another function of SMIL is the possibility of accommodating presentation by specific input parameters. It can be: screen resolution and color depth, internet connection speed, client language etc. This parameterization allowed e.g. video playback with accor-

dant speed, shows texts in language which client understands and any other options which improve the quality of presentation.

Presentations are made with SMIL and can be played by applet in any browser (in this case is limitation in formats – we aren't able to show video) or via external program (e.g. Real Player). This external program can be implemented like plug-in in browsers so than we can play the presentation in a browser window.

Finally, the language SMIL works with individual objects like with complex (it doesn't matter what is the content of objects) and it is described by the language based on XML which we can edit so quickly and for creating and editing a presentation. Therefore, we make do with common text editor (except editing objects). Here is the huge problem in this option. Today Real Player exists for almost all operating systems so one problem is here – sometimes the presentation has to be opened in an external window and it isn't the best for education course.

4.3 JavaScript

JavaScript is a scripting programming language which is processed and executed on the client side and it can control browser behavior. It is inserted both straight into HTML code or it is written in external file and in HTML is link to this file. The main properties of JavaScript:

- Works on client size only (limited functions – not allow to accessing files)
- Case sensitive
- Support object oriented programming
- It is interpreter – needn't be compiled

With JavaScript we can achieve similar products like with SMIL language and Java applet. We can use video as well in this combination. Video is playback with player (it is necessary to control player by JavaScript).

There is the greatest problem in JavaScript with correct function of written text in different browsers. In the meantime, we succeed in joining video with other media types only in Internet Explorer. However, it is deficient even the Internet Explorer is the most popular. We should find any other way of displaying also in other browsers. Another disadvantage of JavaScript is difficult editing in the final presentation. The best resolution is

to make the initial effort to make API interface which would guarantee the correct function in almost all browsers and, on the other hand, will allow for simple creation and editing presentations. If we use JavaScript it isn't possible to apply special software and we can use simple text editor.

4.4 SMIL – basic elements and construction

I shall now turn my attention to SMIL support in different programs and I have already explained SMIL technology theoretic in the previous chapter. Now I want to explain some of basic SMIL language elements.

4.4.1 SMIL 1.0

As first we look on SMIL 1.0 specification. It is very simple language which becomes by not many marks. Most important part of SMIL document is DTD:

```
<!DOCTYPE smil PUBLIC "-//W3C//DTD SMIL 1.0//EN"
    "http://www.w3.org/TR/REC-smil/SMIL10.dtd">
```

Root element of document is element `smil`. SMIL document is divided into two parts (almost like (X)HTML) HTML document). First part is composing by element *head* and second by element *body*. Final document may be:

```
<!DOCTYPE smil PUBLIC "-//W3C//DTD SMIL 1.0//EN"
    "http://www.w3.org/TR/REC-smil/SMIL10.dtd">
<smil>
<head>
    Dokument head
</head>
<body>
    Dokument body
</body> </smil>
```

Document head

Head of document contains common information and settings of whole document. In head we can set up background, size of presentation, individual element layout etc. Besides element *head* can contain some other elements:

- *Meta* – application similar in (X)HTML document
- *Layout* – it set up layout of document

For layout of document determination layout element may contains another two elements – *root-layout* and *region*. Element *root-layout* set up properties of all document using some attributes – *background-color*, *height*, *width*, *title*. Element *region* generate imaginary squares where multimedia data can be inserted later. So at inserted picture we set up where it has been inserted. Element *region* has these attributes:

- *Id* – region identifier
- *Background-color*
- *Left*
- *Top*
- *Height*
- *Width*
- *Z-index* – this element solve problem of overlap of two or more regions (like in CSS)

Elements *region* and *root-layout* are blank elements.

Document body

A body is made by element *body*. Listed elements are inserting individual media:

- *Img* – picture
- *Audio*
- *Video*
- *Text* – text object (HTML, TXT)

For these elements exists many common attributes:

- *Src* – source URL
- *Alt* – legend
- *Dur* – duration on time line
- *Begin* – begin on time line
- *End* – end on time line
- *Region* – id of region where object has been inserted
- *Top, left, height, width* – size and location

Time line and synchronization

Attributes *begin*, *dur* and *end* works with time line. The time line begins with zero (from document launch). Attributes *begin* to define the second when the object has been loaded. Attribute *dur* define duration of object. We can use the attribute *end* which sets up the second of the objects closely.

