The Distributed Multimedia Plays Architecture

Technical Report on Futuristic Architecture and Technology[1] (2007, 2009), v3.20 2011

Professor Leif Arne Rønningen, PhD

Norwegian University of Science and Technology, Dept. of Telematics.

O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

Note: This is a live document. When updated, it gets a new version number.

The document is about creation, proposals, philosophy, design and synthesis.

[1] Futuristic - being ahead of the times; innovative or revolutionary

PREFACE

This report is an update of earlier editions of the technical report "The DMP System and Physical Architecture", http://www.item.ntnu.no/~leifarne/The%20DMP%2014Sep07/The%20DMP%20System%20and%20Physical%20Architecture.htm

The work on DMP started in 1997 and has relations back to the introduction of the Traffic Shaping concept introduced by the author by 1980.

From 2002 a number of master's students at Item have contributed to the research on basic problems of DMP. Many thanks to master students Tor E Helgesen, Michal Karpinski, Stig Salater, Erlend Heiberg, Stein O Berg, Dai Kaiyu, Tor S Jenssen, Håvard Berge, Ola Norbryhn, Sindre Grønningen, Håkon Smeplass, Tor A Lye, Hans O Ingeborgrud and Marius H Gundersen.

From 2003 PhD students J Zhang and A. Lie worked on DMP related problems which were included in their final Phd theses. Professor O. M. Aamo, IKT, contributed with knowledge in control theory and queuing. Many thanks for valuable contributions.

PhD student Mauritz Panggabean started in 2009 and has presented some very interesting solutions for DMP. Thank you Mauritz.

Ozgur Tamer was an Ercim post doctor fellow at Item in 2009 and proposed very interesting FPGA solutions for fast processing of video. Thank you Ozgur.

Harald Øverby has lectured DMP related topics as responsible for Item courses, and has lately together with master student Marianne Bøhler proposed business models for DMP services. Thanks to both of you.

More than one hundred master students have carried out a lab assignment studying transient traffic behavior of the Quality Shaping drop mechanism in DMP.

In 2008 and 2010 NTNU AVIT granted all together mNOK 3.6 for development of Collaboration Surfaces, which builds upon the Hems Lab described in Part 5.

Thanks to present head of Item, Poul Heegard for supporting the DMP project, former heads of the department, and to Randi S Flønes and Pål Sæther who has been very helpful in ordering equipment, and to Pål for installations and all technical support.

PART 1: INTRODUCTION TO DMP

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

The concept of Distributed Multimedia Plays, DMP, was introduced as a proposal for an extension to the coming digital TV system, Multimedia Home Platform, MHP, in a Telenor project in 1996-1999 (RON99). At that time, the intention was to enhance the existing and coming TV systems with new features, but this focus has now changed to a long-term project, see section Philosophy of DMP below

THE MAIN FEATURES OF DMP

can be summarized as follows:

The Distributed Multimedia Plays (DMP) three-layer systems architecture provides near-natural virtual networked auto-stereoscopic multi-view video and multi-channel sound collaboration between players, and players and servers. To guarantee the end-to-end time delay less than 10-20 milliseconds, and obtain high network resource utilization, the quality of audio-visual (AV) content is allowed to vary with the traffic in the network. To approach the natural level of human perception, the audiovisual quality has to be increased to levels that temporarily require data rates of the order 10^3-10^5 larger than in existing telepresence systems. The adaptation scheme is called Quality Shaping, and uses Scene Profiles and Quality Shaping Profiles in the shaping process. The Quality Shaping concept includes controlled dropping of sub-objects in the network, controlled dropping of sub-object at access nodes, based on feedback of measured load in network nodes, and prioritized and end-to-end guaranteed delivery sequence of packets. In addition to traffic and scene control, the architecture also supports pertinent security and graceful degradation of quality when nodes or links fail. Parameters included in the quality concept, and that can be controlled *adaptively, are the end-to-end delay, the number of 3D scene sub-objects and their temporal and spatial resolution, adaptive and scalable compression of sub-objects, and the number of spatial views. The integrated scene composition and Quality Shaping scheme uses traffic classes, measurements and forecasting of traffic, feedback control, and traffic and scene quality shaping. The scheme includes scene object behavior analysis, selected packet drop and source scaling. To support Quality Shaping, a new three-layer architecture and the novel AppTraNetLFC protocol has been defined. Due to separate compression of sub-objects, and a packing strategy providing independent parallel audiovisual packets, both high quality- and lower quality multicast users can directly sort out from the large stream the quality they want. To provide DMP services and implement the Quality Shaping concept, Collaboration Spaces (user), Access Nodes with various servers, and Network Nodes have to be installed in well planned (continental, global with reduces delay requirements) network structures The security philosophy of DMP is to give more responsibility to network- and service providers, and reduce the user's ability to make changes in other user's control premises. This concept gives new opportunities for making business with the network itself, in addition to services. Actual DMP applications from arts are jazz sessions, song lessons, and distributed opera. Other applications are in coming generations TV (MHP extended with DMP), games, education, and near-natural virtual meetings.*

Figure 1-1 below illustrates the DMP (a distributed musical play) and some main services/collaboration.

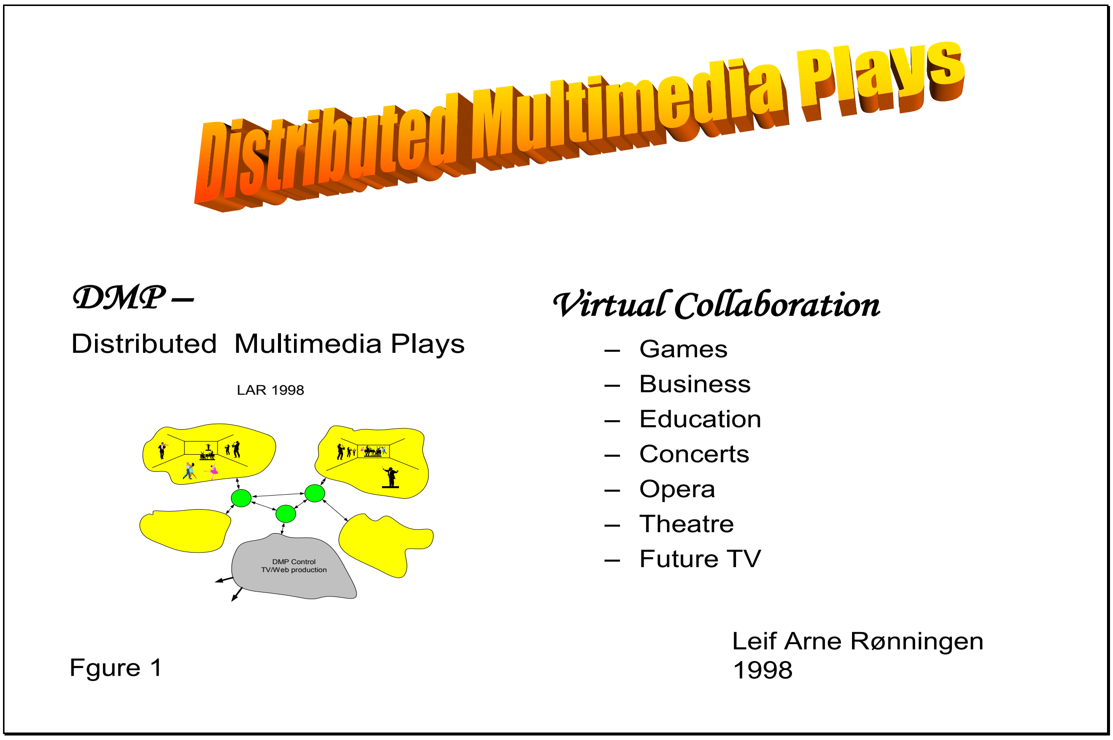


Figure 1-1. Distributed Multimedia Plays

PHILOSOPHY OF DMP

The intention of the DMP Architecture is to present a system architecture that can handle Multimedia Home Space (MHS) distributed services and public and enterprise services using Collaboration Spaces that may be introduced in a fifteen year's time. The history has shown that most people ('experts' and others) do not have the ability to predict the technological development after ten years. We do not know whether DMP will be successful or not, but since the start of the project several fundamental DMP research questions have been answered, e.g., the controlled dropping of sub-objects in the network and regeneration by interpolation and edge correction at receiver have shown successful. Another property of DMP has been validated; the network can be overloaded (by a factor, say higher than 4) and DMP reacts to it in a graceful way, providing services with a lower but controlled, acceptable quality during the overload period.

In this report, it is not the intention to speculate whether the marked wants the services or not, cost aspects are not focused at all, and some technologies needed do not yet exist (and we do not know if they will exist in fifteen years or so). We neglect invested capital, we assume that there will be enough capital to invest in a total new network and DMP equipment. It is not a goal to have this new system working together with existing systems, it is self-sufficient. It is not a goal to build on or reuse standards, we feel free to modify standards for our purpose. But of course, the intention is not to re-invent wheels, we utilize knowledge and research (see references) as much as possible and when it is applicable, and then we add what is needed in a totally free and futuristic manner.

Three main quality goals of DMP are 'near-natural virtual collaboration', 'simple-to-use', and privacy. This implies that the 'service providing system' has to be intelligent and responsible, so it can track and interpret user behavior and talk, and take the right actions to the benefit for users, and relieve the user from complicated configuration procedures and button-pushing.

The following citation from Wikipedia [WIK07a] is also highly relevant:

"**Ubiquitous computing** is a model of computing in which computer functions are integrated into everyday life, often in an invisible way. The model requires both small, inexpensive computers and wired and wireless ("dumb") devices connected to larger computers. A household controlled by ubiquitous computing might have remote-controlled lighting, automated sprinklers, a home entertainment center, devices to monitor the health of occupants, and a refrigerator that warns occupants about stale or spoiled food products.

Modern devices that may serve the ubiquitous computing model include mobile phones, digital audio players, radio-frequency identification tags and interactive whiteboards. Other terms for ubiquitous computing include **pervasive computing**, **calm technology**, **things that think**, **everyware**, and more recently, **pervasive Internet**.[1][2]"

The proponents of ubiquitous computing envision a progression in computing functionality from the primacy of desktop computing, with its focus on programming and publishing, to an age of "natural" computing, wherein computers are accepted and utilized in all aspects of work and leisure. Rapid changes in technology, combined with an increasingly mobile society, ensure that the average person is continually challenged to use unfamiliar electrical and mechanical devices. This requires that devices operate in accordance with the intuition of the user, and serving that intuition requires computing power. Ubiquitous computing is, therefore, (arguably) not a dream in need of pursuit, but a predictable outgrowth of technical solutions to societal trends.

REFERENCES

[RON99] L A Rønningen. 'The Combined Digital Satellite Broadcast and Internet System'. Technical report, Telenor Satellite Services 1999.

[WIK07] Ubiquitous computing. Wikipedia 2007.

PART 2: METHODOLOGY

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

A development approach, where ideas, requirements, and architectural co-design are interleaved and iterated, has been applied. Two phases are used as suggested in Figure 2. The architectural layering philosophy is quite different from the normal, and intends to support efficient hardware design. The approach has something in common with the XP (Extreme Programming) methodology and other methodologies, and can be illustrated by the verses below:

"XP is the evolutionary design approach

Make something that works ASAP,

iterate and refine when requirements pop up

Show them who is the Coach!"

"UML is so beautiful, no one can really fail

And OO is a virtual world, the holy high-tech grail

The phases are so wonderful, they simplify it all,

But iterations, get away, nail them to the wall!

Time has come, and spec shall start,

seems as easy as ringing a bell

This is the portal to hell!"

"System architects, you know best,

20 years of Co',

has shown the way to go

Partition your design,

use hardware for fast and fixed,

use software for fuzzy and mixed

Architects, clever and brave, thump your chest!"

Leif Arne Rønningen, 2006

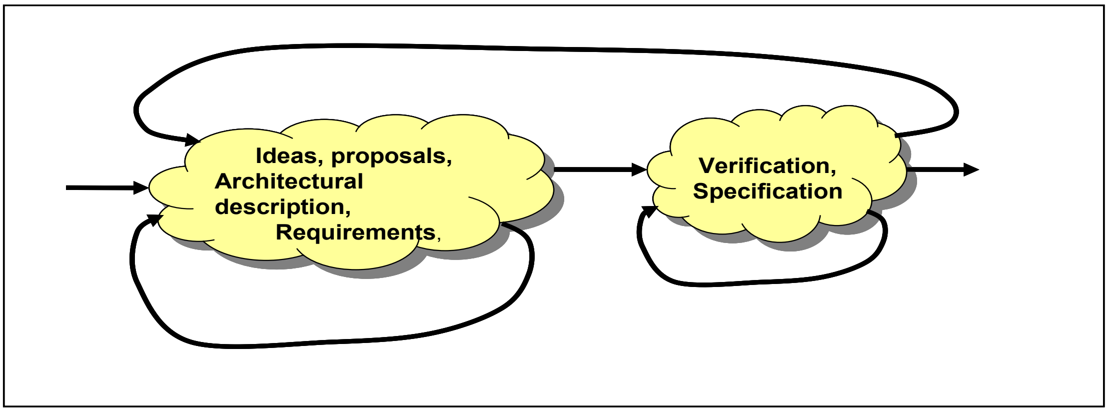


Figure 2-1. Design approach

REFERENCES

XP programming, http://en.wikipedia.org/wiki/Extreme_Programming, 2011

UML, http://en.wikipedia.org/wiki/Unified_Modeling_Language, 2011

HW/SW Co-design, http://www.synopsys.com/Systems/Pages/default.aspx, 2011

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

In fifteen years we expect the quality requirements to become much higher than today. We think about perceived video and audio quality so high that the users shall not perceive any difference between a real collaboration and a virtual collaboration. How near this ideal we can come time will show, but we use the term 'near-natural virtual collaboration' to describe what we expect to obtain. Tests have to be conducted to find the limits. A major problem is that good enough test environments do not exist for our purpose.

Figure 3-1 shows a virtual 'work lunch'. The quality requirements are 'near-natural' (the six boxes below will be described later)

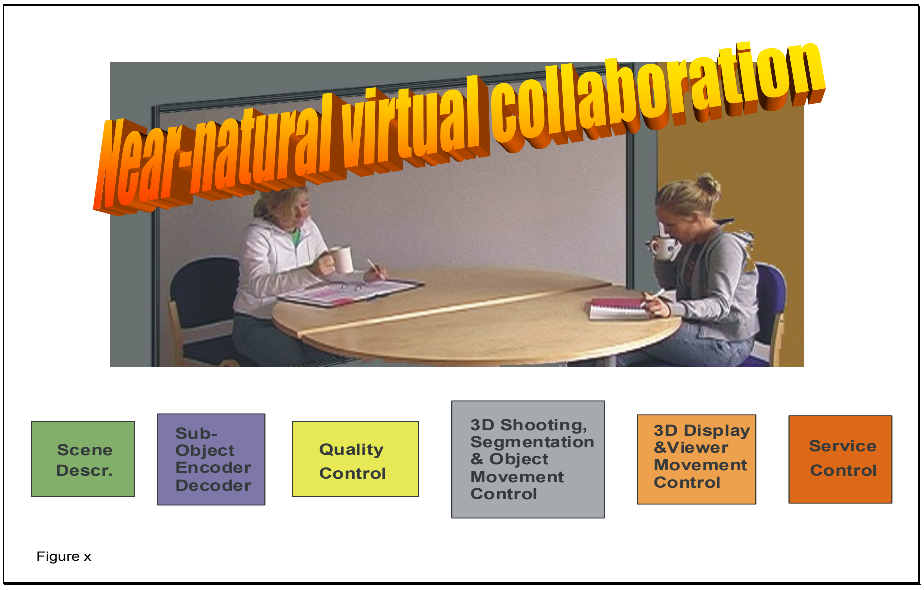


Figure 3-1. A work lunch as DMP

Figure 3-2 indicates that several collaborations require less than 20 ms end-to-end delay. The propagation delay then limits the distance between two users (see Figure 7-1). 'Near-natural' requires an extreme resolution, which means data rates of Gbps, even for a face alone.

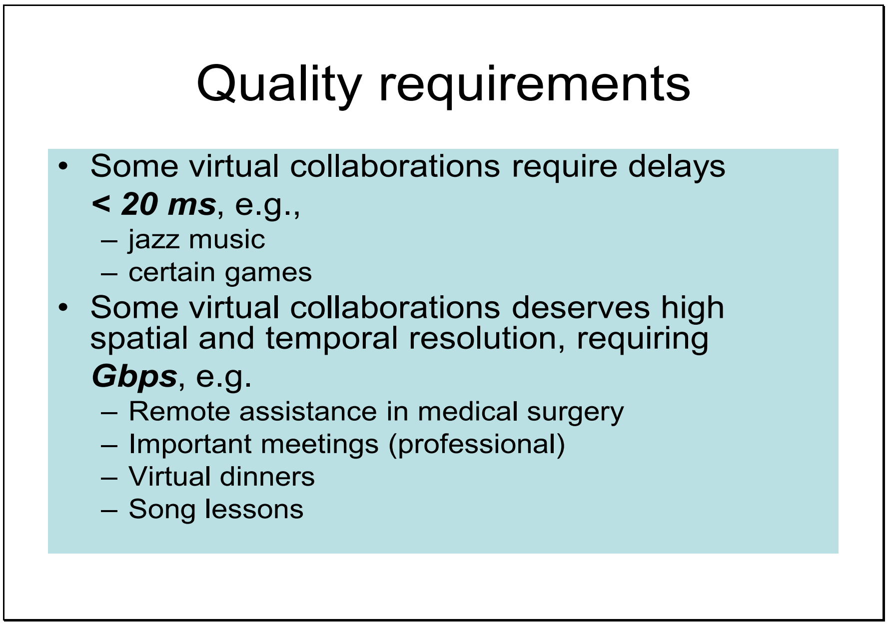


Figure 3-2. Some DMP quality requirements

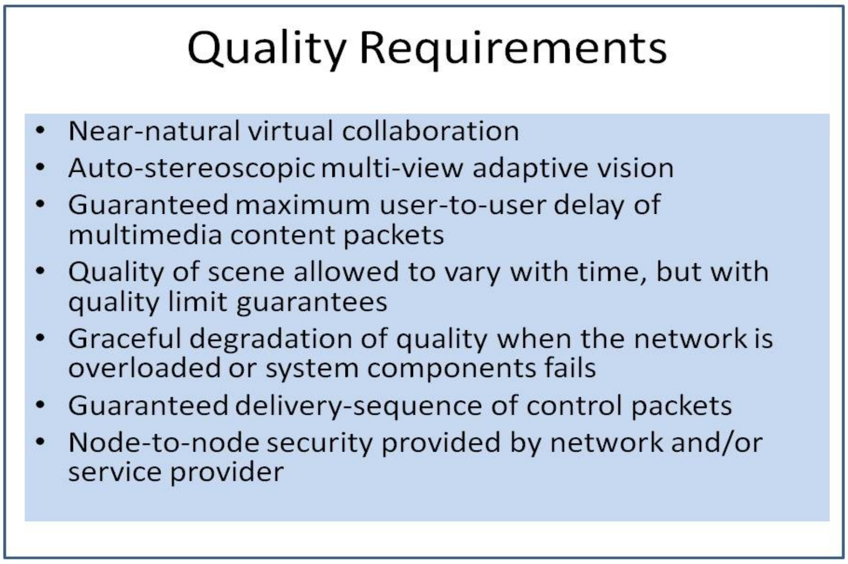


Figure 3-3. Formal DMP quality requirements

By definition, the near-natural virtual scene has a quality that approaches the natural scene, that is, users should not perceive any difference when experiencing a real scene and the corresponding virtual scene. This is expected to be obtained in say 10 years from now. Recent tests [HEI07] indicate that stereoscopic video at HDTV quality (2k x 1k pixels, 60 Hz progressive scan) has substantially lower perceived quality than the corresponding real scenes. A user is defined as a group of humans or other objects in a real scene, or a network server. The scenario, as exemplified in Figure 4-1, is a futuristic, virtual scene that shall support near-natural virtual quality. This prerequisites auto-stereoscopic multi-view, surround multi-channel sound, guaranteed maximum user-to-user time-delay less than 10-30 ms, hierarchic object oriented scenes described by SceneProfiles, scene object quality that varies with time and space, graceful degradation of quality, and a defined security level.

Services are to be understood in a broad sense: the total service (with a well defined adaptive quality) received by users from one or more service providers. SceneProfiles give standardized descriptions of how to shoot and present standardized stereoscopic multiview adaptive scenes. Users negotiate SceneProfiles as the first step of establishing a service.

Traffic generated from near-natural virtual scenes is extremely high, up to $10^3 – 10^5$ times larger than from today's professional videoconferencing systems. This traffic also is extremely variable during the collaboration.

The concept of Quality Shaping was introduced to give graceful degradation of quality when traffic overloads the network or system components fail. The concept builds on controlled dropping of sub-objects (selected packets), and scaling of scene resolution/composition and coding parameters based on feedback. The scheme guarantees a maximum user-to-user delay without any reservation of resources. However, to guarantee a minimum quality level, admission control is needed. Controlled dropping of sub-objects as part of Quality Shaping will be treated in this part. The division of scenes into sub-scenes, objects and finally sub-objects, see Figure 6-1, is of fundamental importance for DMP. This is the basis for making multimedia content packets independent. The DMP architecture synchronizes sources. Since maximum delays can be guaranteed, maximum jitter at the destination can also be guaranteed. For control and management packets only static routing can be permitted, in order not to destroy sequences or loose packets. To guarantee the maximum delay of content packets, routes of lower delay than the maximum permitted can be selected. Basic design goals of DMP are to simplify and to extend the quality compared to existing collaborative systems. Video conferencing systems using standards such as the H.323 need a large number of different protocols to work properly. The aim here is to reduce the number of protocols to two, and correspondingly reduce the number of architectural layers to three. To handle the high data rates, data processing have to be performed by hardware (ASICs). Software solutions shall be used for complex functions without severe real-time requirements.

REFERENCES

M.Gurevich, C. Chafe, G. Leslie, S. Tyan "Simulation of Networked Ensemble Performance with Varying Time Delays: Characterization of Ensemble Accuracy" Proc. Intl. Computer Music Conf., Miami, 2004

PART 4: THE MULTIMEDIA HOME SPACE

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

The Multimedia Home Space, MHS, can be any room in a single family house or an apartment. MHS can be a specialized room, built for near-natural virtual collaboration, or it can be the living room or kitchen. The 'Virtual Dinner' described below typically takes place in the kitchen or dining room, while the 'Interactive Futuristic Movie' probably takes place in the specialized room. From those spaces the users can participate in any networked or local collaboration, private or public, with other people, or with servers.

THE LIVING ROOM

All walls, the ceiling and the floor should ideally be stereoscopic multiview screens, see Figure 4-1. Sofas, chairs, boards and cupboards probably will not disappear. But physical books and book shelves, paintings, photos, curtains, lamps, ornamentation in the ceiling, TV, PC, collaboration equipment, can be provided by the screens. Various user interfaces to services and house equipment (heating, cooling, refrigerator, etc, can be touch-sensitive parts of the wall screens. Or, you can of course use remote controls or just speech commands.

When you want to read a book, you may have a stereoscopic projection of the bookshelf and books, and you touch a book, and your 'virtual-book device' is automatically loaded with the book's content. The virtual-book device can be read exactly as a normal book today, it has physical pages as today, but each page can be loaded with selectable content. Several pages can be spread out on a table. Hyperlinks in the books can be used to fetch more information (like web-pages) and displayed on the surfaces in the room.

The appearance of the room can be changed any time. If you want a big décor in the ceiling, you just load it. If you want green walls and a white ceiling one day, and read walls another day, you can have it. If you want a blue sunny sky, you just project that in the ceiling, and mountains on the walls. Paintings by Rembrandt, Munch, or other can be displayed. New computer generated art works can be viewed in stereo.

Lighting can be handled by the screens. If you like you can display various types of light sources, placed in any position.

The perceived size of the room varies with what you display on the walls or ceiling. If you want a small room, just project (stereoscopic) walls nearer you that the physical walls.

The windows in the house can today give you different views. MHS will give you panoramic views around your house, just by electronically making the whole wall transparent towards the fjord or mountains (like a one-way mirror). This wall can in addition either reflect the heat from the sun on hot days, or absorb the radiation for internal heating on cold days. If the views around your house are boring, you can of course use artificial views.

You may want to set up a virtual visit to your old mother. She can be 'placed' in your (real) sofa just in front of you. You may even have a virtual discussion on the theory of relativity with Albert Einstein (if Einstein is properly modeled as an expert humanoid).

Service providers start selling online varying room appearances: 'On Fridays I want paintings of van Gogh, on Saturdays from 10-12 I want live pictures from the Piazza dei Signori'.

EXAMPLE, A VIRTUAL DINNER SCENARIO

Researcher A in Trondheim enters his dining room, sits on his sofa and requests a Virtual Dinner with researcher B in Padova, also sitting on his sofa. This interaction generates different levels of traffic from A sent to B. The system identifies two faces and a plate with food for researcher B. A and B talk for about 30 s (7 Gbps), then researcher A arises (0.5-1 s), goes out for a plate of food (5-10 s), is out (1-2 min), comes in again (5-10 s), and sits down (0.5-2 s), increasing traffic to nearly 60 Gbps, which then drops to the background of 127 Mbps. The system tracks the plate and the food. They start eating and talking, and the face, arms & hands, and plate & food dominates the data rate, about 8 Gbps. Researcher B stands up and walks sideways out of the room. After a few minutes they need to talk to researcher C in Poznan, and set up a three party DMP. After eating, B leaves the room. A asks the system to disconnect B. Researcher C has to leave home and go to his office. But he wants to continue the session with A while traveling to the office.

THE PLAY ROOM

The Item Experts in Team, Village 24, 2006 [EIT06], with the author as the problem owner and Village professor, presented four proposals for how the Multimedia Home Space (MHS) should look like in 2016. Some requirements were given. The MHS shall be a network based multimedia system, and the perceived quality shall be 'near-natural', or near-real. The four proposals are shortly described below. The requirement to the play room is much the same as of the living room. But in addition, physical simulators, physical 'downhill' and 'uphill', and realistic input devices (haptic) should be provided.

The proposal from group 1 was called VvV (Virtual real Reality, translated from Norwegian, Virtuell virkelig Virkelighet), and assumes a spherical space where advanced AV technology can give users new experiences. The VvV sphere can be applied in many activities, such as interactive multiplayer games, virtual travels, work collaborations, education, remote surgery, and general education. The VvV sphere shall provide stereoscopic multi-view for all viewer positions. The system cannot be realized by existing networks and existing technology. Data rate of Gbps are required between users, and the quality of shooting and presentation equipment has to be improved dramatically compared with existing equipment.

Figure 4-1. All surfaces in the room are stereoscopic multiview projection screens.

Group 2 looked into how to realize the home environment for distributed poker playing. A large, flat screen with lenticular multiview 3D and ultra high resolution is used together with a touch-sensitive 3D screen placed on a table, where poker cards or other can be shown. High-quality 12.1 sound is needed. Present technology is not advanced enough to build such a system, but in ten years it should be possible.

SINDI, Sensual and Interactive almost Natural Dating environment, was the title of the project of group 3. The group ended up with using Virtual Retinal Display or other types of Head Mounted

Displays. Virtual physical contact can be provided by haptic techniques with force feedback, but the present quality is still far from satisfying the group's requirements.

Group 4 tested networked musical collaborations, and found the perceived quality to be varying with genre, degree of improvisations, the ensemble composition, and the player's musical level.

REFERENCES

Documentation of the EIT work is not published (take contact with L A Rønningen to get a copy).

PART 5: THE HEMS LAB AND THE COLLABORATION SURFACE PROJECT

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

SUMMARY

The Hems Lab is a realization of a networked virtual collaboration space, intended for virtual music and musical theatre education, design, rehearsal and performing arts. The quality of the scenes intends to approach 'near-natural quality' in the long term, but to start with, sota[2] equipment has been installed.

The Hems Lab needs a minimum space of height * width * depth = 9 * 12 * 7 meters. In addition, a separate control room of 15 square meters is needed.

The goal is to establish the basic lab, and to run educational and arts performances combining virtual (from digital library and remote collaborators) and live musical theatre drama elements and sequences, including actors/singers playing/singing roles, and scenography, selected from musical theatres/operas such as Thora på Rimol (Borgstrøm) and Pagliacci (Leoncavallo).

Using an *action research* approach, in-depth knowledge will be obtained in an iterative way. Drama sequences and the technical environments can gradually be elaborated to improve the total performance (pedagogical, artistic, psychological, and technical performance).

The budgets for the establishment of the Hems Lab are preliminary and incomplete.

The use of the Hems Lab is not limited to music, and for non-music applications the quality requirements might be substantially reduced.

THE HEMS LAB

[2] state-of-the-art

The Hems Lab is a realization of the DMP architecture, and is part of the ongoing research at the Caruso Lab, Dept. of telematics.

Goal: Extend the Hems lab/DMP test facilities to a 5-surface stereoscopic collaboration space for audio and video, optimizing the use of multi-camera arrays integrated into the screens, multi-projector edge blending and correction problems, projection wall forms, and audio systems when applying DMP in virtual music and musical theatre education, design, rehearsal and performing arts.

Figure 5-1 shows six building blocks. The central block (red top) is the virtual Collaboration Space, CS. Seen from inside the CS, five of six surfaces are combined back-projection displays and multi-camera arrays. This arrangement provides spatially true, multiview, autostereoscopic shooting of video, and single view, stereoscopic presentations (later to be extended to multiview autostereoscopic). To start with, surround sound shooting and presentation shall be provided by head-sets.

The blocks LP (left projection), RP (right projection), TP (top projection), BP (bottom projection) and FP (front projection) encompass arrays of projectors and cameras.

A surface including both camera arrays and projector arrays, are here denoted a collaboration surface.

FP includes at least 12 projectors, with edge blending and warping, and shadow correction for cameras and light-guides for cameras.’ A 6 x 3 meter Clusterwall (Cyviz) provides a passive stereoscopic wall with spatial resolution of about 9 Mpixels.

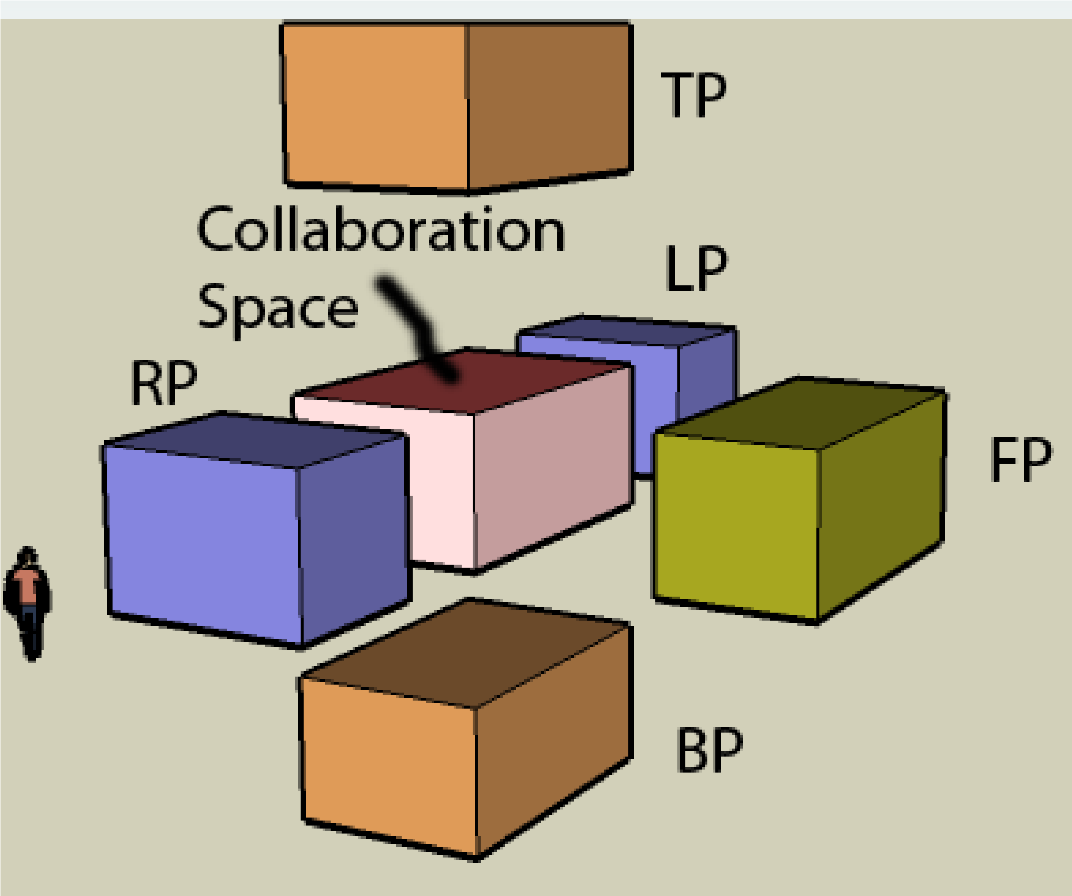
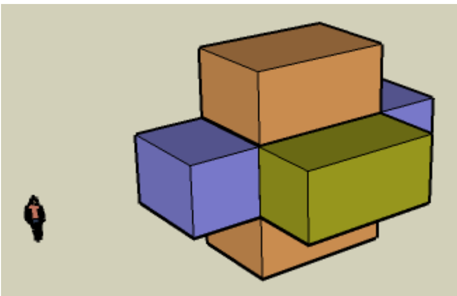


Figure 5-1. The Hems Lab building blocks.

The size of the six blocks assembled is height * width * depth = 9 * 12 * 7 meters. A control room with servers and network nodes should acoustically be isolated from the building blocks.

Figures 5-2 show the building blocks assembled, from two different views.



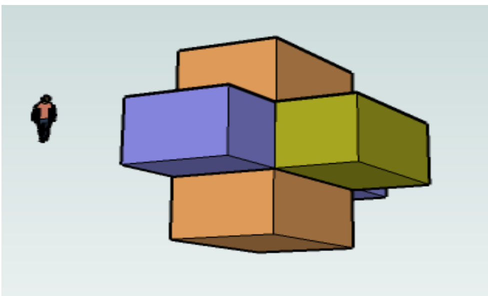


Figure 5-2. The Hems Lab assembled, two views.

Figure 5-3 illustrates a possible case, the Hems Lab is built on the mezzanin under the glass roof of the ‘Elektrobygget’ (the measures are not to scale).

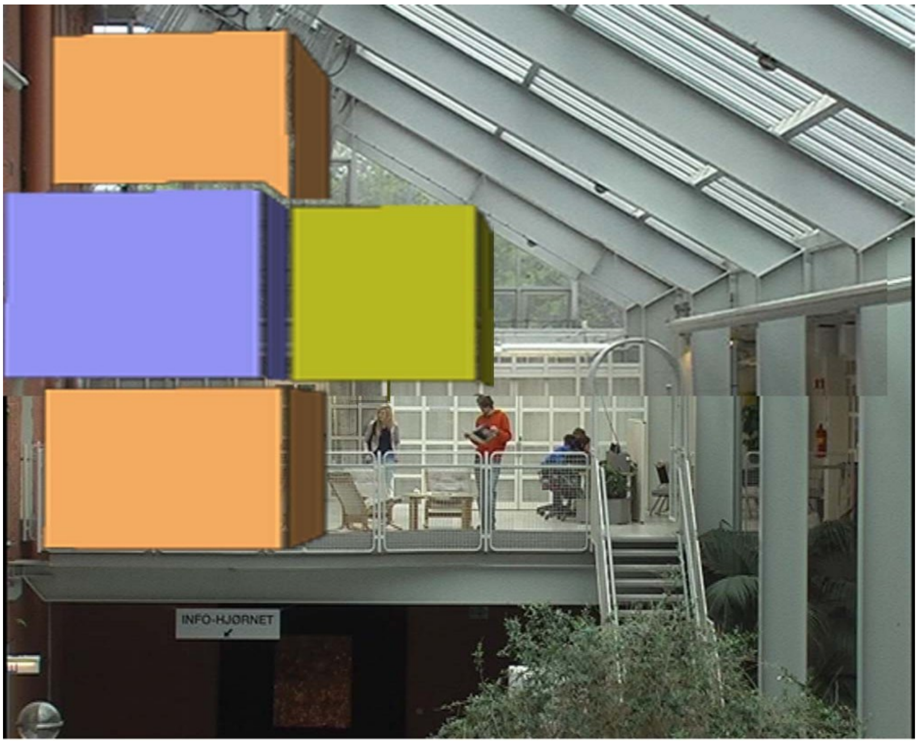


Figure 5-3. The Hems Lab in the ‘Elektrobygget’.

RESEARCH APPROACH

The research challenges of establishing and using the Hems Lab are multidisciplinary, including pedagogy, arts, psychology and technology. Sota theory and practice shall be used as a starting point.

An *action research* approach will be used, to give in-depth knowledge of a few test cases. Action research is chosen because, in an iterative way, the drama sequences and the technical environments can gradually be elaborated to improve the total performance.

In order to generalize, 20 test cases should be run after the total test environment is stabilized. Groups of singers (students and professionals) shall repeat the experiments as described below, selecting different drama sequences. Qualitative and quantitative interviews will be carried out.

Simulation methods will be used to study behavior and traffic performance. The DEMOS and NS2 simulation environments are good simulation tools.

TECHNOLOGICAL RESEARCH

The technological research is in networking, collaboration space design (video-based rendering), video and audio coding/decoding, service management, and Quality Shaping.

The main long-term goal is to approach the near-natural quality of the collaborations.

Simulation will be applied to study the performance of the Quality Shaping scheme and the AppTraNetLinFC protocol. Scene Profiles will be derived from the Hems Lab, and general formats will be studied and specified. Forms for Quality Shaping Profiles shall be specified. Simulation results shall be compared with the results from the action research.

The Hems Lab application requires extremely high audio quality (multi-channel, real surround, 100 kHz sampling, 24 bits resolution) to be successful. This is partly given by the 'near-natural' requirements, but also from the fact that the collaboration may end up as a HD DVD (or similar). Besides, time delays are critical for networked music collaborations (less than 10-20 ms). To start with, head-sets will be applied to obtain the 'position-realistic' surround sound. Later, arrays of microphones, sound processing, and arrays of loudspeakers will be introduced, and integrated in the collaboration spaces.

THE ERCIM POSTDOC TRAINING PROGRAMME

Dr xyz has been awarded an Ercim postdoctor fellowship at NTNU, Dept. of telematics, duration 12 months, starting 1st of January 2008. Xyz's research background strengthens and completes the overall competence needed for establishing DMP test facilities. The fellow's research will be part of the ongoing research activities of the Caruso Lab/Hems Lab, related to establishment of a test system for DMP music applications. The fellow's duties will be restricted to the research program outlined here and he will not be called upon to undertake other tasks. He shall contribute to

- the state-of-the-art description of theory and practice for networked collaborative virtual environments and related systems, and do research beyond in order to approach the near-natural virtual scene quality requirements of DMP in musical theatre applications.

- extending the Hems lab DMP test facilities to a 5-surface stereoscopic collaboration space for audio and video, optimizing the use of multi-camera arrays integrated into the screens, multi-projector edge blending and correction problems, projection wall forms, and audio systems when applying DMP in virtual music and musical theatre education, design, rehearsal and performing arts.
- the study of multi-view techniques for DMP.
- documenting and publishing the research results in globally recognized media.

Some Technical Research Challenges

- Lighting
- Shooting by means of
- multiple micro cameras, integrated into the screens
- Edge blending
- Corrections
- Sound shooting, sound leakage removal
- Realisation of DMP network node
- Object behaviour analysis
- Object movement tracking
- Eye tracking

DMP IN EDUCATION, DESIGN, REHEARSAL AND PERFORMING ARTS

Users of the system are geographically distributed groups of (opera) singers, pop musicians, jazz musicians, chamber musicians, players in large symphonic orchestras, stage directors, scenographers, other arts designers, and the audience. The system enables the various users to experience the virtual environment as seen from their individual viewpoints in real scenes.

Applying the Hems Lab, experiments will be carried out to evaluate system quality performance, and establish user requirements and system quality specifications when applying DMP in virtual music and musical theatre education, design, rehearsal and performing arts. Combined virtual (from digital library and remote user groups) and live musical theatre drama elements and sequences, including actors/singers playing/singing roles, and scenography, will be selected from musical theatre/operas such as Thora på Rimol (Borgstrøm) and Pagliacci (Leoncavallo).

Panoramic video and sound sequences will be planned and shot from the views of the current users. The sequences will be stored in the DMP digital library. The sequences can be modified off-line using editing tools. Tests will then be carried out by streaming the pre-stored contents from the digital library, while groups (one or more participants) of singers/actors play their roles live.

Test groups: 20 groups of 1-5 soloists, professional singers/actors/players and song students.

PRODUCTION FOR LIVE TV AND STORED MEDIA

Several distributed virtual groups play together in real time, using also stored material in digital libraries. The collaboration is composed as one scene, and can be played out live as HDTV, or be stored on media such as HD-DVD, Blueray DVD, or HD Streaming Server.

ARTS RESEARCH

Action research is chosen because, in an iterative way, the drama sequences and the technical environments can be elaborated to improve the total performance (pedagogical, artistic, psychological and technical). Reductive analyses is a tool that can be used to shape and enhance arts performance. The quality of the DMP drama sequences will be measured against the quality of corresponding real sequences. Critics and professional opera singers shall participate as audience, and shall be interviewed about their experiences. The participants perceived adaptable scene quality shall be evaluated. From the experiments, the user requirements and the requirements to the technical system from all participants shall be derived, using in-depth structured interviews and observation of behavior (own and others behavior).