Individual objects are possible to group and then synchronize with elements *seq* and *par* which we can dip into.

- *Seq* – defines the sequence of display (if it contains more than one object, objects are display sequential one by one where the attributes *begin* to set up a time delay from the previous object)
- *Par* – defines the parallel display of objects (they are display in the same time where attributes *begin* to set up time delay from the whole presentation launch)

Content check

Content check means the conclusion of a competent media type. SMIL can check the properties of a client computer (language, resolution etc.) and then can afford user alternatives. For example:

```
<switch>
  <element evaluative_attribute="value" />
  <element evaluative_attribute="value2" />
</switch>
```

There are many useable attributes:

- *System-bitrate* – connection speed in bps (28800, 56000...)
- *System-language*
- *System-screen-size* – screen resolution (640x480, 800x600, 1024x768...)
- *System-screen-depth* – screen color depth

Language assignation example:

```
<switch>
  <video src="video-cz.mpg" system-language="cz" />
  <video src="video-sk.mpg" system-language="sk" />
  <video src="video-fr.mpg" system-language="fr" />
  <video src="video-en.mpg" />
</switch>
```

This code detects whether the playback video is in Czech, in Slovak or in French and if it is possible to use another corresponding language than English.

Links

Using this element is similar to using in (X)HTML. The value of attributes *href* is URL or *id* another object. The SMIL ads attribute *shows* which can take the value *replace*, *new* and *pause*. If we use attributes *replace* the actual presentation will be stopped and replaced with new ones. The *new* attribute will open presentation in new windows and the original presentation will still be running and with attribute *pause* will be new presentation opened in new windows and the original presentation will be paused. The content of element *a* can be one of objects (*img*, *video*, *audio*...) or synchronizing elements (*par* or *seq*).

```
<a href="http://www.tilt.cz" show="new">
  
</a>
```

The second link is created by element *anchor*, which is almost similar to clicking maps in HTML. Attribute *coords* allocate a rectangle area influenced by the values indicated in the *top*, *left* (top corner) and *bottom*, *right* (bottom corner).

```

  <anchor href="http://www.tilt.cz" show="new" co-
    ords="10, 10, 100, 100" />
</img>
```

4.4.2 SMIL 2.0

As we see it isn't too difficult to learn SMIL 1.0. If we are able to count the times of all objects on 5 fingers then the presentation is complete. However, the differences between the SMIL 1.0 specification and SMIL 2.0 specification are immense. In new specification, they exceed many pages.

SMIL 2.0 ads modules are above all for interactivity (the user can react in presentation – change parameters of objects) and animations (change of color, parameter, move on curve). The main change is modularization – dividing SMIL elements into groups/modules that allowed eke out another format with selected SMIL properties. For example, animating modules are used in SVG. SMIL 2.0 containing 45 modules divided into ten groups:

- *Animation Modules* – modules containing elements and attributes to incorporation animation into documents. SVG takes up some animation modules (with some differences of course)
- *Content Control Modules* – modules containing attributes and elements for content control (construction of decision). These are modules which probe information about the user's system - connection speed, resolution, language etc. It depends on the probe information as it is possible to afford user appropriate content e.g. audio in their preferred language. Part of this module is included in SMIL 1.0 as element *switch*.
- *Layout Modules* – Modules for presentation layout. It contains special elements like *root-layout*, *layout* and *region*. This module doesn't describe only document layout but audio and video volume too. This module describes rather appearance than semantics.