In order to get more experience with applying the DMP architecture in musical theatre education and design, groups of singers (students and professionals), shall repeat the experiments as described above, selecting different drama sequences. To get experience with DMP in rehearsals and performances of musical theatre with audiences in concert halls, sequences stored in the digital library shall be played out and used as the virtual part of the play (actors and scenery), together with a group of live actors/singers and physical scenery.

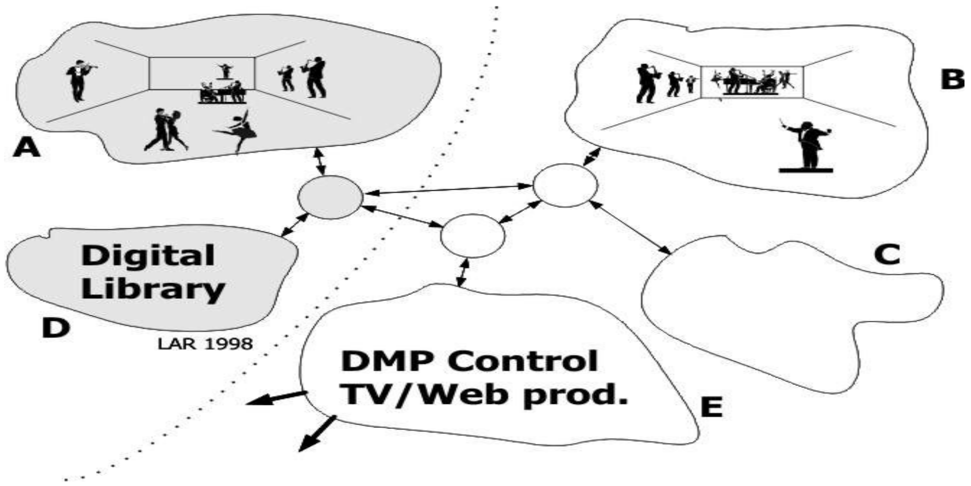


Figure 5-4. Collaborating groups A and B, and Digital Library D

VIDEO EQUIPMENT BUDGET

Visualization and streaming equipment budget, three collaboration surfaces, one site.

(vat not included).

A main 6 * 4 meter Cyviz Clusterwall, with backprojection glass, using 12 F30 projectors, with mechanical framing, costs € 0.33 millions.

Two other collaboration walls (using standard projectors) cost together € 0.13 millions.

Camera equipment cost for all three walls together is € 63,000.

Streaming technologies such as QuickTime on Fast PCs with Raid, or Playstation 3 cluster, € 50,000.

Total equipment cost for the video part of the main front wall and two side walls is € 0.573 million.

PARTICIPANTS, SUPPLIERS

Project leader: Leif Arne Rønningen, Dept. of telematics.

NTNU, Dept. of telematics (DMP, musical theatre). L A Rønningen, Erlend Heiberg (master student), two PhD students (Verdikt, EU FP-7 ICT), xyz (Ercim postdoc), master students, EiT – Experts in team.

NTNU, Dept of electronics and telecommunications (audio), Jan Tro, master students.

NTNU, Dept. of music (musical theatre, song, jazz, etc), Kåre Bjørkøy, Carl Haakon Waadeland, song students, music students.

NTNU, Music technology (audio)

UiB, The Grieg Accademy (musical theatre, song), Harald Bjørkøy, song students.

Q2S (audio), Peter Svensson, PhD students, master students

Cyviz (developer, video walls)

Tandberg (developer, videoconferencing)

RELATED PROJECTS, THEORY AND PRACTICE

The research on future generations networked multimedia systems to be carried out in Hems Lab shall be based on the state of the art research and design of digital scenography (such as Intermedia, UiO [MOR05] , University of Århus [CAV07] , Telematics, NTNU [RON03]), the participant's extensive experience from staging and acting in opera performances [RON06], the ongoing work on DMP (including simulations and performance evaluation), the research on digital libraries performed by Ercim [SØL06], action research (learning) methodology [DIC99] and experience (Experts in Team) [EIT06]. The state of the art of multimedia streaming systems (such as QuickTime), is presently far from satisfying the severe requirements of future DMP, and shall be extended.

The Thora opera was staged using a combination of live actors/singers on a physical stage (proscenium) and projected video showing both scenic elements and actors. The participants (NTNU, UiB) were heavily involved in the staging and performance of Thora [RON03].

At the University of Sydney, Australia, researchers have investigated the role of the '3D virtual place', its effect on the activities, discourse and learning of students using a 3D virtual learning environment that encourages collaboration and constructivism. The study shows that locating students and facilitator (as avatars) is important for identity, presence, discourse, and learning [CLA06].

Immersive 3D videoconferences provide immersive tele-presence and natural representation of all participants in a shared virtual meeting space to enhance quality of human-centered communication [KAU02].

Telepresence refers to a set of technologies which allow a person to feel as if they were present, to give the appearance that they were present, or to have an effect, at a location other than their true location [WIK07].

A Virtual learning environment (VLE) is a management system for educational courses. An example is it's learning [ITS07].

A Virtual Collaborative Environment (VCE) is one that actively supports human-human communication in addition to human-machine communication and which uses a Virtual Environment as the user interface [VCE07].

Spatial faithfulness [NGU07] simply means that the viewer should see no difference between a virtual scene and a real scene. This complies with the near-natural concept of DMP, see below. The position of cameras is critical. The cameras should ideally be placed behind the object of a scene a viewer is focusing on. This can be obtained by using a semi-transparent mirror, moderator. A moderator was used in a two-way video conference between the University Colleges of Lillehammer and Gjøvik, Norway. The total system evaluation shows that the time delay is critical, the overall quality was too low, and the naturalness was not satisfactory [VOL06].

The multi-camera array project at the Stanford University has contributed significantly to the knowledge of video-based rendering [WIL04].

The textbook, Video-based rendering, by Marcus A. Magnor, gives an excellent introduction to camera- and projector arrays, and to image- and video-based modeling and rendering [MAG05].

Tandberg is a leading vendor of videoconferencing systems worldwide. Their products represent the state of the art [TAN07].

The company Cyviz designs and markets advanced products for visualization. Their Clusterwall provides excellent stereo in ultra high resolution, e.g, 9 Mpixels on a 2 x 1.5 meter screen, in a configuration using 12 true state of the art F30 projectors with Intensity Transfer blending. The intention is to use a Clusterwall configuration for the Hems Lab [CYV07].

EXAMPLES OF SIMPLIFIED IMPLEMENTATIONS

Applications using simplified versions of Hems Lab were implemented by Experts in Team villages in 2008, 2009, and 2010. Contact the author for more information.

OTHER MULTIMEDIA SPACES - STATE OF THE ART VISUALIZATION SYSTEMS

The Cave and Rave 3D visualization systems were developed by the University of Illinois (1992), and later commercialized by Fakespace Systems, now a Mechdyne company. The Extended C6 visualization system delivered by Fakespace to Iowa State University's Virtual Reality Applications Center, represents the present state of the art. The system gives 100 Mpixels presentations, using 96 graphical processors from HP and 24 projectors from Sony [IOV07].

See also ***3D Globe and 3D Room*** [3DR07] and ***Second Life*** [SEC07].

THE COLLABORATION SURFACE PROJECT

The project is funded by NTNU/NFR through the AVIT program. In this project the plan is to develop and test the quality of a modular, quality-enhanced (beyond state-of-the art) Collaboration Space, Hems Lab 3.0. This includes carrying out experiments to evaluate the quality of experience and establish user requirements and system quality specifications when applying the Collaboration Surface in virtual music and musical theatre. Moreover, the Collaboration Surface will be applied for new gaming concepts, to obtain real feel by using stereoscopy in serious games, and to enhance the satisfaction of First Person Shooter (FPS) games. The following task shall be carried out:

- Build and evaluate a wall-size auto-stereoscopic continuous-view display (own proposal) based on:
  - Continuous-view convex lens array screen (in front)
  - Pixel-integrating concave lens array screen (behind)
  - Pico projectors (back projection)
  - FPGA-based distributed processing and play-out system
- To be able to build large Collaboration Spaces with continuous displays of any size using smaller displays and camera arrays, the joints between the displays can be covered by combined stripes of auto-stereoscopic continuous-view displays, camera arrays and audio equipment.
- Test, calibrate, configure and improve a newly designed Camera Cluster Array, which is capable of shooting 3-9 (more than RGB) spectral bands, in order to support the design of enhanced concepts for object segmentation and object (eye) motion tracking, and 'ultra definition' in time and space.
- Apply a newly designed processing array of FPGAs with 3-dimensional systolic arrays to implement processing of the output of Camera Cluster Array. Develop software and hardware by co-design, real-time embedded programming, and methods for allocation of functional processes to processing units.

The joints between displays shall be used for camera arrays and audio equipment. The intention is to make the joints nearly invisible and the whole image continuous by introducing a combined camera array and auto-stereoscopic continuous-view stripe. Such stripes also integrate microphones and loudspeakers. To start with, headsets will be applied to obtain the 'position-realistic' surround. Later, arrays of microphones, sound processing, and arrays of loudspeakers will be introduced.

In addition to object segmentation and motion tracking, various methods shall be applied to correct for small lens aberration and imaging chip artefacts, and for stitching of images. Calibration schemes for the collaboration surfaces shall be proposed and evaluated.

The software and hardware architectures for Camera Cluster Array and display processing involve research on embedded hw/sw co-design, real-time, heterogeneous multi-programming, systolic arrays, FPGA realisation, and methods for allocation of functional processes to processing units. As can be seen, this development involves several research areas and extensive evaluation.

The concept of SceneProfile [RON07a], which defines the possibilities and limitations of DMP collaboration spaces, shall be further elaborated. Eye tracking and object focus (receiver) shall be implemented to decide the number of views and the resolution of the focused objects (sender).

The results from quality evaluation and design of new concepts and research can be published at Electronic Imaging, ACM Multimedia, IEEE, and other well-known conferences.

VIRTUAL MUSIC THEATRE RESEARCH

Experiments shall be carried out to evaluate the system quality performance and establish user requirements and system quality specifications when applying The Hems Lab in virtual music and musical theatre education, design, rehearsal and performing arts. For virtual song rehearsal, it is desirable to have the score/notes presented on the screen in front of the singer.

HEMS LAB GAME APPLICATIONS

The Hems Lab will be highly valuable for gaming research. The research area encompasses development of new gaming concepts, how to use stereoscopy to enhance the satisfaction of First Person Shooter (FPS) games, and to use stereoscopy in serious games. The Hems Lab can also be used to recreate the reality as true as possible. This can be modeled by stereoscopy to obtain a synthesized world where the gamer can move freely. Alternatively, one can realize this by shooting a number of 360-degree images at different points and letting the user jump between the points. This technique will not provide the same degree of freedom as synthesized 3D, but will give a much more realistic picture of the surroundings since the pictures are shot by a high-quality camera. In this way, one can experience Rome (or the moon) in the Hems Lab!

The equipment shall extend the existing Hems Lab versions 1.0 and 2.0 as described above. The rooms A264/268 in the 'Elektrobygget' give sufficient space also for a small Hems Lab version 3.0.

The new Collaboration Surface shall be built around a new concept of using two lens arrays and pico projectors. The lens arrays will be developed in cooperation with Inst. for produktutvikling og materialer, NTNU. The display hardware shall be PC-cards with FPGA onboard.

The newly designed Camera Cluster Array shall be extended to handle 5 to 9 sub-colors, and camera arrays that shall be integrated with display lenses as stripes on the joints between display modules. The image processing shall be performed by FPGA modules, configured in a 3-dimensional processing architecture.

To store audiovisual content, SATA 3.0 hard disks configured as RAID shall be applied.

The audio arrays shall be integrated with the joint stripes.

Mechanics for the displays/Collaboration Surfaces and general purpose PCs will be needed.

TELEPRESENCE, 3D TV AND VIDEO-BASED RENDERING

Sisco/Tandberg [TAN10] and others produce state-of-the-art videoconferencing systems denoted telepresence systems, using protocols such as TiP (RTP, RTCP), SIP and H.323 over UDP/IP. The EU research project, 3DPresence, [3DP10], estimates the end-to-end time delay is about 400 ms.

The research of DMP builds upon experience from telepresence, state-of-the-art research of networking, digital video and computer vision, such as 3D-TV and Video (Image)-Based Rendering.

[KAU07] gives an excellent overview of the history of 3D TV. [MAT04] demonstrates a 3D TV system architecture, using a multiview camera array and autostereo multiview lenticular-based projection. [KAU07] further shows a generalised 3D TV system based on N x Video + Depth streams after images are captured. The following rendering steps are performed: image rectification, disparity matching, depth map creation, and de-rectification. While most works assume standard cameras and processing on standard images, other researchers have proposed introducing infrared light to support segmentation of scene objects [DAV98], and chroma keying using invisible light [BEN02].

RESEARCH METHODOLOGY

The main characteristics of the action research methodology as applied in this project are the iterations between Design/Creation and Research, and iterations within Design/Creation and within Research. To evaluate scene quality, both subjective (perceptional) and objective measurements are used. The subjective tests are based on comparing the virtual quality of the technical systems with the real world scene quality.

REFERENCES

[3DR07] Virtual Reality, Carnegie Mellon University 2007. http://www.cs.cmu.edu/~virtualized-reality/

[CAV07] The Centre for Advanced Visualization and Interaction, CAVI. Aarhus University. http://www.cavi.dk/

[CLA06] Clark, S and Maher, M.L. COLLABORATIVE LEARNING IN A 3D VIRTUAL PLACE. Investigating the Role of Place in a Virtual Learning Environment. The University of Sydney, Australia 2006.

[CYV07] Cyviz. http://www.cyviz.com/, 2011

[DIC99] Dick, B. Sources of rigour in action research: addressing the issues of trustworthiness and credibility. the Association for Qualitative Research Conference "Issues of rigour in qualitative research" at the Duxton Hotel, Melbourne, Victoria, 6-10 July 1999. http://www.scu.edu.au/schools/gcm/ar/arhome.html

[EIT06] Course notes, EiT, NTNU 2006.

[IOW07] Iowa State University's Virtual Reality Applications http://www.iastate.edu/%7Enscentral/news/2007/mar/C6.shtml http://www.fakespacesystems.com/

[ITS07] Virtual Learning Environment. it's learning , http://www.itsolutions.no/imaker.exe?id=111

[KAU02] Kauff, Peter., Schreer, Oliver. An immersive 3D video-conferencing System using shared virtual team user Environments. The ACM Digital Library.

http://portal.acm.org/citation.cfm?id=571895&coll=portal&dl=ACM

[MAG05] Magnor, M. Video-Based Rendering. A. K. Peters 2005.

[MOR05] Morrison, A. Unsettling the setting? Reflections on digital scenography. Invited presentation at Research Seminar on Digital Scenography. 24 January 2005. CAVI, Aarhus University, Denmark. http://www.intermedia.uio.no/people/home/andrewm/andrewm

[NGU07] Nguyen, D, and Canny, J. MultiView: Improving Trust Formation through Spatial Faithfulness, Proceedings of the 2007 ACM Conference on Human Factors in Computing Systems.

[QUI07] QuickTime Streaming Server. http://www.apple.com/quicktime/streamingserver/

[RON03] Rønningen, L A. Video scenography for Thora. The seminar, The education of singers – important for regional opera?, at Griegakademiet, UiB 12 feb 2003

[RON06] CV for Leif Arne Rønningen, http://www.item.ntnu.no/~leifarne

[SEC07] Second Life, 3D Online Digital World. http://secondlife.com/

[SØL06] Sølvberg, Ingeborg, IDI, NTNU, Thanos, Costantino, ISTI-CNR. Towards the European Digital Library – introduction. Ercim News 66, July 2006. http://www.ercim.org/publication/Ercim_News/enw66/Ercim

[TAN07] Tandberg video conferencing systems. http://www.tandberg.com/

[TEL07] Telepresence http://en.wikipedia.org/wiki/Telepresence

[VCE00] Virtual Collaborative Environments. VCE http://www.crg.cs.nott.ac.uk/groups/cve/

[VOL06] Vold, E. Open Channel. 'Utterance Rights' and Distributed Plays. Masters' Thesis NTNU 2006.

[WIL04] Wilburn, B. High Performance Imaging Using Arrays of Inexpensive Cameras. PhD Dissertation, Standford University 2004.

PART 6: OBJECT ORIENTED SCENES

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

Scenes are composed of real and virtual (stored or generated) sub-scenes. Sub-scenes are composed of hierarchies of sub-scenes. The smallest real entities are called objects. Examples are a human being, a football, or a background. After being shot by a camera array, objects can be further subdivided into sub-objects [LIE03]. The number of sub-objects per object could be 4, 9, 25, or other. Figure 11 shows an example where object 1 is divided into 4 sub-objects, and moves from one position in space to another with a certain speed. The sub-objects can be coded as independent streams and independent (AppTraNet) packets within each stream. This possibility is the 'cornerstone' of DMP. It supports graceful degradation of quality when traffic overloads the network, or when network components fail.

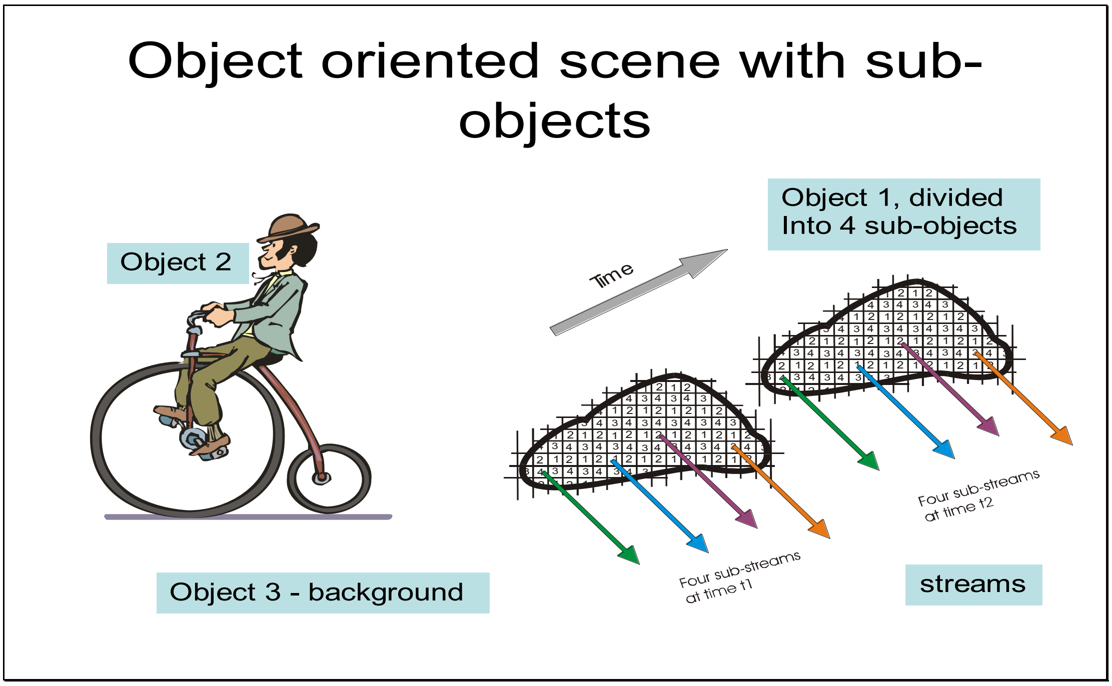


Figure 6-1. Object oriented scenes with sub-objects.

REFERENCES

[LIE03] Lie, A, Rønningen, L A. Distributed Multimedia Plays with QoS guarantees over IP, Proc. of IEEE Wedelmusic '03, 14-17 Sep, Leeds UK, 2003

PART 7 - DMP NETWORK TOPOLOGIES

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

DMP networks are hierarchical, built with combinations of star and mesh topologies. It is not the aim to provide the optimal network topology. Partly because we do not know what the traffic interests will be in 15 years, partly because it is not the focus of this report. A number of possible topologies are suggested, given the population densities and distances between parts of Europe.

The DMP network uses fixed routes as the basic rule. This is necessary to guarantee delays. However, to provide alternatives when a processing part or a link goes down, loadsharing of links between nodes, and hot standby of the processing section for each link, shall be used. Note that the use of sub-objects and independent packets already provides a graceful degradation of quality, not only when packets are dropped, but also when packets are lost because of high BER, and when packets are lost because of parts of the network go down.

Alternate routes (via different nodes) give the same number of hops (maximum delay may vary only because of different propagation times). The traffic is equally shared between alternate routes. This is a must for the QualityControl because each node reports its traffic load independently.