- *Linking Modules* – Modules intended for link creation (from SMIL 1.0 there is still element *anchor* and *a*)
- *Media Object Modules* – modules serve to incorporate the media into documents (*audio, video, img* etc.)
- *Metainformation Modules* – Metainformation modules representing like meta element in (X)HTML and recently *metadata* which works with RDF.
- *Structure Modules* – Structural modules describing the segmentation of document (elements *smil, head, body* and its attributes)
- *Timing and Synchronization Module* – *par, seq* with new elements and attributes
- *Time Manipulations Module* – Modules for changing time action, including attributes for multimedia playback speed - speed up and speed down on original.
- *Transition Effect Module* – module for creation of change over (color change in individual time)

The final result of the application of all modules is very effective multimedia presentation comparability e.g. with PowerPoint.

4.5 SMIL – Making a presentation in practice

In this capture we will attend to multimedia presentation creation with all what comes with it using a practical example. We will use most information from previous chapters.

Before a web designer starts with presentation creation, he should decide which format to use. We pass away PowerPoint and programs such as this and two possibilities remain:

- Create presentation in SMIL. It can be displayed in some multimedia players (Apple Quicktime Player, Real Player and Adobe SVG Player).
- Use form HTML+TIME. Presentation which can be displayed in browsers (Internet Explorer 5.5 and higher) will arise.

Deciding between these formats is a in fact a dull process in that each type of presentation is intended for another device. We can decide again leastways depending on players and browsers extensive. Microsoft Internet Explorer 6.0 is the most extensive browser

today. Against that, the Apple Quicktime Player is the common player on Apple platform and Real Player isn't too extensive (it is like Winamp or Windows Media Player). Next, it is demanding to reveal how many users are able to display a presentation made in Power Point – not many but viewer of files in this format is free (support versions PowerPoint 97 up to 2003).

Finally, many users go ahead to formats HTML+TIME. But if any analogy exists in SMIL the presentation will be readable for many more users. Thus, we try to a make short presentation in SMIL and the same presentation in HTML+TIME too.

Presentation in SMIL and presentation in HTML+TIME aren't different only by end device but there are other differences. Presentation in SMIL must be valid otherwise the player isn't able to play it. This does not apply in HTML+TIME in Microsoft Internet Explorer even if it contains mistakes as well. Other differences are text elements. SMIL works with text like with object so that means that all using text have to be save like **.txt* or **.html* (webpage is for some players unreadable) file. HTML+TIME work with text normally. So behind functional presentation in SMIL skulk more work than in the same presentation in format HTML+TIME. Nevertheless, it is superior and at future more accessible.

4.6 TIME+ HTML

Time Interactive Multimedia Extension for HTML (HTML+TIME) is the outcome of co-operation between Microsoft, Macromedia, Compaq/Dec and Digital Renaissance. HTML+TIME are an expansion of SMIL to common browsers especially to Microsoft Internet Explorer 5 and higher. This format involves time and synchronization into (X)HTML documents.

HTML+TIME 1.0 is based on specification SMIL 1.0. Furthermore, the HTML+TIME 2.0 specification come and is supported in MSIE 5.5. This specification is based on HTML+SMIL language profile according W3C. HTML+TIME 2.0 involved into browser some SMIL 2.0 modules especially Timing and Synchronization Module, Animation Module, Media Object Module and Integration Module.

HTML+TIME is different from SMIL in a few points:

- Some elements and attributes are missing because they are needless – especially as they concern the visual part of SMIL which is replaced with CSS
- Some elements are added
- Integration into code of page is done differently

HTML+TIME successfully integrate SMIL into some browsers and allows for effective use in company networks. In future we can assume that HTML+TIME will be diffused into other products.

4.6.1 HTML+TIME Specification

HTML+TIME eliminate some elements from the original SMIL 1.0 specification because they were found redundant:

- *Smil* – in html replaced by html
- *Layout* – provided by CSS
- *Region* – replaced by CSS
- *Root-layout* – replaced by CSS
- *Text* – not realized as object
- *Anchor* – picture maps are not available

Furthermore, other attributes are replaced (mainly in XHTML) by CSS – *top*, *left*, *height*, *width*, *background-color*, *z-index* and others. New attributes have come up. They assign the behavior of documents (*beginEvent*, *endWith*, *endEvent*, *repeatDur* etc.). Many original (X)HTML elements were amended by SMIL attributes with a view to manipulating them on time line (e.g. *p*, *div*, *span*, *h1-6*, *strong*, *em*, etc.).