However, in some cases, when the traffic interests between for example two cities of two different European regions are high, a shorter path can be used, and shall be used until it is saturated. Then the main route takes over.

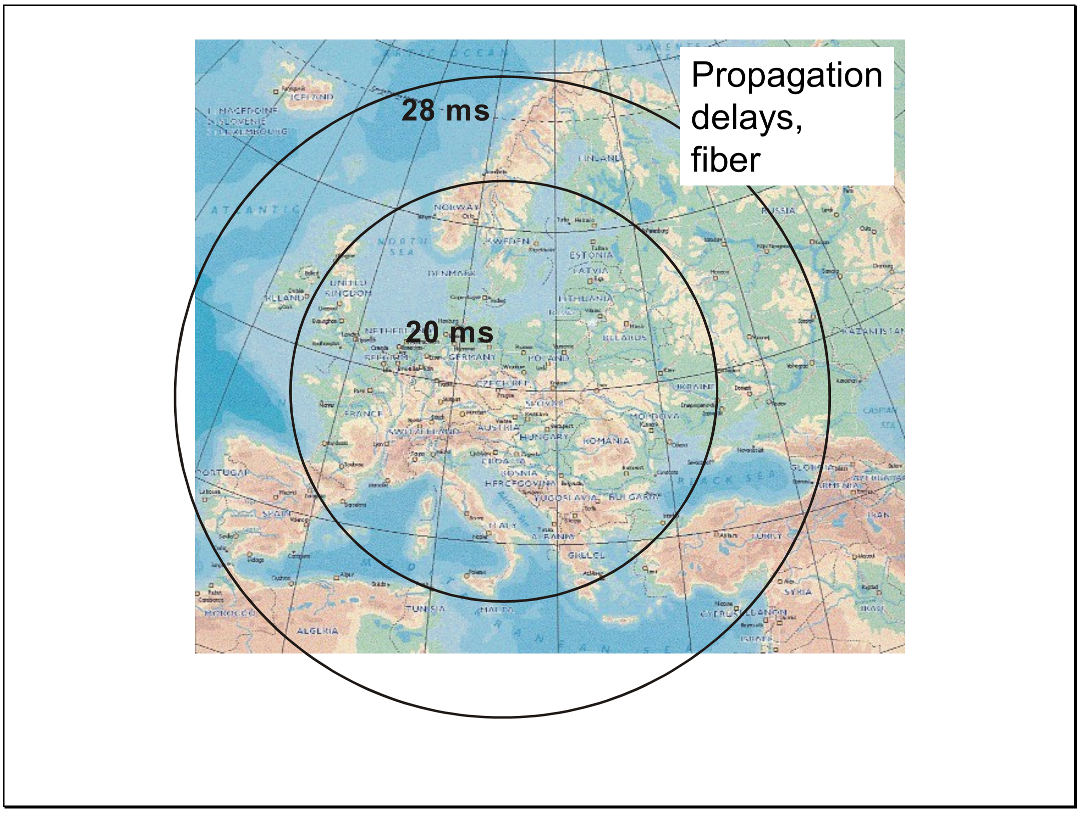


Figure 7-1. Propagation delays in Europe.

To provide a DMP service to all inhabitants in Europe, a hierarchical topological and addressing structure is proposed, with the following node levels.

Node		*IPv6 address,*	*bits*
		(Format prefix)	3
• GlobalNode,	**G**	(TLA)	13
• EuropeNode,	**E**	(reserved)	8
• DistrictNode,	**D**	(NLA1)	12
• CityNode,	**C**	(NLA2)	12
• VillageNode,	**V**	(SLA)	16
• AccessNode,	**F**		16
• AccessLink,	**FL**		48

The GlobalNode routes traffic to/from areas outside Europe, while a EuropeNode handles one European region, and traffic to/from other European regions.

Within Europe, the maximum number of hops between two users is 11, but depending on traffic interests, shorter routes can be defined as suggested in the figures below. The number of IO links from each router in the figures is 128, but could be 256 or 512. The allocation of links to neighboring nodes depends on traffic interests, which are presently unknown. Three example structures up to European-, District-, and City level respectively, are shown.

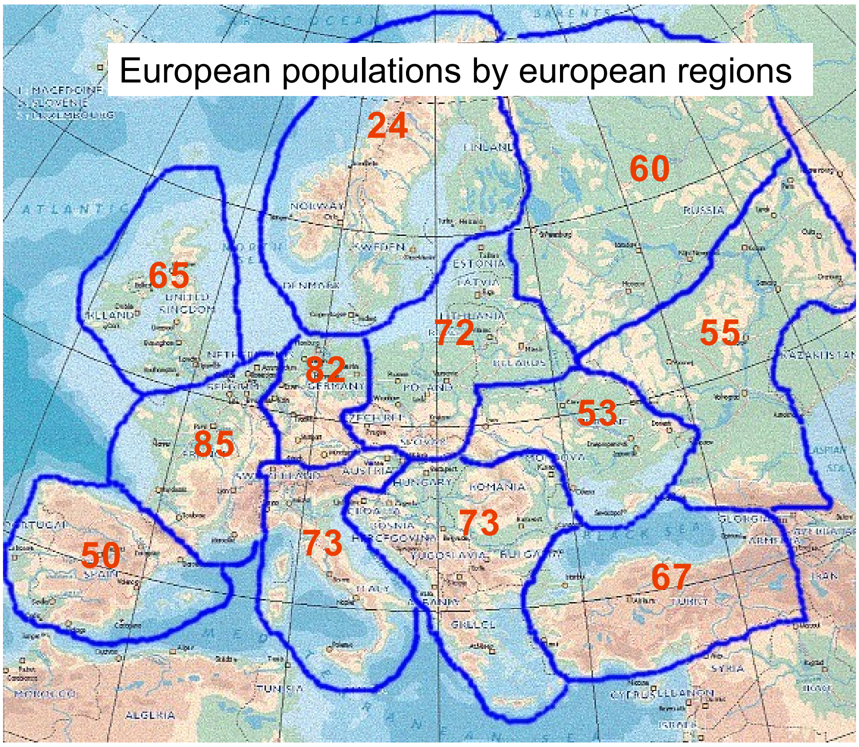


Figure 7-2. European regional populations.

The population of Europe plus Turkey is less than 1000 millions. If we assume 512 users per access node, roughly 2 million AccessNodes will give at least one access link for each person.

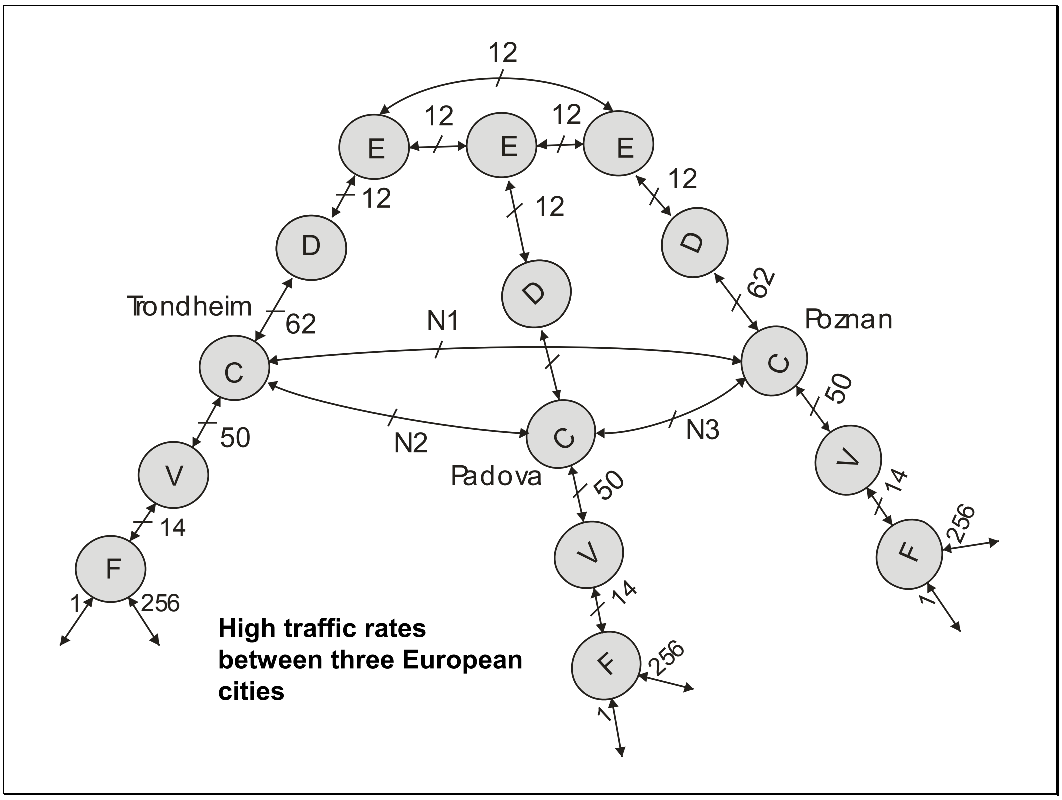


Figure 7-3. Network between three European cities.

Some assumptions can be made about traffic interests in the future, to see what the consequences would be. More international cooperation than today is expected, it might be that certain cities in different countries in Europe, for one or other reason, cooperate much more than cities in one country. Such a scenario is dipicted in Figure 7-3, where the cities Trondheim (Norge), Poznan (Rzeczpospolita Polska) and Padova (Italia) have a huge need for DMP interactions. The DMP topology solves this by letting direct high-capacity links be added between the cities. The consequence for Quality Shaping is considered in another Part.

The largest cities (or highly concentrated areas) in Europe have less citizens than about 12 million. Few cities are larger than 3 millions. Network topologies for those two maximum sizes are suggested in Figures 7-4 and 7-5 (this is not base on a formal optimization, it is an intuitive design based on assumptions on traffic interests).

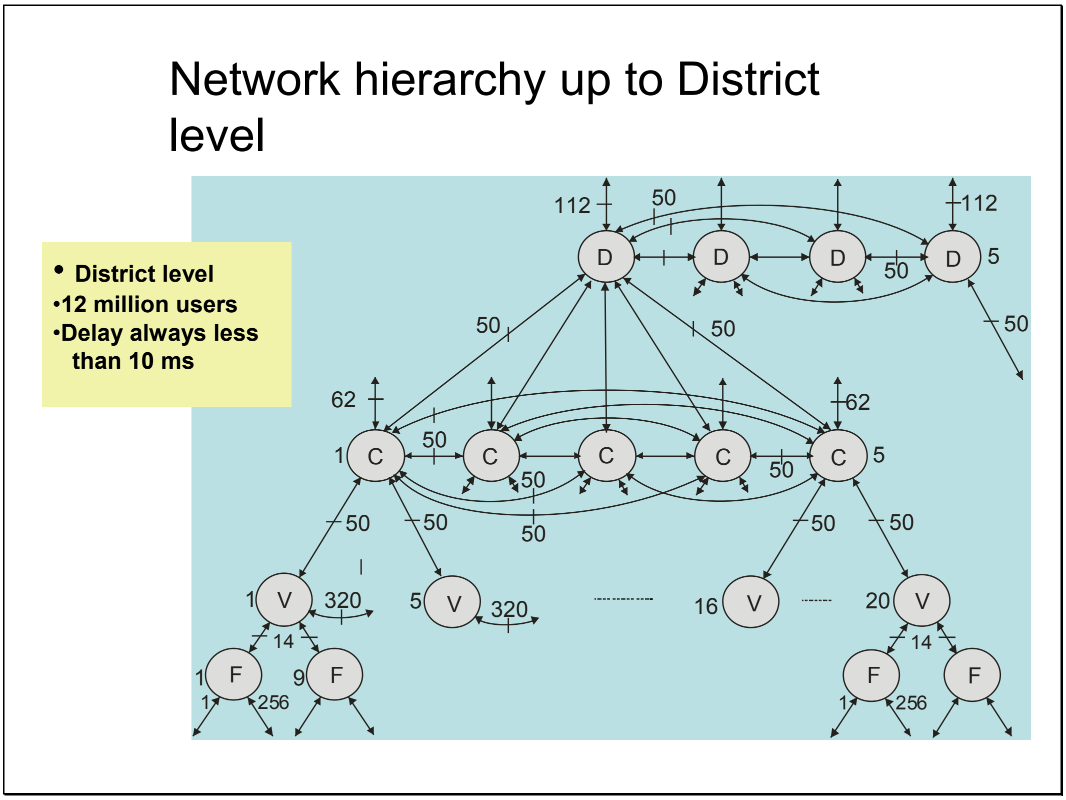


Figure 7-4. A possible hierarchy up to district level.

The hierarchies shown assume that both multimedia content traffic and signaling traffic are carried by the network. It is assumed that all servers handling network maintenance and operational aspects are distributed over all nodes. It is also in another Part assumed that all servers handling Customer Databases, Scene Profiles, Quality Shaping Profiles and the like are distributed over Access Nodes. But letting those servers handle only up to 512 users may not be economical (hm, we said in the introduction that we didn't care about that) even if it maximizes availability. The next step would be to place all servers at the next level, at Village Nodes. Then each server would be responsible for say around 2000 users.

A possible philosophy could be

- Special 'unbalanced' traffic interests are solved with directs links (as in Figure 7-3)
- Normal traffic interests are handled by structures as described in this Part. Overload of central network parts are either 'synthetic' (not originated from normal users) or intentionally injected (by unfriendly groups)
- Central customer databases
  - Collect data from distributed servers
  - Should not be updated in real-time, only in periods of low traffic
- Distributed customer databases and servers
  - Handle real-time updates
  - Overloads affects only a few servers and several thousands of users

- The overloaded local network reduces overload by 'back-pressure' flow control, the originating signal traffic sources are automatic throttled, while established services can finalize their session

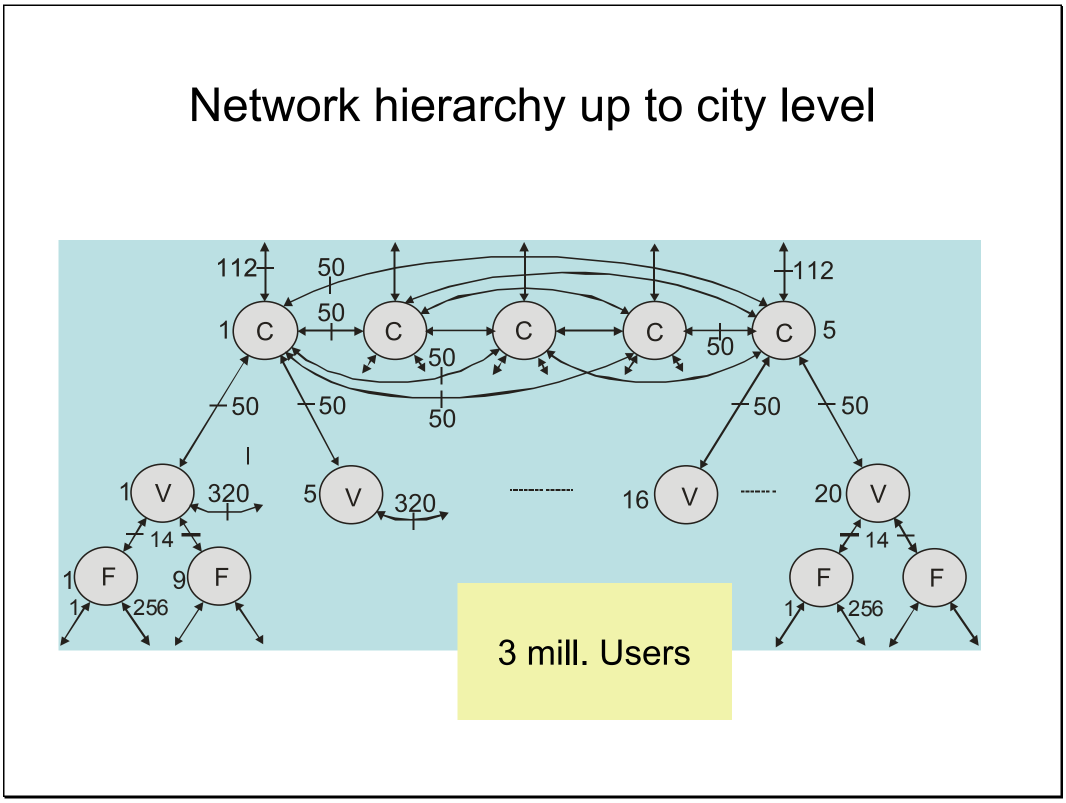


Figure 7-5. A possible hierarchy up to city level.

The optimum topology for the European network is not considered here.

The traffic interests in the future are most uncertain. Who will communicate with whom? Will a person in Padova communicate much with a person in Poznan?

PART 8: THE THREE-LAYER DMP ARCHITECTURE WITH THE *APPTRANETLFC/PCI EXPRESS* PROTOCOLS

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007, 2009), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

In this section, the new AppTraNetLFC protocol, which is an extension of the AppTraNet protocol from 2007, is introduced together with a short description of the PCIe protocol used in Posted Transaction mode. To understand the protocol, descriptions of behavior that involve user equipment and network nodes simultaneously are required. There are two different node types in the network, AccessNodes and (Core) NetworkNodes. In addition, a number of specialized servers are needed, for example to support collaboration establishment and management. The user premises are denoted Collaboration Spaces.

ARCHITECTURE AND THE APPTRANETLFC/PCIE PROTOCOLS

Audio-visual (AV) packet delays through nodes and processing delays in user equipment can be guaranteed lower than a specified value. Except for output link queues in nodes, there is negligible waiting. Information is included in packets so that an AV packet is independent of all other packets. The content of a packet is used to present parts of an object immediately, at the right place and with the right quality (variable, however), not waiting for other parts of the object or other objects. Pre-stored (negotiated) configuring data such as SceneProfiles, are used (without delay) in the rendering process. The pre-stored configuring data can be the result from negotiations when a complex multi-party scene has been set up, or some configuring action taken during the collaboration. In short, objects are automatically synchronized from the source, and are presented within a guaranteed time, with a controlled variable quality due to controlled drop of packets in the network, and controlled changes in scene quality.

Control packets are used for setup of collaborations, adaptive control during collaborations (Quality Shaping), and teardown. Such packets are defined to have various priority classes, seen from the application.

A. Guaranteed real-time requirements, the probability for loss shall be very low
B. Moderate real-time requirements, shall be delivered in a correct sequence.

C. No real-time requirements, but shall be delivered in correct sequence.
D. No real-time requirements, can be delivered out of sequence, all packets shall be delivered.
E. No real-time requirements, can be delivered out of sequence, packets can be lost

PCIE TRANSACTION MODES

PCIe is the protocol recommended for the link and physical layer, since there are a lot of PCIe hardware solutions supporting the needs of DMP, such as the PICMG 1.3 standard with PCIe backplanes.

This is a very short introduction to PCIe, and some arguments for using it as will be done in DMP. PCIe has a static flow control mechanism, meaning that credits are not used dynamically. Furthermore, PCIe is not designed for transmission over links with large delay, the sequence numbers (or maximum credit) are too small. This means in a DMP transmit buffer there would be several packets with the same sequence number, and which one should be selected for re-transmission? Therefore, PCIe transactions will be used in posted mode i.e., without flow control, no Ack or Nak. However, non-posted transactions (with flow control, and Ack, Nak) can be used for local transmission (between PCBs, between ICs, and between boxes with a few meters PCIe cable between), where delays are very small.

Actual posted transaction packets to be used for DMP long haul transmissions are Message Request with Data, MsgD, or Posted Memory Write, MWr.

QUESTIONS

PCIe provides virtual channels, which can be used to multiplex the two packet classes of DMP, is this a help to SEL?

Can some of the internal buffers of PCIe be used as Q1 in DMP?

HIERARCHY AND DISTRIBUTION OF FUNCTIONAL SUB-SYSTEMS

Figure 8-1 and 8-2 show the overall protocol hierarchy of DMP. Three layers are defined, the Physical-Link layer, the AppTraNetLFC layer and the Application layer. The AppTraNetLFC layer is a combined layer reusing IPv6 functionality and introducing new functionality for DMP. The protocol has one combined header, the AppTraNetLFC protocol header, and is the only protocol above and runs on top of the Physical-Link layer. The lowest layer assumes optical fibers between network nodes (wireless mobile users are considered in [RON04]). The PCIe transaction frame includes the AppTraNetLFC packet as payload, and frame- and clock synchronization. Conversion between optical signals and electrical signals are made at each end of optical links, 10-100Gbps Ethernet fiber.