At the end were original SMIL elements (no many remain) amended for the sake of *namespace t* (*t:seq*, *t:par*, *t:media*, *t:img*, *t:video*) and its function remain the same. Attributes in HTML+TIME were amended for the sake of *namespace t* (*t:dur*, *t:src*, *t:begin*, *t:end*) but if it isn't used no problems arose, instead conversely in usage some things work badly.

(X)HTML elements attribute

No we look to amount of a new attributes in (X)HTML elements. From originally clean (X)HTML attributes is the most important attribute *id* which allows absolute naming of elements and later call it as follows:

- *beginWith* – value is id of element which have been display in the same time
- *endWith* – value is id of element which have been hidden at the same time
- *beginAfter* – id of element with which have been displayed after the actual element ends
- *beginEvent* – id (form.onSubmit) or name (document.onLoad) of event with demand on it, the next element will be displayed
- *endEvent* – end of demand event
- *repeat* – repeat count of display, value undefine means endless loop
- *repeatDur* – time of repeating (sec), value undefine means endless repeating

If the attribute value is time in seconds it is possible to use decimal number with decimal dot (0.5 = half an second).

Integration HTML+TIME 2.0 into (X)HTML

Document (X)HTML will be necessary to amend in terms of some information like declaration of type of document. Each element which is defined as a block of code with HTML+TIME must have set up time class. Finally we get code like this:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML+SMIL //EN"
    "htmlplusmil.dtd">
<html xmlns:t="urn:schemas-microsoft-com:time">
<?import namespace="t" implementation="#default#time2">
<style>
    .time {behavior: url(#default#time2);}
</style>
<body>
HTML document
    <div class="time">
        HTML+TIME
    </div>
HTML document
</body>
</html>
```

4.6.2 Examples of HTML+TIME for MSIE 5.5 and 6

Blinking texts whether using element blink or using CSS were after a few years, disclaimed with good reason. However, with HTML+TIME we can do blinking text again:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML+SMIL //EN"
    "htmlplusmil.dtd">
<html xmlns:t="urn:schemas-microsoft-com:time">
<head>
    <?import namespace="t" implementation="#default#time2">
    <title>Blinking text</title>
    <style>
        .time, p {behavior: url(#default#time2)}
    </style>
</head>
<body>
    <div style="width: 200px;height: 40px;">
        <t:seq class="time" dur="0.2" repeatDur="1000">
            <p begin="0" dur="0.1">
                Blinking text
            </p>
        <t:seq>
    </div>
</body>
</html>
```

The whole joke is that *t:seq* (sequence with duration 0.2 seconds and repeated for 1000 seconds) contains a block of text which have on time line dedicated section of duration 0.1 seconds. It means that 0.1 seconds the text is visible end next 0.1 seconds is hidden.

HTML+TIME we can apply on VML too using attributes:

```
<v:shape class="time" dur="5s" begin="4s" style='width: 60;
height: 60' stroke="true" strokecolor="black" stro-
keweight="1" fill="true" fillcolor="white" coordorigin="0
0" coordsize="60 60">
```

4.6.3 XHTML+SMIL Profile

With MSIE 6 comes a new profile for SMIL integration into browsers called XHTML+SMIL Profile which integrate subsets of SMIL 2.0 specification. The profile contains support animations, content control, media objects, timing, synchronization and a change over. Some SMIL 2.0 modules are missing but they are replaced in CSS. We have to say that MSIE 6 supports XHTML+SMIL yet somewhat incorrectly. It is much better to manage with HTML+TIME.