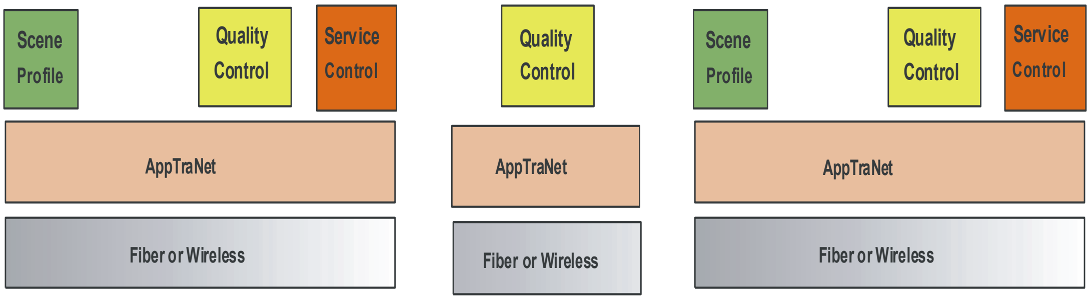


Figure 8-1. Three-layer architecture, Source Access Node, Network Node, and Destination Access Node

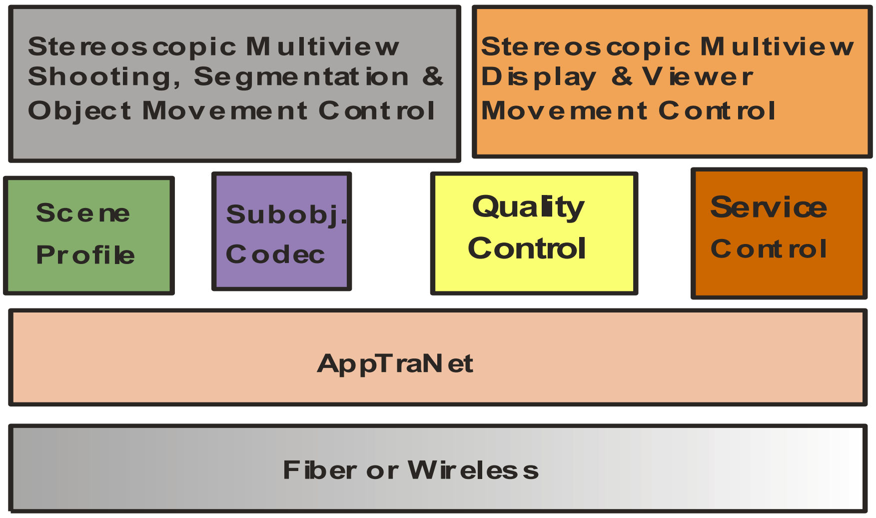


Figure 8-2. Three-layer architecture, user site

THE APPTRANETLFC LAYER

The AppTraNetLFC layer is a combined application, transport, network and partially link protocol layer. The DMP architecture uses the IPv6 protocol and IPsec security, and defines a new part supporting specific DMP requirements. To enable efficient design of high-performance hardware, ASICs, there is only one packet header called the **AppTraNetLFC header**, where parameters can be processed in parallel by physically parallel hardware, ASICs.

THE APPTRANETLFC PACKET

Some parameters of the combined AppTraNetLFC header are introduced to increase the performance, to support efficient hardware design, and to reduce the complexity of the network. The parameters allocate bits and bytes in the order as shown below. The packet length is 1.5k Bytes.

The IPv6 header [RFC07a] is used according to the standard, except for the IPv6 addresses that are used to uniquely identify users.

Node		*IP address,*	*bits*
		(Format prefix)	3
• GlobalNode,	**G**	(TLA)	13
• EuropeNode,	**E**	(reserved)	8
• DistrictNode,	**D**	(NLA1)	12
• CityNode,	**C**	(NLA2)	12
• VillageNode,	**V**	(SLA)	16
• AccessNode,	**AN**		16
• Interface, user			48

The 48 lsbs of the address uniquely identify Interfaces and users, and if 2^{10} user sites can be connected to an AccessNode, there is 38 bits left for the identity of each user individual.

The IPv6 is as follows:

- **Version**, 4 bits, protocol version = 6, drop other
- **Class**, 4 bits, priority (TBD)
- **Flow Label**, 24 bits, Service id (TBD)
- **Payload Length** - 16-bit unsigned integer, the rest of the packet following the IPv6 header, in bytes. Constant for DMP
- **Next Header**, 8 bits, identifies the type of header immediately following the IPv6 header
- **Hop Limit** - 8-bits unsigned integer. Decremented by 1 by each node that forwards the packet. The packet is discarded if Hop Limit is decremented to zero. Set to 32.
- **Source Address** - 128 bits. The IP address of the initial processing sub-system system sending of the packet.
- **Destination Address** - 128 bits. The IP address of the receiving processing sub-system of the packet (not necessarily the recipient if an optional Routing Header is present).

Integrity, authentication and encrypted payload are provided by IPsec AH and ESP in Transport Mode. See RFC 4302 [RFC07b] and RFC 4393 for a guide into IPsec. The ISAKMP (Internet Security Association and Key Management Protocol) [RFC07c] and IKE (Internet Key Exchange) [RFC07d] are used for key exchange (other may be considered).

A drawback could be that the AccessNode and its servers have to be trusted, but in DMP, the service provider generally is trusted (responsible for the services, the quality, the servers, and the network).

Authentication - Integrity and authentication.

- **Next header** – 8bits. Specifies the next protocol header part
- **Length** – 8 bits. Size of Authentication Data payload in 32 bit words – 2
- **SPI** – Security Parameter Index, 32 bits. Pseudo random value that identify the security association (SA) for this packet (value = 0 means no security association, 1 to 255 reserved)
- **Sequence number** – 32 bits.
- **Authentication Data** – variable length. Multiple of 32 bit words.

Encrypted payload, IPsec ESP.

- **SPI** – Security Parameter Index, 32 bits. Identifies the security association (SA) for the packet - an arbitrary number in combination with the destination IP address and the security protocol (ESP). Values 0 and 1 to 255 are reserved.
- **Sequence number** – 32 bits. Is not allowed to wrap (new SA and key needed)
- **Payload data** – variable length, see RFC 4303.
- **Padding** – variable length, to give the resulting ciphertext the right length.
- **Next header** – 8 bits, Specifies the next protocol header part
- **Authentication Data** – variable length, see RFC 4303.

The new part (in addition to the IP and IPsec parts) of the protocol header has the following parameters:

- **PT**, Payload type, 8 bits
- **PacketRate**, 16 bits, the current packet rate from the sub-object, indicated by the user
- **Sub-objectDrop**, 2 x 8 bits, used for controlled dropping of progressive JPEG2000 compression layers, and for controlled dropping of sub-objects compressed by the NOC scheme.
- **Timestamp**, 32 bits, used for delay measurements from a network node to an access node, and from users to AccessNodes
- **ServiceID,** 48 bits, used to identify on-going services
- **SurfaceAdr,** 16 bits, addresses the surface of a space

- **ModuleAdr,** 16 bits, addresses the module of a surface
- **View,** 16 bits, addresses the view of an object
- **PixelAdrB**, 32 bits, addresses the start pixel (x,y) of the sub-object represented in this packet (rectangle)
- **PixelAdrE**, 32 bits, addresses the end pixel (x,y) of the sub-object represented in this packet (rectangle)
- **Layer,** 16 bits, indicate the layer number of objects (background lowest number 1)
- **SPQSP**, 32 bits, reference to SceneProfile and QalityShapingProfile, used for collaborations
- **Reserve count**, 8 bits, counts the number of Reserve bytes
- **Reserve bytes**, 0-64 bytes, for future use

PACKET TYPES, PRIORITY AND DROP

The PT Payload type parameter reserves values for DMP use, defining the payload. Two main groups are defined, multimedia content packets and control packets. The following types are so far defined: visual, audio, graphics, quality shaping, object move, viewer move, scene profiles, addressing, acknowledgement, httpRequest, and httpResponse. Some control packets are used in typical request-response sequences. It is up to the application how to handle the situation if packets that need acknowledgement are not acknowledged. Since such requests do not have real-time requirements (normally), and shall have an extremely low probability for being lost, the output link queuing system shown simplified in Figure 8-3, is introduced as part of the AppTraNetLFC protocol in all nodes after switching (routing). Control packets enter queue Q2 (see later for refinement), which holds packets on a very large store. If selected by Sel, with probability p1, the control packet enters Q1 for link output. The module H is an abstraction of the output Link and physical layer, sending the packets out at the maximum packet rate given by the link capacity. Control packet transport delays are highly variable, depending on traffic patterns. The maximum length of Q2 should be so large that reaching the maximum should have an extremely low probability, before the admission control decreases the input traffic to the network.

Multimedia content packets, with a maximum guaranteed end-to-end delay, but that can be dropped selectively, are sent to buffer B2. The selector Sel fetches packets from B2 with probability 1-p1, and forwards to Q1. Q1 has a limited length and determines maximum jitter of a transfer through the node. B2 is a very short buffer used for dropping according to the sequence number of the packet. If Q1 is full, packets start queuing in B2.

Assuming NOC coding and nine sub-objects, B2 drops arriving packets as follows:

If B2.length 0-12 then join B2 else

If B2.length 13-18, drop from sub-object 8 else

If B2.length 19-24, drop from sub-object 8 and 7

If B2.length 24-29, drop from sub-object 8, 7 and 6

and so on (sub-object packets 9 are never dropped)

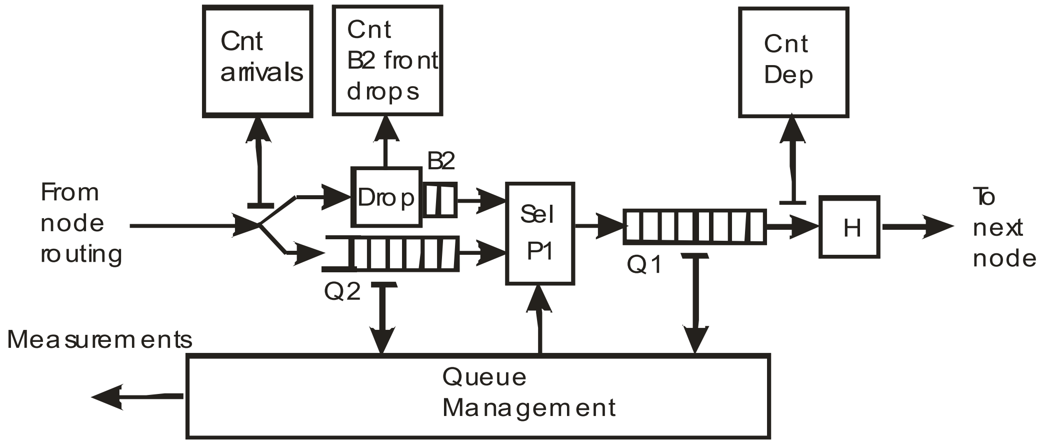


Figure 8-3. Dropping and prioritizing of packets in network nodes, AppTraNet layer (2007)

The Sub-objectDrop number is for NOC coding directly given by the sub-object number, while for JPEG2000 coding both the sub-object number and the progressive layer number have to go into the calculation. Optimizing the maximum length of B2 and Q1, the drop decision lengths of B2, and the value of p1, is part of the QualityShaping scheme design.

Since a path through the network includes nodes as shown in Figure 8-3 in series, the maximum end-to-end time delay of multimedia content packets clearly can be guaranteed. If the processing times in user equipment and switches (nodes) are constant, the end-to-end packet delay is a sum of propagation times in the path, the constant processing times in path nodes, and the waiting times in Q1 in path nodes (the waiting time in B2 can be neglected compared to the waiting time in Q1).

The protocol is a typical Request – Response protocol, and can be used between any two entities that can support the security by IPsec AN and EPS.

The PT Payload type parameter reserves a number of values (TBD) for private use, defining the payload.

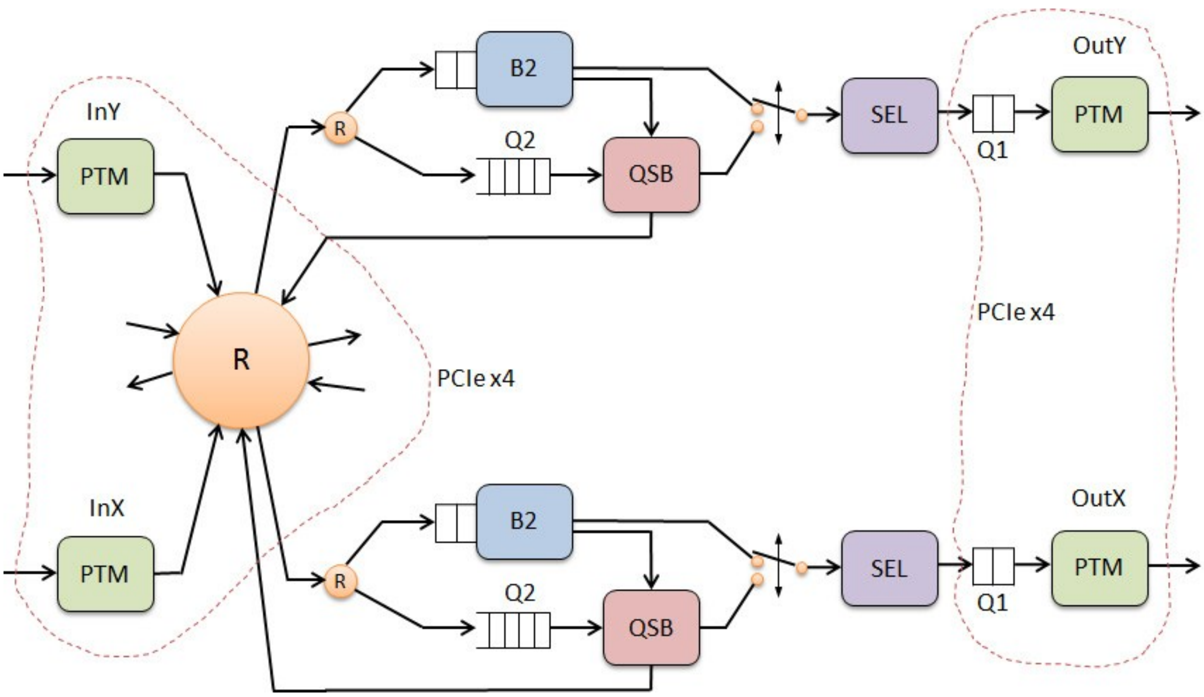
(not all details are yet defined)

PT - Payload type	Value
Visual	x
Visual, change of rate	x
Audio	x

Graphics	x
Quality Shaping(SeqNo,NextCredit)	x
,,,,	
Quality Shaping	x
Object move	x
Viewer move	x
SceneProfile(SeqNo)	x
,,,,	
SceneProfile(SeqNo)	x
Address(SeqNo)	x
SceneProfile and Address(SeqNo)	x
MsgD(SeqNo)	x
ACK(application)	x
AckFC(SeqNo)	x
NakFC(SeqNo1, SeqNo2)	x

LFC – LINK FLOW CONTROL

The flow control is applied to control packets going through Q2, that is control packets with priority B, C and D. Class E do not need any flow control, since there is no real-time requirement, packets can be lost, and can be out of sequence (this is similar to IP or UDP/IP in Internet). Class A control packets get type and Sub-ojectDrop numbers as NOC packets that shall not be dropped, and go through B2. NakFC and AckFC packes are prepared by the Quality Shaping LFC functional block (QSB), which reads the current SeqNo1, SeqNo2 or Credit values, and sent via router R to B2. See Figure 8-4. The QSB also makes measurements on all queues of the output link and forwards to the next node via R and B2. The measurements are carried by Quality Shaping packets with Sub-ojectDrop numbers as NOC packets that shall not be dropped.



Legend:
QSB – Quality Shaping LFC functional block
LFC – Link Flow Control
SEL – Selector
PTM – PCIe Posted Transaction Mode block, with Virtual Circuit table
R – Routing, based on IP address or packet type
(IPsec not shown in the figure)

Figure 8-4. Link flow control, the AppTraNetLFC queuing architecture and functional blocks.

The QSB needs to queue (Q2) control packets class B, C and D that are sent out but not yet acknowledged. When an AckFC packet is received via B2, all queued packets with sequence number up to MaxCredit are dropped. If a NakFC packet is received, packets with sequence numbers between SeqNo2 and SeqNo1 are re-sent. The receiving QSB in the other end of the link has to queue a (TBD) number of received packets if a CRC of a packet fails, and send NakFC(SeqNo2,SeqNo1). When all packets between SeqNo2 and SeqNo1 are correctly received they are put in the right place in the sequence (queue) and can be sent out. It is expected that SeqNo2 most of the time is SeqNo1+1, that is, burst errors are rare. Since the AckFC and NakFC delay and delivery can be guaranteed, the retry delay can be calculated. When packets of class D require retry, the receiving block does not have to wait to maintain the sequence of packet flow out.

When more than one link is installed between two neighboring nodes, the incoming PTM block has to keep a table of selected output links for ongoing services (ServiceID). This is necessary to be able to guarantee the sequence of control packets in a session. The table shall include the ServiceID and the 32 bit routing address of the PCIe TLP 3DW Message Request Header, giving the link allocated to the actual service session. This type of connection is usually denoted Virtual Circuit. Note that for content packets, virtual circuits are unnecessary.

As a rule, a receiving QSB sends AckFC after receiving packets for Tp (the propagation delay) milliseconds. This means that it takes 3Tp from start of sending until an AckFC is recognized by the sending block. The number of packets sent during 3Tp has to be stored in case of packet loss, during first Tp, and so on. The receiving QSB in all nodes makes measurements on Q2, B2 and Q1 and reports by sending QualityShaping packets with CurrentCredit back to the sources, via the forward stream route. See Part 13 for more explanation. All nodes in the path utilize this information to set the flow control CurrentCredit value adaptively. Control packets are then sent with maximum rate in one Tp given by CurrentCredit/Tp, which of course never can exceed the link capacity. If a new QualityShaping packet is received, a new maximum rate is calculated for the next Tp. Note that if Tp is very short (say < 5 ms) then an artificial delay can be added (Tp' =Tp + Tpa). This gives a much smoother window control than the TCP protocol which shows an on-off behavior. The Quality Shaper in the Access Nodes then may drop AV packets and in addition, deny new calls (control packets).

The propagation delay between two nodes A and B is denoted Tp. During the propagation delay Tp A sends out Np packets, Np = CTp, where C is the capacity of the link (the maximum possible data rate). After Tp the first packet arrives at B. If all packets received during the next Tp are successful, B sends an AckFC packet back to A. This takes another Tp, meaning that from start of the burst until reception of AckFC it takes a time 3Tp. See Figure 8-5.

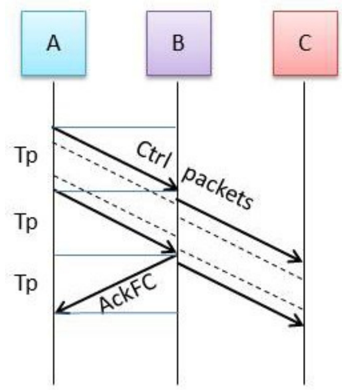


Figure 8-5. Link flow control, a single burst of packets, duration Tp, successful with AckFC.

If, in the worst case, the last packet of the burst from A to B is corrupted, it takes a time 3Tp before A knows this, which means that A has to store 3Np = 3CTp packets to be able to retransmit the packet.

If the packet burst is sent further from B to C, the same sequence repeats.
Figure 8-6 shows a sequence of a burst of four Tp with AckFCs.

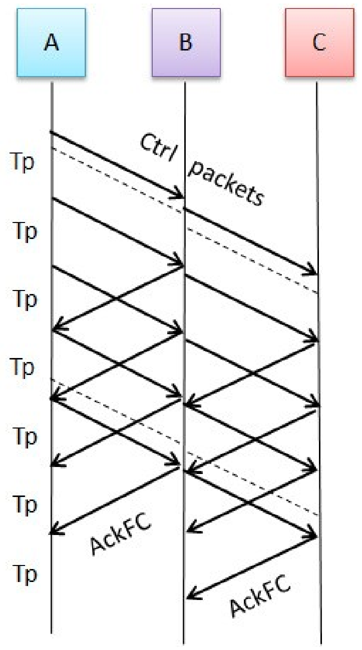


Figure 8-6. Sequence of a burst of 4Tp with AckFCs

When packets are sent with bursts less than Tp duration, that is 1 <= Np < CTp, and AckFC is always sent after Tp of success receptions.

If a corrupt packet is received, a Nak(SeqNo2, SeqNo1) is sent, asking for retransmission of packets with sequence numbers between SeqNo2, SeqNo1. Figure 8-7 illustrates this when a single packet in a burst fails.