Basic head of XHTML+SMIL document:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+SMIL //EN"
"http://www.w3.org/2001/SMIL20/WD/xhtmllplusmil.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
xmlns:smil="http://www.w3.org/2001/SMIL20">
```

4.6.4 The future of SMIL

The world of multimedia presentation looks unnecessarily intricate. This causes the existence of many profiles and pieces of technology. It would be much better to show the development of formats in a few points:

- **1998** – Specification HTML+TIME – implemented with elements SMIL 1.0 into some browsers (MSIE 5) and some elements and attributes are replaced by CSS
- **1998** – Specification SMIL 1.0

- **2000** – HTML+SMIL Language Profile – profile comes out from working draft of SMIL 2.0
- **2000** – HTML+TIME 2.0 – implemented almost all SMIL 2.0 modules
- **2001** – SMIL 2.0
- **2001** – Specification of XHTML+SMIL Profile (Working Draft) – later partly supported in MSIE 6
- **2005** – SMIL 2.1

We can find the language SMIL especially in company networks on the internet as relatively unemployed. Multimedia messages MMS come out from SMIL and it is possible that SMIL will be the domain of mobile devices. SMIL comes out with universal, powerful and not a difficult resolution which is more important for mobile devices than for computers.

4.7 SMIL – animations and presentation effects

SMIL (Synchronized Multimedia Integration Language) allows for creating effective multimedia presentations included video, music, pictures and text. In this chapter we appraise the methods of objects animation in SMIL and with basic presentation elements which will be accommodated in creating complex presentations in the future.

Presentation is instrumental to present information and there is presentation effects used over and over again. Whenever the text seems to fall down, fly in and offset – always it is presentation effect. We can create the same presentation in language SMIL and we can use presentation effect and animations. In previous chapters I described the creation and options of animation in a theoretical manner.

Animation SMIL Module which is integrated in SVG too is a file of a couple types of animations and other effects. What is animation in real? On the web we often find the animated GIF file – animation is expression involve emphasis and movement. So animations can blink, animation can move from any document or jump across the window. Animation possibilities in SMIL can be summed up in a few points:

- Object movement on the curve – a curve is defined by two points and the time delay between them
- Attribute change – change of color, size, location etc. With changing attribute we can achieve e.g. fluent change color from black to white (across shades of gray) or gradual size changing
- Rollover – picture switch after move cursor on – often on web pages
- Picture switching – anime GIF and other elements

Animations are supported even in SMIL 2.0 and their questions are too extensive to the extent that I choose animations in HTML+TIME which are simpler and now more readily available. All source codes are written in HTML+TIME and they can be displayed in Explorer 5.5 and 6.0 without problems.

4.8 Exploitation and resume

All of the mentioned methods are applicable in practice but each of them has their own advantages and disadvantages. The choice of method depends on the evaluating criteria.

JavaScript exploitation appears to be one of the simplest solutions. We needn't install any special software because JavaScript is supported in browsers and its abilities are adequate for simple presentations. However, we hit a few problems. Each browser can interpret JavaScript in its own form (the largest differences are between Internet Explorer and other browsers). We furthermore encounter the possibility of synchronization of streaming video with other elements by using JavaScript necessary video player which can be controlled by JavaScript.

If we appreciate these methods in light of applicability in different browsers and operating systems it will be the best to use Macromedia Flash Media Server because Flash player as they are available to plug-in to almost all browsers. The next advantage is the wide field of miscellaneous options which can be created (interactive animation and according to interaction run video from individual position). The disadvantage of this method is that we have to know Flash environment (environment where are educating materials creating) very well. It is necessary to have Flash Media Serve installed. Because of today CESNET runs streaming server it seems to be purchasing MFS inappropriately. Therefore,

we have to disclaim this method from our point of view. However, if organization is constrained to buy streaming server and has programmers which are good in creating Flash animations then purchasing the FMS is the best choice.

In light of the simplicity of creating and editing educating materials SMIL language is from the mentioned methods the best method. The structure based on XML is very simple to edit presentations and quoted options are usually sufficient. The disadvantage is the less competent extensibility on Macromedia Flash and possible displaying presentation into new window in external application (e.g. Real Player). Audio, video, sequence of pictures, Flash and other formats are possible to combine into the final presentation in SMIL (Synchronized Multimedia Language). If this language is more expanded SMIL then streaming videos will become a very good candidate for other media types.

5 Description of accessible application for work with streaming data

Today, some formats are for streaming use— RealSystem from Real Networks, Windows Media from Microsoft, QuickTime from Apple and then MPEG formats.

5.1 RealNetworks RealSystem

RealNetworks Company is a streaming media initiator on the internet and its products are still pike in this area. Today is the actual version 10.5, radical changes were with coming version G2 (6.0) in 1999. There are three main components of broadcasting string - RealProducer for video encoding, RealServer for distribution and RealPlayer for playback.

5.2 Windows Media

Microsoft Corporation begins with the development of their own solution approximately at the same time as RealNetworks. After the assembly version RealSystem G2 is this technology benighted. Version 7 eliminates almost all defections and products, which become comparable after a long time. The advantage of Microsoft is its price because all components and tools are free just like components of operating system in category Server. The distribution chain consists of Windows Media Encoder, Windows Media Server and Windows Media Player.

The youngest system is Microsoft Streaming Media system. The advantages and disadvantages in one is that Microsoft Streaming Media is in close integration with Windows. Coding software is free to download from the Microsoft WWW server and it is a component of the Microsoft Windows 2000 and 2003 server. We can say the same about the broadcasting server. The management of the broadcasting server is fully integrated into Windows administration (at others system there is WWW interface).

System Windows Media is advisable for broadcasting the content of course (one broadcasting server and many clients).

Coder joins to the DirectDraw/DirectShow interface so it works with many supplying products. The performance of broadcasting computer is at the same with other products. As with the RealNetworks system, it supports the bandwidth negotiation which means

that is possible to send to the server one package which contains many video streams in different bitrates and then the client and server can adjust the quality of transmitting video in dependence on available connection speed. Microsoft Streaming Media unlike Real- Networks differ only in the video part. The audio part of this function doesn't support so it is problematic if we can broadcast the stream of the radio.

As the broadcasting server is integrated into Windows, there isn't a problem with the set up. However, other operating systems users have resin. On the broadcasting server there is one problem. Other operating platforms work so that the coder makes a connection with the server (it is logical because the user is sitting near the coding station and the server, which is somewhere in the hosting center of ISP). At Microsoft Streaming Media the connection serves as the server and there are the problems in organization where is server behind the firewall. In this case the connection is impossible because the transmission can be threatened for the server (firewall) on the organization side.

5.3 Apple QuickTime

Apple Company chooses another business sight of their product as they were dissimulating their server part of string for a long time. Today this format is used for distributing ready records. In Europe this format is not expanded because Mackintosh computers are not augmented in Europe like in the United States.

5.4 MPEG

Formats MPEG-1 and MPEG-2 are for streaming high quality video using relatively for a long time. Due to high data demands they aren't good for transmitting via the Internet. They are used for distributing digital television broadcasting DVB like allowance for classical transmission ways on broadcaster, for distribution and contribution video supply in realtime etc. For internet transmission only MPEG-4 which today undergoes a roaring development is used today.

Result

This work is only a fraction of a thing, which can process and transmit multimedia data. This problematic element is extensive especially for its extensibility and for its possibilities. If the development goes ahead at this rate we can take pleasure in the future of interesting things.

The first chapter is about the possibilities of video compressing where I divided video codecs into two main groups – Lossless and loss codecs. The main direction of my work was to show how many codecs exist and to compare them. I only chose the main ones of all codecs.