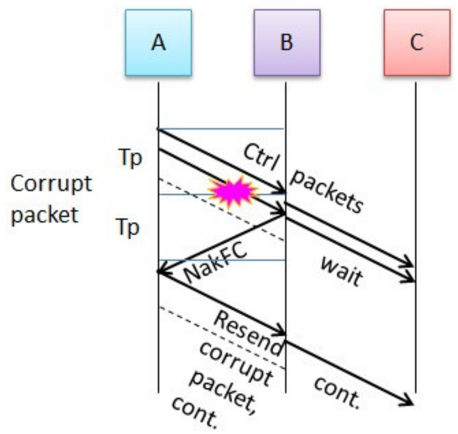


Figure 8-7. Corrupt packets and NackFC

It takes a time 2Tp before a corrupt packet is resent, and 3Tp before it is received successfully (hopefully). The consequence of this is that the SeqNo parameter needs a maximum value of at least 3Np before it wrappes.

In DMP profile (TBD) the maximum propagation delays permitted is 30 ms. If the link capacity is 100 Gbps then Np = 3 Gbit. If the link capacity is 1Tbps then Np = 30 Gbit. The maximum SeqNo shall then be at least 9E10. A 34 bit SeqNo would cover the last example. To be future proof, a 64 bits SeqNo is decided.

APPLICATION LAYER

Figure 8-2 shows six main functional blocks on the DMP application layer for the user scene. The sub-systems SceneProfile, QualityControl and ServiceControl have common functions with AccessNodes, while the three other sub-systems are implemented only in the user equipment. The network nodes handle only one application layer block, the QualityControl.

ServiceControl

The ServiceControl, allocated to the AccessNodes, are the 'beacons' of the DMP system. The ServiceControl has the top responsibility for setting up, manage, and release collaborations between users, on request from a user. The ServiceControl manages and routes the packet exchange between users and various servers allocated to AccessNodes, and between access- and destination network nodes. It cooperates via user ServiceControl with the Stereoscopic Multiview Display & Viewer Movement Control and the Stereoscopic Multiview Shooting, Segmentation & Object Movement Control in the user terminals. For this purpose, the ServiceControl has a number of supporting servers to its disposal:

QualityControl – handles scene quality shaping by dropping sub-objects.

Security Server – implements IPsec AH and ESP, receives authentication requests from ServiceControl, returns acceptance or not. Encrypts and decrypts payload.

Proxy and Address Server – receives translation requests from the ServiceControl, returns acceptance or not. Stores info about on-going services, user IDs and address relations in the domain. This Server also handles adaptation of protocols and formats for other ITC systems, such as WWW and email.

SceneProfile and QualityShapingProfile Server – receives SceneProfile check request from ServiceControl, returns acceptance or not. Stores standardized SceneProfiles. Stores standardized QualityShapingProfiles. Supports the ServiceControl to adapt scene quality according to traffic load, using the optimal QualityShapingProfile.

The user has a corresponding ServiceControl, cooperating closely with the ServiceControl of the Access Node. The following modules are used:

QualityControl – shapes scene quality by adjusting QualityShaping parameters

Security Module, implements IPsec AH and ESP, should be a detachable hardware module, e.g. with a smart-card.

SceneProfile and ***QualityShapeProfile*** Descriptions.

Sub-object encoder and decoder. The NOC lossless compression scheme and JPEG2000.

A sub-system for ***Stereoscopic Multiview Shooting, Segmentation and Object Movement Control***.

A sub-system for ***Stereoscopic Multiview Display and Viewer Movement Control***.

A special type of user is the servers storing objects and scenes that are shot or synthesized in advance. The first four bullet points above are part of such a server, and the behavior is as for a normal user.

EXAMPLE, ESTABLISHMENT OF A MULTIPARTY DMP

Figure 8-8 shows a successful setup of a one-way collaboration from user A to B. When there are N users in a collaboration, all users have to set up a one-way collaboration to all the others, giving N(N-1) set-ups. The ServiceControls in the AccessNodes handle the setup and the management of the sessions until they are released. All servers and users are uniquely identified by their IP addresses. All possible services are described by standardized SceneProfiles. Users must choose SceneProfiles during set-up of a service, but this can be changed during the session. The SceneProfile defines how to shoot videos and record sound, and how to present on the receiving user display. The control packets indicated by 1a to 4c in Figure 8-8 have the following payload description:

Packets Description

1a - AH and ESP secured service request (ServiceID, UserIDs, A, B, C, D, E and F,
{set of prioritized SceneProfiles})
1b - 1a sent to Security for security check
1c - 1a sent to Proxy for address validation and translation
1d - Positive response to 1c (ACK)
1e - Positive response to 1b (ACK)
1f - 1a sent to SceneProfile for acceptance of scene
1g - positive response to 1f (ACK)
1h - Service request sent to Security for AH and ESP addition
1i - Positive response to 1h (ACK)
2a - AH and ESP secured service request
2b - Positive response to 1a (ACK)
2c - as 2a
3a - 2c sent to Security for security check
3b - 2c sent to Proxy for address validation
3c - Positive response to 3b (ACK)
3d - Positive response to 3a (ACK)
3e - 2c sent to SceneProfile for acceptance of scene
3f - Positive response to 3e (ACK)
3g - 2c sent to user for acceptance
3h - Positive response to 3g (ACK)
3i - Service acceptance sent to Security for AH and ESP addition
3j - Positive response to 3i (ACK)
4a - AH and ESP secured service acceptance
4b - as 4a
4c - as 4b

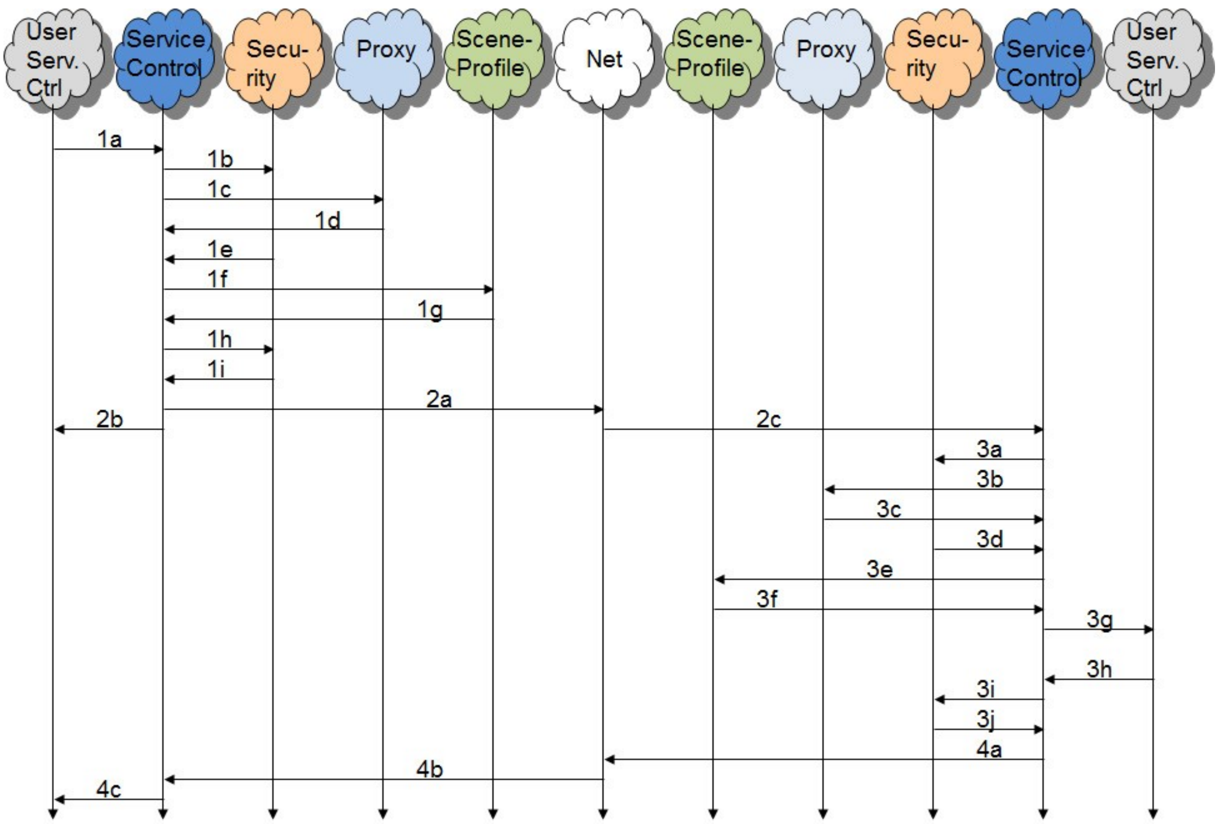


Figure 8-8. Set-up of multi-party DMP

User A can now start shooting objects in the agreed scene, sort out sub-objects according to the agreed SceneProfile and send audio and vision AppTraNetLFC packets to user B. User A Control activates the Stereoscpic Multiview Shooting, Segmentation and Object Movement Control, which tracks the objects. At user B, multimedia content packets immediately present their content on the display. User B ServiceControl activates the Stereoscopic Multiview Display and Viewer Movement Control, to track the eyes of the viewer to find the object on the display he focuses at any time. The QualityControl is also activated to adapt the quality of the scene to the traffic load in the network.

EXAMPLE, COLLABORATION SPACE PACKET HANDLING

A multimedia content packet should always represent only one sub-object. When generated by the sender, the packet is coded, and finds the way through the network, through decoding, and to the display RAM at the receiving end, using only the AppTraNetLFC packet header parameters {IP address, PT, ServiceID, SurfaceAdr, ModuleAdr, View, PixelAdrB, PixelAdrE, SPQSP}. The packet is regarding coding, decoding and display, totally independent of all other packets. The content (pixels) is presented to the user immediately (update rate 250 Hz). If a sub-object is not updated after a certain time (typically 10 ms), the missing pixels are constructed by interpolation by the Display Processing block. In network nodes, as shown in Figure 8-4, content packets enter the buffer B2, and are forwarded to Q1 or are dropped in B2. From Q1 the packets are output on the link to the next node.

Figure 8-9 shows the blocks that process the multimedia content and control packets at the receiving user. All packets coming in to or going out from the router R are secured by IPsec. The IP address points to users which may set up individual services. The PT parameter determines whether it is a multimedia content packet or a control packet. The B2 drop module drops sub-objects if necessary, according to the Sub-objectDrop number. The Decoder decodes the packets (NOC, JPEG 2000, or other). Routers then send the packets to the right place in the Display Processing buffer, given by the SceneProfile (SPQSP parameter), the SurfaceAdr, the ModuleAdr, View,and the PixelAdrB and PixelAdrE parameters.

Control packets join queue Q2, are subjected to link flow control, and then passed to the Service Control or other blocks.

Scenes are shot by Camera Cluster Arrays, processed according to the agreed Scene- and Quality Shaping Profiles, and coded into AppTraNetLFC packets. After the IPsec procedure is applied, the packets are routed to the allocated Access Node.

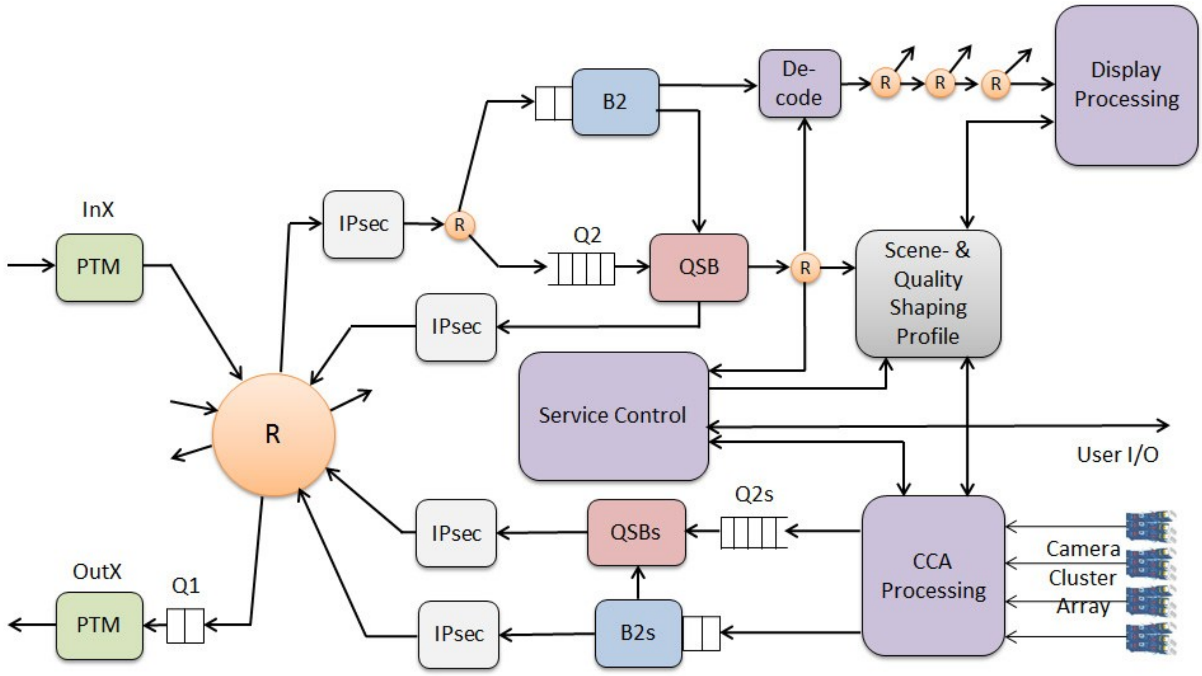


Figure 8-9. Collaboration Space Hardware/ Software architecture

[RFC07a] RFC 2640, IPv6, www.ietf.org/rfc/rfc2460.txt

[RFC07b] RFC 4302 IPsec, IP security. www.ietf.org/rfc/rfc4302.txt

[RFC07c] RFC 2408, ISAKMP, Internet Security Association and Key Management Protocol. www.ietf.org/rfc/rfc2408.txt

[RFC07d] RFC 4306, IKE, Internet Key Exchange. http://tools.ietf.org/html/rfc4306

[RFC07e] RFC 2396, URI, uniform resource identifier, www.ietf.org/rfc/rfc2396.txt

[RON04] Rønningen, L A. Scalable, distributed wireless transmission at 40 GHz and above. Weekly seminar, IET, NTNU 2004

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

The basic principle of DMP Scenes, objects and sub-objects are introduced in Part 6.

SceneProfiles describe the characteristics and limits of the shooting and presentation space, scenes, sub-scenes, objects and sub-objects, movements of objects, and characteristics of each sub-object. During a collaboration the scene may vary fast, objects pop up and disappear, and the importance of objects changes. The dynamics of the scene shall be within the limits described by the SceneProfile of the scene, otherwise events will not be shot and presented. These limits are determined by physical constraints as well as the technical specifications of the object shooting equipments and the display equipment.

A subset of SceneProfiles are the QualityShapingProfiles, described later.

To give the virtual collaboration a near-natural feel, the size, form and position of objects should be near natural, the displayed picture shall be flicker-free, and individual pixels should not be visible at say 20 cm viewing distance. The normal comfortable distance between people may vary from 0.5 meters to several meters in close situations, but may also be several hundred meters. The natural viewing area of human eyes is used as the ‘frame’ of the scene.

If a user A is moving around in his room, the other users B and C watching him on their stereoscopic walls/screens shall have the feeling that A is actually walking in a real extension of their own rooms. If B and C are moving and A is not moving at all, B and C shall see A from different angles (as natural) as they move (multi-view).

A problem encountered in video conferences, is that persons feel that the remote persons do not look into their eyes when they should. This is due to that a camera cannot be placed exactly in the position where a viewer focuses. In paper their paper on ‘Multiview’ [NGU07], Nguyen and Canny show how to solve such problems of ‘spatial distortion’. The concept of *spatial faithfulness* was introduced as a measure for the phenomenon (see how the Hems Lab solves such problems).

The concept of frame (video frame) and field are not used in DMP. Instead we use hierarchical scenes, sub-scenes, objects and sub-objects. Instead of frame buffers (for display), object buffers (hardware) and screen buffers are used.

The views are treated independently. Streams representing one view are independent of all other view streams. If packets from one view are dropped, it does not affect other views at all. There is no 'compression' between views, because the selection and number of views are adaptable during collaboration. The number of views may vary from only one to say fifty, depending on the traffic in the network and the number of viewers that are present in the room and how they are moving around. In the following, it is therefore sufficient to describe only one view. Generating intermediate views by interpolation is straightforward [WIK07].

A scene is divided into several sub-scenes, each consisting of a number of objects. Each spatial object is divided into 2, 4 or more sub-objects, which are sent in separate packet streams. Each spatial object is updated after a certain time, giving temporal objects that are also divided into 2, 4 or more sub-objects which are sent in separate packet streams.

The AppTraNetLFC packets defined in this report, are the basic entities for presentation, security, quality, storage, and transport data units. To allow segmentation of packets for wireless transmission, the packet length is nominal 1500 bytes, but its body is divided into five parts each of 300 bytes so fragmentation into five parts can be done. The content of one audio or visual packet part does not depend on the content of other packet parts in order to be presented for the users.

The packets, using IPsec, are self-sufficient regarding security. Audio or visual packets can be dropped as individuals (however selected, controlled) independent of other packets. The drop only gives a temporal reduction in quality, and do not stop the stream or the presentation.

A packet cannot contain more than one sub-object from one object. If a packet is not completely filled (small objects), the quality of the payload can be increased in order to fill the packet, or optionally, sub-objects from other objects can be included.

ScenProfiles can be described by means of Collada, as outlined in the next section.

COLLADA

The XML based Collada language [COL07] can be used for scene descriptions. The main argument in favor of Collada is that it defines an interchange format, which can support interchange of data between the private scene data formats used by numerous computer graphics tools. Examples of tools are 3D Studio max, Maya and Softimage.

However, in DMP all data structures are binary, in order to obtain high performance when executed by specialized hardware. To make it easy for the developer, a software tool shall be developed to provide translation between the binary executable data and the easy-to-understand Collada language's textual representations.

Collada supports the transfer of 3D models, vertices, polygons, textures, shaders, lights, cameras, and much more, between applications. Collada can be extended to handle the description of SceneProfiles, as defined by DMP (remains to be done).

The Collada language is defined by a set of language rules called CoalladaSchema, based on the XML Schema language.

EXAMPLE OF SCENEPROFILES, A VIRTUAL WORKING DINNER BETWEEN THREE RESEARCHERS IN TRONDHEIM, PADOVA AND POZNAN

Two SceneProfiles defined in advance are negotiated, X1 for the researcher in Trondheim, and X2 for the researchers in Padova and Poznan. (A bullet-point type of English is used instead of Collada here).

The scene viewed from Trondheim looks like Figure 9-1, while from Padova it may be as in Figure 9-3.

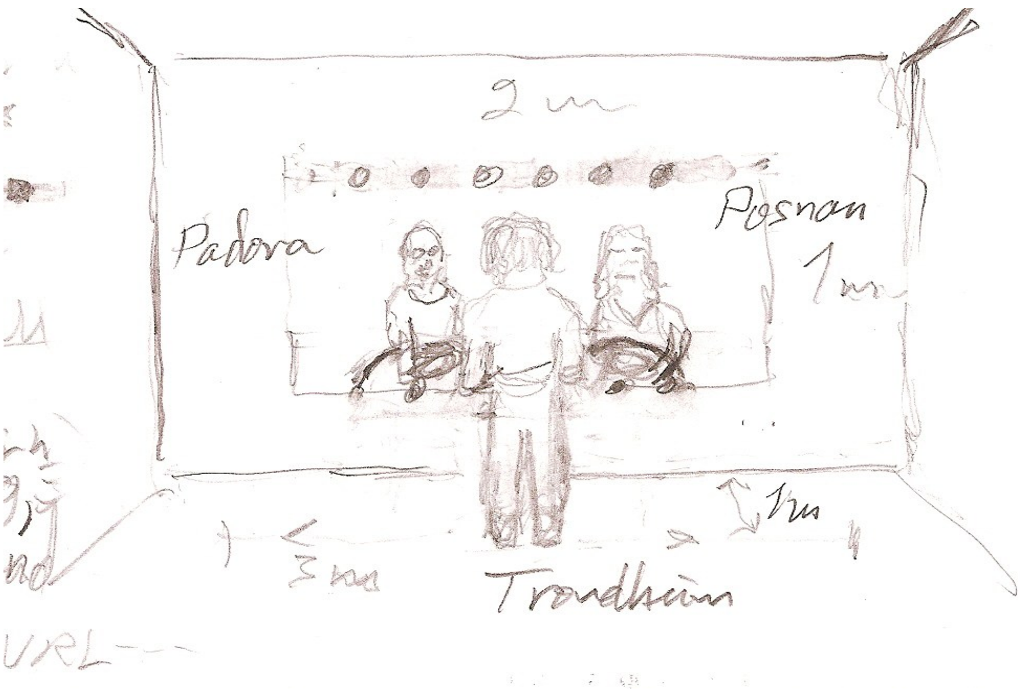


Figure 9-1. Virtual Working Dinner, scene Trondheim.

Figure 9-2. Virtual Working Dinner, scene Padova.

The X1 and X2 SceneProfiles includes the following:

- One researcher in each place, two objects are to be tracked, the very important face object and the important body object
- Two researchers are projected side by side, in two windows, 1 x 1 m each
- The background object is synthetic, stored on a server with URI = x.
- The researchers are shot by 2 x 6 camera systems against a blue wall
- There are 5 views sideways, and two views up-down
- Back-projection is used
- Real round tables in Padova and Poznan, researchers are sitting and eating from plates on the tables, X2 profile only
- The researchers are not closer than 1m from the screen (and cameras), X2 profile
- The researchers are shot in a 1 x 2 m area, and the body is segmented from the blue background, X2 profile only
- The faces are shot in a 0.4 x 0.3 m area, and segmented from the background
- The researcher in Trondheim is standing, eating, keeping the food in his hands, X1 profile

- The researcher in Trondheim is moving around, not closer than 1 m from screen (cameras), and within a 3 x 3 m square
- The shooting is 1:1, that is, shown in real size in the projection screen
    - video camera array configuration (TBD)
    - video projector array configuration (TBD)
- The pixel size is 0.25 x 0.25 mm, 0.05 mm for a view
- 9 sub-objects are used, adaptable down to 2
- The mod JPEG2000 compression is used for the background object, and the compression can be varied 100 – 50
- The face objects and the body objects are compressed using NOC

REFERENCES

[COL07] COLLADA, **COLLA**borative **D**esign **A**ctivity for establishing an interchange file format for interactive 3D applications. www.collada.org/public_forum/index.php

[NGU07] Nguyen, D, Canny, J. MultiView: Improving Trust Formation through Spatial Faithfulness, Proceedings of the 2007 ACM Conference on Human Factors in Computing Systems

[WIK07] Interpolation. http://en.wikipedia.org/wiki/Multivariate_interpolation

PART 10: SUB-OBJECT ENCODER AND DECODER

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007, 2009), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

Sub-objects are sorted out according to the SceneProfile associated with an actual service. Each sub-object of arbitrary shape and size is generally represented by rectangular stereoscopic images with the right shape and required spatial and temporal resolution. A mask for the sub-object determines the transparency for each pixel. The motivation for this sub-division is to avoid progressive layering, and obtain equal-priority entities that are independent until viewed on the screen by the user. In the case of progressive coding, basic layers are sent as important AppTraNetLFC packets, while higher layers have decreasing priority. At first sight, progressive coding with successive refinement seems to be an advantage regarding scalability, but in the network where packets are dropped randomly, it is a drawback. Important packets have to be specially protected or prioritized. A flat coding structure is expected to give a simpler and smoother degradation of quality when packets are dropped or lost in the network.

The sub-object division also provides unique possibilities for variable resolution over the surface of any object, simply by coding each sub-object of an object with variable shape, size and resolution.

Sub-objects shall be compressed independently using shape adaptation, either with a new scheme proposed here, called NOC (Near-natural Object Coding), a slightly modified JPEG2000 [JPE02], or other approaches based on Discrete Wavelet Transforms with Zero Tree Coding [YIN03]. NOC supports the extreme quality requirements of DMP, the ‘near-natural feel’, and is inspired by the PNG compression standard [PNG07] and the VASARI (the visual arts system for archiving and retrieval of images) project [MAR02]. The NOC scheme is lossless and can have a flat priority structure. With JPEG2000, the division of an object into sub-objects before applying the algorithm, helps flattening a progressive structure.

The terms, definitions and concepts used for NOC are related to the PNG standard, Chapter 3 and 4 [PNG07], with some additions given here. The PNG standard, Second Edition, has to be read to get a full understanding of this section.

Figure 10-1 shows some examples of sub-object division, where the distance between pixels of each sub-object is constant over the area. This is easily extended to 25 sub-objects or more. Note that an object can have any shape. When a fast moving object is shot, the definition of the object can also include the background covered by the object in the previous picture. This is important when the background is updated at a lower rate than the object.

NOC

NOC (Near-natural Object Coding), uses the filtering and deflating compression principles from PNG [PNG07], and adds new specifications. NOC can use up to 64 spectral sub-bands from the visible spectrum, each with a bit depth of up to 24 bits, and includes the PNG RGBA truecolour type, using a bit depth up to 16 bits per component. Objects consisting of several layers (from e. g., Adobe After Effects), can also be treated separately.

1	2	3	1	2	3
4	5	6	4	5	6
7	8	9	7	8	9
1	2	3	1	2	3
4	5	6	4	5	6
7	8	9	7	8	9

Figure 24. Sub-object allocation, 9 sub-objects

Principal Component Analysis (Karhunen-Loeve Transform) [KAR07] can be performed on the visible spectrum, to obtain a smaller number of uncorrelated components that can be used further. As shown in [DEC90], the number of spectral sub-bands applied can be reduced to 12, obtaining near-perfect images of paintings. In the VASARI project, 7 spectral filters (Gaussian, with 50 nm intervals between 400 and 700 nm, and a bandwidth of 70 nm, measured at half maximum transmittance) was used, giving a quality close to the 12-filter approach.

The contenders to the NOC compression scheme could be lossless JP2000, the lossless JPEG-LS [JPE07], arithmetic compression, or a scheme based on the Burris-Wheeler transform [WIK07a]. On average, the performance of these schemes is quite close [SAN00]. The reason for ending up with NOC and JPEG2000 is as follows. JPEG2000 has become the cinema standard, and performs very well on natural pictures and compression ratios 10-50, and on still pictures. The NOC scheme will be (as PNG) free of patents or fees. It is a further development of the filtering and deflating compression of PNG, and is expected to outperform JPEG-LS (and JPEG2000) on artistic pictures [SAN00]. In addition, the filtering scheme of NOC (PNG) has in special cases shown to give extreme compression ratios without loss [ROE03]. The filtering has a great potential to be improved further in the future. The NOC scheme also supports parallel and independent decoding of packets.

CONCEPTS

The terms, definitions and concepts used for NOC correspond, with some changes and additions, to the PNG standard. An important change is that chunks are covered by the AppTraNetLFC protocol and the adaptive QualityShaping Profiles of DMP. Another important change is the definition of image as used by NOC compared to PNG. In NOC, an image is one (of many) representation of a sub-object of arbitrary shape. The image has the same shape as the 2D projection of the sub-object. To obtain a stereoscopic image, two images of the same object are shot with two cameras (at least) as described in the introduction. To obtain stereoscopic multiview, a large number of cameras have to be applied, depending on wanted quality. Lacking views can also be synthesized from one view and a depths map. It is important to notice that each sub-object is treated independently, transmitted in independent streams and independent packets (this may be called n-dimensional interlace, other types of interlace or pass extraction is not used). For NOC four kinds of image are distinguished (a special case is the PNG RGBA tricolor type, which follows the PNG standard).

- The source image is the image available from the camera (shooting) array system. It must contain information necessary for the encoder to extract a reference image.
- The reference image, which only exists inside the encoder, represents an arbitrary shaped area of pixels. It represents a sub-set of the source image. It can always be recovered from a NOC datastream, in case of packet drop with lower quality, otherwise exactly. The pixels are equal, containing a number of samples, from 4 to 65, where the last sample is an alpha sample and the others represent sub-bands of the visible spectrum. The sample depth is adaptable 1- 24 bits. Each horizontal row of pixels is called a scanline. A bitmap is used to indicate which pixels within a rectangular area belong to the reference image. This is the most general case, since an object can have ’holes’ where objects in the background are visible.
- The NOC image is obtained from the reference image by a series of transformations. An encoder generates a NOC datastream from the NOC image. A decoder generates a NOC image from the NOC datastream.
- The delivered image is constructed by a decoder by a series of transformations applied to the NOC image.

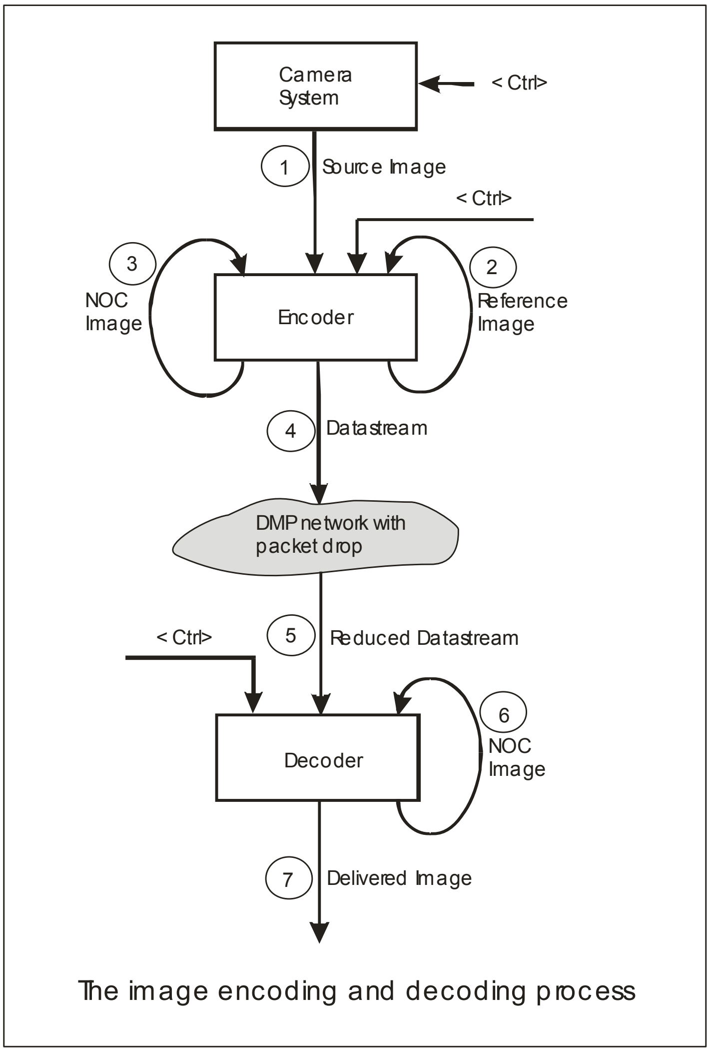


Figure 10-2. Encoding and decoding.

REFERENCE IMAGE:

A reference image represents a sub-object, see Figure 6-1, that shows an example of a scene divided into objects and sub-objects in space and time. The sub-objects have arbitrary shape, and a shape mask with one bit per pixel, which indicates with a ‘1’ that a pixel belongs to the sub-object. En edge sub-object is always generated, giving the outer edge and inner edges of the object. The shape mask, which is rectangular, uses the lower left corner as position address relative to the scene. The address (x,y pixels) is included in the shape mask data. Each horizontal row of pixels is called a scanline.

COLOUR SPACE

n spectral sub-bands, one alpha channel

Defined up to now and mandatory: n = 12 ([DEC90]), 9, 7 (VASARI [MAR02]), 3 (RGB). In addition, an alpha channel is included in each case.

Sample depth = 24 bits, adaptable down to 8 (TBD) bits.

Gamma correction is mandatory, as described in 13.16 and Annex C of the PNG standard [PNG07].

Alpha representation

The alpha channel has the same sample depth as the other components. Alpha = 0 means full transparency, and all bit of the alpha sample equal to ‘1’ means full opacity. In between, there is a linear relationship.

YCrCbA with 4:2:2 sampling and scalable sample depths from [24,16,16] is (TBD) and optional.

Up to m layers of a scene can be sent. An object may be represented on one or more layers (see Adobe AfterEffects). Each layer is handled independently. Each layer is then a scene of objects, which in turn are divided into sub-objects.

NOC IMAGE:

Scanline serialization means that each row of pixels is represented as a sequence of bit words (8-24 bits).

FILTERING

Adaptive filtering based on PNG filter types 0 to 4 in the following way is mandatory: compute the output scanline using all five filters, and select the filter that gives the smallest sum of absolute values of outputs (as proposed in 12.8 of the PNG standard).

New filter methods shall be studied (TBD).

COMPRESSION

PNG Compression method 0 is mandatory for NOC. This method is the deflate/inflate compression based on an LZ77 derivative [LZ707] and with Huffman coding. For NOC, a sliding window of 5*256 bytes is applied. The Shape mask and edge sub-object are compressed using run-length coding. See RFC-1950 and RFC-1951.

DATASTREAM:

The compressed datastream consists of independent AppTraNetLFC packets. Each packet contains data from a certain part of a sub-object. The packing format of the compressed data is described by SeneProfiles, which in turn are carried by AppTraNetLFC packet type x. Generally, only a part of a sub-object data can be placed in each packet. A packet shall be filled up with compressed data (5*256 bytes) from, for example, n spectral sub-bands, one alpha channel and the shape mask, starting at the last pixel position of the sub-object put in the previous packet. Each packet is self-contained, but each decoder needs the current SceneProfile/QualityShapingProfile to be able to decode the packet correctly. On average, the compression is about 5x, and this means that 5*5*256 bytes of un-compressed data are packet into one packet.

Since the NOC compression ratio varies between 1.0 to say 10 or more, NOC uses a deflate window of 5*256 bytes, that means that the compression algorithm is run repeatedly on batches of 5*256 bytes until a packet is 'overfilled'. Then the last compressed batch is dropped, and instead uncompressed data representing a number of pixels are used fill up the packet. If RGB (8,8,8) is used, each packet carries 2133 pixels, or a squared sub-object of 46 x 46 pixels.

DECODING

Drop of packets can be controlled or random. In case of wireless links, random drop of packets is normal, while for fiber based nets, all dropping occur in node queues. It is important to have an overview of when objects and sub-objects last were updated. If the sequence number and time stamp for the last packets for each sub-object are stored by the receiver, the 'presenter' can decide when 'outdated pixels' have to be substituted by new ones found by interpolation from the surrounding pixels. For visual important objects, the 'living age of pixels' can be about 5-10 milliseconds. In every case, new arriving pixels are buffered as long as needed for decompression, and then immediately presented on the screen (the concept of frame or field is not used in DMP).

RGBA TRICOLOR, MODIFIED PNG

From the PNG specification (second edition) the RGBA tricolor type is with some modifications mandatory. Scenes are divided into objects and sub-object, where sub-objects of arbitrary shape are coded into independent images. However, the other image types (section 4.4, point b., c., d., and e., Chapter 6) are made optional. Pass extraction and interlacing as described by the PNG specification (section 4.5 and Chapter 8) are optional. The data stream structure follows the mandatory PNG specification (Chapter 5 and 7). The pre-filtering in PNG is extremely important for images with certain characteristics. The current pre-filtering standard (filter method 0) shall be further extended and developed for DMP cases (TBD). Compression method 0 (deflate/inflate (derived from LZ77), Chapter 10) is mandatory. When wireless links are used, the window size shall be set to 256 bytes (to fit into packet segments for wireless). The Chunk specification (Chapter 11) is mandatory.

PNG RGBA TRICOLOR TYPE

This follows the subset of the PNG standard version 2 regarding the RGBA tricolor type.

JPEG2000

Using sub-objects as components in JPEG2000 is straightforward. Tiling will not be supported. The ROI can be used to enhance resolution of arbitrary shaped objects. The surroundings can be set to the lowest possible resolution, giving one specific color and very efficient compression of unimportant surroundings. At the receiver the surroundings can be made transparent, so that composing a scene from ROIs is possible [JPE07].

FUTURE WORK ON COMPRESSION

(TBD)

REFERENCES

[DEC90] Deconinck, G, “Multi-channel images: Acquisition and coding problems”, M.Sc. thesis, Katholieke Universitet Leiven and Ecole Nationale Superiere de Telecommunications, Leuven, Belgium 1990.

[JPE07] JPEG-LS http://www.jpeg.org/jpeg/jpegls.html

[JPE02] JPEG2000, www.jpeg.org/jpeg2000/

[KAR07] http://en.wikipedia.org/wiki/Karhunen-Lo%C3%A8ve_theorem

[LZ707] LZ77 compression http://en.wikipedia.org/wiki/LZ77_and_LZ78

[MAR02] Martinez, K., Cupitt, J., Saunders, D. and Pillay, R. (2002) Ten Years of Art Imaging Research. Proceedings of the IEEE 90(1) pp. 28-41.

[PNG07] PNG, Portable Network Graphics, www.w3.org/Graphics/PNG/

[ROE03] Roelofs, G. PNG: The Definitive Guide. O'Reilly & Associates, Inc, 2003.

[SAN00] Santa-Cruz, D, Ebrahimi, T, Askelöf, J, Larsson, M, and Christopoulo,C A. JPEG 2000 still image coding versus other standards. *Proceedings of SPIE* Vol. 4115, 2000. http://www.jpeg.org/public/wg1n1816.pdf

[WIK07] Burrows-Wheeler Transform. http://en.wikipedia.org/wiki/Burrows-Wheeler_transform

[YIN03] Yin-Tsung Hwang; Sung-Bo Chen; Shi-Shen Wang. Efficient shape-adaptive zero tree coding scheme and its FPGA implementation for DWT based progressive coding Signal Processing Systems. SIPS 2003. IEEE Workshop on Volume , Issue , 27-29 Aug. 2003 Page(s): 357 – 362

PART 11: 3D SHOOTING, SEGMENTATION AND OBJECT MOVEMENT CONTROL

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

This control (SSOMC) handles multiview tracking and shooting of objects moving relatively to a background and other objects. Note that this part has been followed up in a separate project, The Collaboration Surface project, see [COL11].

Consider the following simple example. A red ball with fine, important details are moving over a blue not-so-interesting background, but with a certain depth. The static background is shot every 50 ms in stereo by two DV cameras (A). The ball is tracked and shot every 5 ms with two cameras (B) with 200 Hz and 240 x 480 pixels. If only the picture of the ball is shot and presented, the 10 previous ball pictures will be shown together with the current on the background. To update the part of the background covered by the previous balls, the B camera also shoots the background the previous ball covered, and the whole B picture is copied on top of the A picture. However, the spatial resolution around the ball should be gradually reduced to the same as the background. This has not yet been tested.

THE CAMERA CLUSTER ARRAY

A Camera Cluster Array consists of an array of Clusters. A Cluster consists of a number of densely placed cameras. A number of Camera Cluster Arrays are used in Collaboration Surfaces.

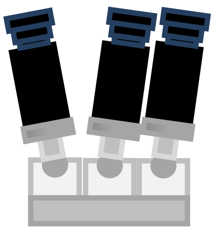


Figure 11-1. Camera Cluster

As described in [CCA11], extremely advanced shooting of scenes can be accomplished.

SYNCHRONIZATION

It is assumed that signals (analog voltages) representing the pixels of an image from the imaging sensor, are generated simultaneously. However, due to limiting pinning and IO of the sensor chip, the readout will take some time, but have to be finished before the next image is shot. As a compromise, it is assumed that the pixel data from each sub-object can be read out in parallel, while the readout from sub-objects is performed serially. The compression and packing are assumed to take a constant time (for a certain scheme), and is also carried out in parallel for sub-objects. With the scheme as described above, the synchronization of sub-objects is automatically taken care of. It is assumed that packets from one sub-object shot at a certain time are spread evenly over the interval before the next shot of the same sub-object. The time between packets are given by the QualityShapingProfile known in advance.

REFERENCES

[COL11] The Collaboration Surface project, http://www.item.ntnu.no/people/personalpages/fac/leifarne/collaborationspaces

[CCA11] Rønningen, L A. "CCA Functional Structure & Architecture". Technical Report, Item, NTNU 2011. http://www.item.ntnu.no/_media/people/personalpages/fac/leifarne/ccafustruct_arch1.docx

PART 12: 3D DISPLAY AND VIEWER MOVEMENT CONTROL

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

This Control (DVMV) identifies viewers, their current position and movements in the room. The DVMV tracks each viewer's eye movements, and sorts out which object in the scene each viewer is focusing on at any time. The importance of the object, the object characteristics, and the likely objects to be focused on, is determined by the SceneProfile. DVMV at user B informs the shooting system SSOMC at user A to shoot the objects individually with a specified quality from the correct views. This shooting information is sent via user B's and user A's ServiceControls in AppTraNetLFC packets, Payload type = yy. These packets are important and independent, and need not be acknowledged.

FOVEATION

In his Master's thesis at NTNU, Salater has shown [SAL05] that by applying foveation (applying enhanced resolution to the object a viewer focuses at, also denoted ROI) to videos, the data rate can be reduced substantially, varying with the type of video content. In one case, the video 'Pulp Fiction', in 90 % of the time, the users focused at only 7.3 % of the screen. Tests also indicated that viewers did not hardly perceive any difference whether the unfocused part of the screen has low or high resolution. In other cases the advantage of applying foveation is noticeable, but not that big as in the 'Pulp Fiction' case.

Other research that supports the advantage of foveation was carried out by Peli et al [PEL05]. Six video clips were shown to 26 viewers at different age, and an eye-tracker was used to register what the viewers were looking at. In more than half of the tests, 15-20 out of 26 viewers focused at an area less than 13% of the screen.