In the second chapter I aimed at the main problematic elements and their streaming. It is watching a video via computer network. We can divide Streaming into two parts which are described in the work. Today, a fast internet connection isn't a common affair but places or houses where the provider doesn't offer high connection speeds. In streaming is taking consult to accessibility of stream so I don't aim only to parametric set up video for the sake of the best quality. I mentioned the possibilities of offer video stream in many qualities for wider spectrum of users.

In the next chapter I tried to describe the exploitation of progressive download. Its exploitation is very wide from education to commercial stuff. We can use Streaming like a videoconference.

In the fourth chapter I wrote about the whole work with streaming data on the client side. I mentioned some possibilities of creating online presentations. I mentioned about SMIL language when client can stream not only run but client can create online presentations which can be consist of many streams, texts, pictures and other data.

In the last chapter I added some applications which work with streaming data. This chapter broadens the chapter about streaming.

It was very difficult to choose the most important information from such a huge amount of accessible literature. Especially in the chapter about SMIL I mentioned only the basics for work with multimedia data.

This work should be instrumental to introduce into making online presentations with support multimedia stream. It compares many codecs and their availability. It shows the possibilities of streaming multimedia data and points out how to process this data into online presentation. It is an introduction into creating internet portals for example where students will be able to watch lectures

Resume

The main theme of the work was to show the course of development multimedia and the possibility of working with them. Not even thing that are about processing on computers but also their publishing via computer networks. Another thing was to make instructions for creating multimedia online presentations for students who would be able to teach from anywhere where they have internet connection.

In my work I compare many types of codecs and their exploitation. I mentioned about streaming and possibilities of creating multimedia presentations. That information is for readers to understand this problematic and for ability to make their own online presentation.

Used literature

- [1.] Long, Ben (Schenk, Sonja): 2005. *Velká kniha digitálního videa*. Portál: CP Books. 478 pages. ISBN: 80-251-0580-6
- [2.] Vlček, Karel: 2004. *Kompresa a kódová zabezpečení v multimediálních komunikacích*. Portál: Praha. 258 pages. ISBN: 80-7300-134-9
- [3.] Myslín, Josef: 2005. *Digitální video v praxi a jeho zpracování v prostředí Pinnacle studio*. Portál: Computer Media. 136 pages. ISBN: 80-86686-42-6
- [4.] Dunn, Jason R.: 2003. *Digitální video*. Portál: Computer Press. 264 pages. ISBN: 80-251-0038-3
- [5.] Stolarz, Damien: 2004. *Mastering Internet Video: A Guide to Streaming and On-Demand Video*. Portál: Addison-Wesley Professional. 504 pages. ISBN 978-0321122469
- [6.] Lee, Jack: 2005. *Scalable Continuous Media Streaming Systems: Architecture, Design, Analysis and Implementation*. Portál: Wiley. 394 pages. ISBN 978-0470857540
- [7.] Wootton, Cliff: 2005. *A Practical Guide to Video and Audio Compression: From Sprockets and Rasters to Macro Blocks*. Portál: Focal Press. 800 pages. ISBN 978-0240806303

Used internet sources

- [1] http://www.pvdmedia.com/streaming_ptatese.htm
- [2] <http://www.civ.cvut.cz/info/info.php?&did=443>
- [3] <http://www.civ.cvut.cz/info/info.php?&did=447>
- [4] <http://www.fi.muni.cz/~xpavlov/xml/examples/bc3/bc3.xhtml>
- [5] <http://www.cesnet.cz/doc/techzpravy/2005/streamvideo/>
- [6] <http://gimli.mysteria.cz/komprese/algoritmy.html>
- [7] http://en.wikipedia.org/wiki/Streaming_media
- [8] <http://interval.cz/clanky/smil-zakladni-elementy-a-konstrukce/>
- [9] <http://interval.cz/clanky/smil-jazyk-pro-multimedialni-prezentace/>
- [10] <http://www.mediacollege.com/video/streaming/overview.html>
- [11] <http://www.root.cz/zpravicky/twibright-hyperluma-lepsi-kodovani-vidoa/>
- [12] <http://ronja.twibright.com/hyperluma.php>
- [13] <http://interval.cz/clanky/smil-timed-interactive-multimedia-extensions-for-html/>
- [14] <http://interval.cz/clanky/smil-animace-a-prezentacni-efekty/>
- [15] <http://interval.cz/clanky/smil-tvorba-prezentace-v-praxi/>
- [16] http://www.pvdmedia.com/streaming_technologie.htm

European Polytechnic Institute, Ltd.

Record of bachelor thesis

Author	Jan Válek
Name B.T.	Digital process and transmission of video stream
Educational Branch	Electronic computers
Advocacy year	2007
Number of pages	55
Number of enclosure	0
Leader B.T.	RNDr. David Skoupil
Opponent B.T.	Ing. Vladimír Grohmann
Annotation	Digital video stream review. Possibilities and tutorials for make medial stream and online multimedia presentations and their application into internet.
Key words	Video, stream, codecs, compression, SMIL
Location	Library EPI, Ltd.
Signature	