Viewers focus on objects within 1-2 viewing degrees at a time [SAL05]. And the eyes normally move from one object to another, the movements are denoted saccads. The movements are fast, up to 900 degrees/ second = 30 degrees in about 33 ms, and the viewer can hardly perceive any image information. The saccadic movements come typically when a viewer loses focus on fast moving objects (> 100 degrees/ sec).

Depending on position of the viewer (distance to the screen), the size area the viewer can focus is registered by DVMV, and the corresponding information is sent to the remote SSOMC in order to apply foveation immediately. When a viewer moves, the updates have to be sent promptly to SSOMC which can change its fovation. The delay performing the operation is probably not the most critical, but this has not yet been tested. If the foveated area is somewhat larger than the focused area, the viewer will probably not notice any difference.

The rate of the saccadic movements tells us that if the behavior of DMP collaborations shall be near-natural, the DVMV system has to track eye movements in milliseconds, and transfer this information to the remote SSOMC system in milliseconds, and the SSOMC system also has to position and configure its cameras in milliseconds. This indicates that the cameras should always track important objects, set their configuration to 'right' all the time, shoot images, but not send them. When a request for focusing comes, it is just to start the packing and start sending.

PERCEPTION OF OBJECTS WITH TEMPORAL CHANGING RESOLUTION

In his master's thesis Tor S Jenssen experiments with regeneration of DMP object based images, as described by the DMP Quality Shaping concept. Using a qualitative (subjective) approach, test images were decomposed into objects and sub-objects, and sub-objects were then dropped and regenerated by known interpolation techniques, such as bi-cubic interpolation. The regenerated images were compared with original images in blind tests. Object-oriented scene composition seems to give better performance than standard images. He also showed that periodic drop/ regeneration with a visible frequency (perceived as flicker) gives annoying disturbances [JEN10].

PRESENTATION AND SYNCHRONIZATION

The packets are synchronized with constant inter-departure time from the source, and are then sent through the network. The maximum queue length (to be configured) of the nodes in the path limits the jitter of the receiving time at the destination.

When a visual packet arrives to the DVMA system the content is decompressed, and the pixel samples are loaded into image RAM and shown to the viewers immediately. There is no waiting for other sub-object packets. The various parts of a sub-object (and object) arrive with a jitter caused by link queues only. The maximum jitter is guaranteed, and the minimum jitter is zero. How this jitter is perceived is not known, and has to be tested. An important system configuration problem is to set the maximum queue lengths in nodes so that the jitter is below the level of acceptance for important objects.

Packets are dropped by the shaping scheme, and packets can be lost due to failures. The SSOMC system checks every 10 (TBD) ms if the whole display has been updated, and if an area of the display has not been updated, the missing pixels are found by interpolation. To assure that at least some of the sub-objects of an object are updated every check interval, sub-object packets are dropped in a controlled way. To know when an area of the display has not been updated, we introduce a picture mask which is updated by setting '1' in updated pixel positions when visual packets arrive. After the missing pixels are found by interpolation, the whole mask is set to '0'. To speed up the interpolation process, the display RAM can be segmented to support parallel processing. The interpolation can be performed using linear or non-linear interpolation.

Figure 12-1 shows the sequence of packets when a session is already set up, and sub-object streams are OTA[3]. Users always indicate in the packets the actual packet rate, and when a rate change occurs. After a while, the user at B change focus from one object to another, and a Viewer move packet, Payload type = x, is sent from DVMV at user B to SSOMC at user A, which adjusts its camera configuration.

Security handling of the packets are not shown in the Figure 12-1, but is similar to that in Figure 8-4.

Packets *Description*

5a, 5b SceneProfile is negotiated
6a-d AH and ESP secured audio and vision packets are sent from user A to user B
7a-d User B changes focus from an object to another

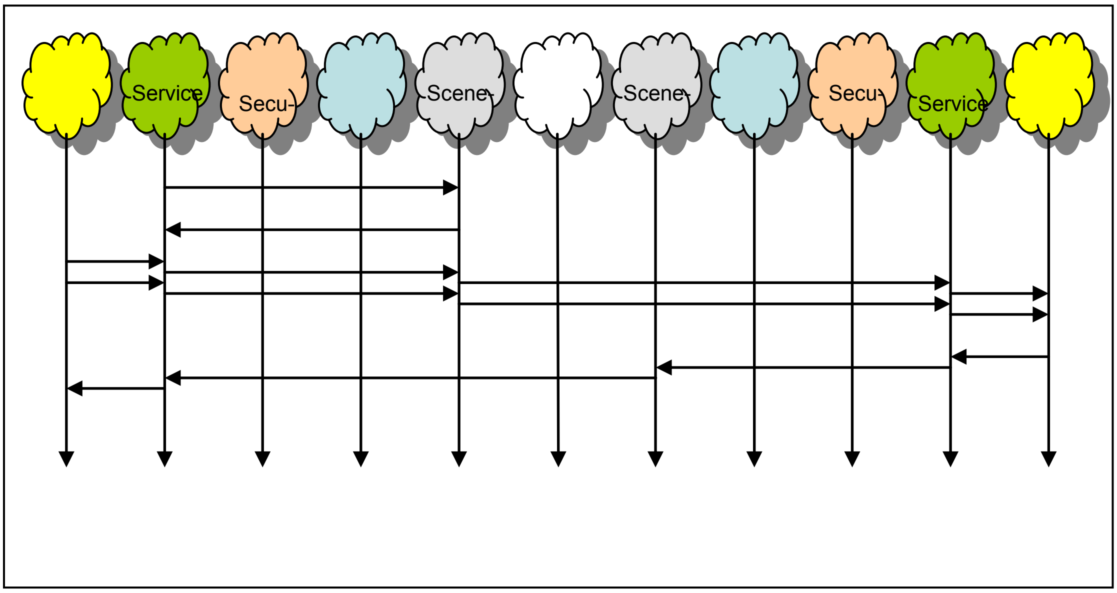


Figure 12-1. Change of object focus

REFERENCES

[3] OTA – over the air

[SAL05] Salater, S., Video Object Resolution Controlled by Eye Tracking. Master's Thesis, NTNU 2005.

[PEL05] Peli, E., Goldstein, R. B., Woods, R. L., (2005). Scanpaths of motion sequences: where people look when watching movies. *Starkfest IV Conference on Vision and Movement in Men and Machines*, UC Berkeley, CA, 27-28 Mai. URL: http://www.eri.harvard.edu/faculty/peli/papers/Starkfest05_10d.pdf

[JEN10] Jenssen, T S. "Rekonstruksjon av objektbaserte scener ved hjelp av interpolasjon i DMP". Master's thesis, Item, NTNU 2010.

PART 13: QUALITY SHAPING

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

Quality Shaping is a quality control scheme used in DMP networks that allows the quality of scenes to vary with the load on the network, but guarantees a minimum quality of scenes. The prerequisites for Quality Shaping are presented in Parts 1 to 12.

RELATED WORK

Traffic Shaping, the concept that QualityShaping builds upon, was first proposed in [LAR 82], where variance reduction of traffic streams by queuing and dropping were evalutated. A few years later traffic shaping was further developed by the 'ATM world', and the Leaky Bucket [TUR86] and Token Bucket flow-control schemes were proposed.

A large number of traffic control algorithms have been proposed. The selection below was largely developed in the 1990s and 2000s.

ABE – Alternate Best Effort, is a queuing scheduling proposal that trades loss for latency in giving fairness to links carrying both TCP and UDP traffic (HUR01).

ECN – Explicit Congestion Notation, packets are marked instead of dropped. The destination node signal the marking back to the source which acts accordingly (FLO97), (RAM01).

AIMD – Additive Increase, Multiplicative Increase, the control method used in traditional TCP. A stochastic analysis is presented in (DUM02).

TFRC - TCP Friendly Rate Control, gives TCP friendliness in the long run, but the variance of data rate is less than for AIMD (HAN03).

DCCP – Data Congestion Control Protocol, can switch between AIMD and TFRC. DCCP is connection oriented and runs on top of UDP (KOH04).

RED – Random Early Detection, incoming packets are randomly dropped with a probability based on a cost function of average queue length (RED04).

REM – Random Exponential Marking (ATH01).

AQM – Active Queue Management, regulators applied in routers and terminals in packet networks with mixed TCP and RTP/UDP traffic, in order to obtain fairness (LIE04a), (LIE04b), (KIM03), (HOL01).

XCP is new transport protocol described by (KAT02) which generalizes the ECN, and decouples utilization control from fairness control. Instability problems are avoided.

DMP GUARANTEES END-TO-END TIME DELAY

It is easily shown that the end-to-end time delay for AV packets can be guaranteed, and that the propagation delay is the limiting factor. Assume a propagation delay between two European network nodes of d_i ms, and a network hierarchy of nodes giving maximum l links between any two users. Assume a constant packet length and a constant link capacity, that gives a constant delay h_i for sending a packet. There is a single queue with limited waiting room N_{1i} for each link, see Figure 8-3. If the queue is full when a packet arrives, the packet is dropped. When the last packet in the queue arrives, it has to wait until all packets in the queue are sent out plus the remaining service time of the packet being sent out. The maximum remaining service time is h_i (uniformly distributed (0, h_i)). The total maximum delay is then the sum $\sum_{l}(N_{1i} * h_i + h_i + d_i)$. The design question is then to decide N_i and the number of parallel links, in order to obtain an acceptable time delay and a minimum time varying resolution, when the input traffic varies.

THE QUALITY SHAPING SCHEME

The quality of a scene can be changed in many ways. Possible parameters that can be varied are described under the section QualityShapingProfiles.

All nodes, AccessNodes included, perform selective packet dropping (from sub-objects) based on agreed QualityShapingProfiles.

Only the user can change the content within packets, also according to QualityShapingProfiles.

Quality Shaping is not based on individual streams, but on the aggregated packet streams from each source AccessNode loading a network node. Many AccessNodes (maximum approximately 2 millions) normally contribute to the packet stream into an output link of a network node. The 80 msbs of the user's IP address uniquely identifies the AccessNode he is connected to. AccessNodes intitiate QualityShaping packets at certain intervals (say 5 ms, an important configuration and adaptation parameter). An AccessNode receiving packets from other AccessNodes sends QualityShaping packets to those AccessNodes, but not to others.

After a collaboration has been set up, the users start shooting scenes and sending AppTraNetLFC packets. The packet header parameter PacketRate, indicates the scene sub-object's packet rate, and when the rate is to be changed. It is indicated by sending a packet of type, e.g., PT = 'Visual, change of rate' and a Timestamp value that defines the time until the rate change actually takes place. The QualityControl of the AccessNode measures the overall packet rate of each scene, and take action (TBD) if it is different from that indicated by the scene's sub-object packets (summed up using the SceneProfile and QualityShapingProfile).

Nodes regularly forward QualityShaping packets (AppTraNetLFC packets with parameter PT = 'QualityShaping') to source AccessNodes, reporting packet arrival count, drop cont, departure count, output link capacity, mean length of control packet queue Q_{2i} , and other measurements in the interval. All nodes in the established routes check the Payload type of all packets, and if it is a QualityShaping packet, it is delayed shortly allowing the nodes to write their measurements into the packet body. When the packet arrives at the source AccessNode, the QualityControl analyses the measurements, and take the decision to start dropping user packets or not. Normally several users contribute to the overload in the remote node, and the QualityControl/ServiceControl sends new QualityShaping packets to all these users. It is then up to the users to do nothing or change parameters according to the QualityShapeProfile.

Each AccessNode includes a QualityShaper that receives QualityShaping packets from network nodes, and that actually scales the packet rates from users by controlled dropping from sub-objects. See Figure 13-1.

Figure 13-1 shows an example where a CityNode C (Trondheim), measures the traffic load generated by three different sources F, Fa and Fb, that send packets to Fc via C. Packets from F use about 30 ms, from Fa about 12 ms, and from Fb about 1 ms, when there is no queuing and only the propagation delays contribute to the delay.

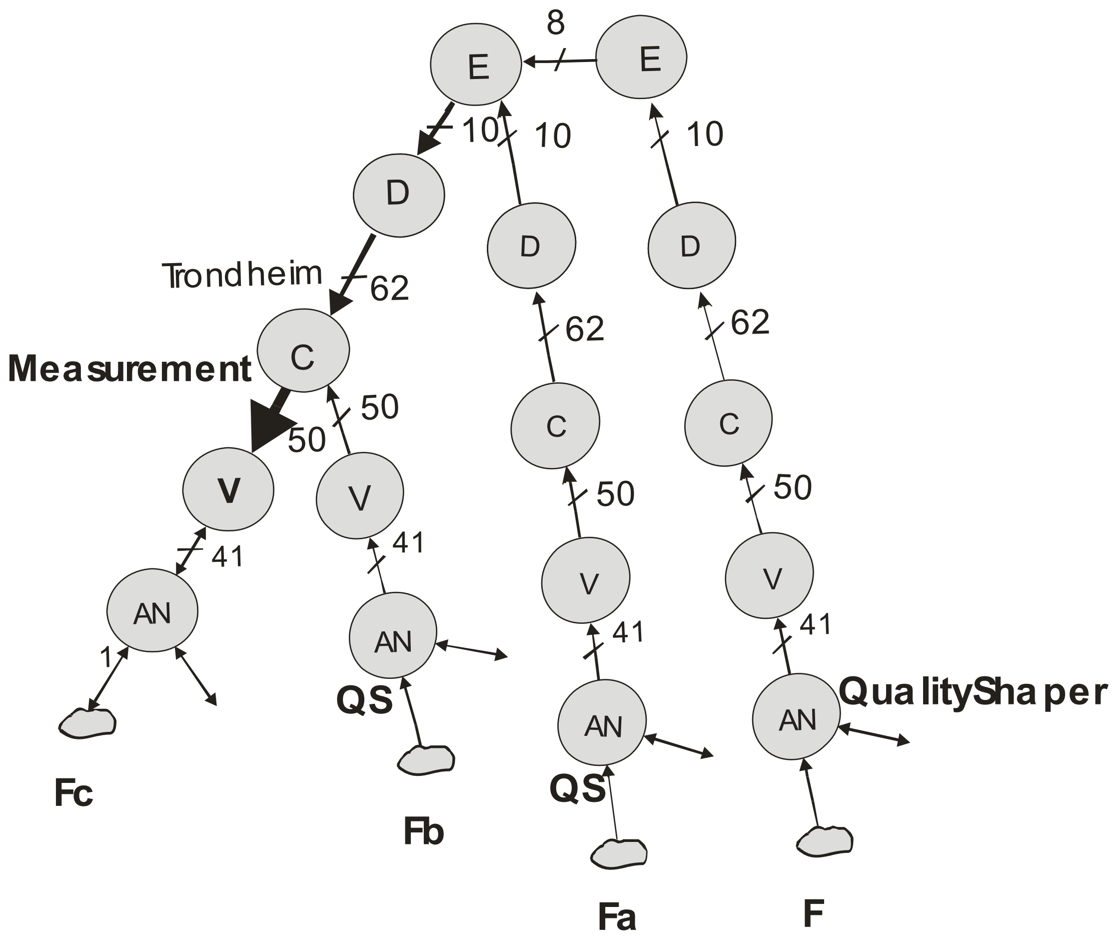


Figure 13-1. Three paths, 30ms, 12ms and 1 ms propagation delay.

Figure 13-2 shows some sequences of packets regarding shaping. At a certain time the Trondheim C node in the network is overloaded, and the node starts dropping packets of low priority, given by the packet Sequence number. The actual down- and up-scaling can be handled as a feedback control problem (see section QualityShaping and Feedback Control). At the end of a measurement interval, it sends a QualityShaping packet to the ServiceControl in AccessNodes of F, Fa and Fb. The ServiceControl immediately starts dropping packets of low priority in order to decrease the traffic, and forwards QualityShaping packets to the users F, Fa and Fb. Users are generally allowed to ignore such indications, but may optimize their parameters according to their QualityShapingProfile.

Assume that after a while, the destination AccessNode for user Fb is saturated, and the ServiceControl serving Fb sends a QualityShaping packet to the ServiceControl serving Fa. In this case user Fa wants to change his parameters according to his QualityShaping Profile, so user Fa sends a packet indicating by the PacketRate parameter that he intends to change the packet rate at the time given by the Timestamp.

Security handling of the packets are not shown in the Figure 13-2, but is similar to that in Figure 8-4.

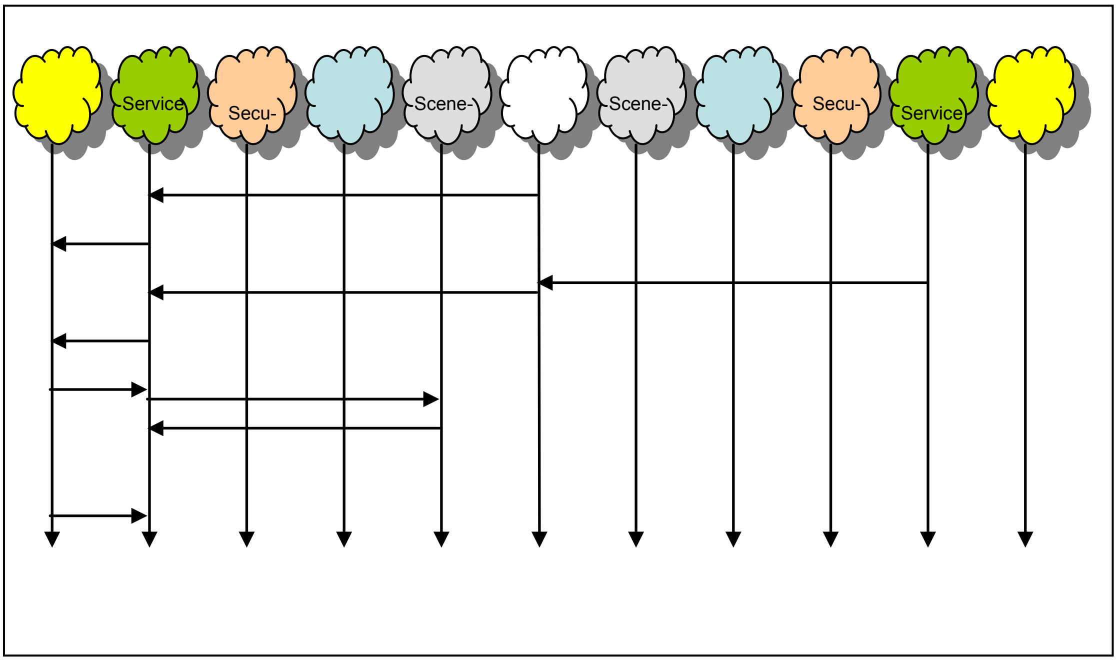


Figure 13-2. QualityShaping sequences.

Packets *Description*

8a Node C sends QualityShaping packet to ServiceControl for user Fa
8b After started dropping packets, ServiceControl for Fa sends QualityShaping packets to all active users under this AccessNode
8c,d The ServiceControl in AccessNode for user Fb sends a QualityShaping packet to the ServiceControl in AccessNode for user Fa
8e ServiceControl for user Fa sends QualityShaping packets to all active users under it's AccessNode
9a-c User sends QualityShaping packet to ServiceControl to indicate that he will change his packet rates at a given time
10 User sends audiovisual packets to the ServiceControl, with the changed packet rates

Alternate routing is discussed in Part 7. Statistical loadsharing is used, and is necessary when each node sharing a load reports its load to AccessNodes independently.

QUALITYSHAPINGPROFILES

A QualityShapingProfile includes several classes of objects with a predefined value-set of quality parameters, which can vary as shown below.

The QualityShapingProfiles are based on parameters describing temporal and spatial scene resolution and composition. The following parameters are considered:

- The number of sub-scenes of a scene
- The number of objects per sub-scene
- The number of stereoscopic views per object
- The number of sub-objects per object
- The up-date rate of each sub-object
- The components representing the sub-object (e.g., RGB,,,,)
- The Alpha channel depth (e.g., RGBA)
- The sample-depth of each components (e.g., RGBA – [16,16,16,16])
- The sampling rate of each component (e.g.,YCrCb – 4:2:2)
- The shape and size of the sub-object, shape mask
- The compression and coding scheme for sub-objects
    - NOC. The priority of packets representing the sub-objects is equal. But to drop from selected sub-objects, the packet Sequence number is used as described in Part 8.
    - Mod JPEG2000. The parameters to vary here are the quantization steps and the number of layers. 20 layers is maximum, typically 10-12 are used. The layering is progressive, and packets representing highest layers are dropped first. The packets are dependent. The packet Sequence number is used as described in Part 8.

A background object can be shot live, shot and stored, or synthesized. The quality requirements are not as high as for important objects. The modified JPEG2000 compression can be applied for the less important objects. For important objects like faces, the loss-less and graceful degradable compression algorithm NOC, is very well suited.

The data rate from a sub-object can be calculated as follows:

$$SO_r = \sum_{components} \text{(\#pixels of object)/(\#sub-objects))*(\#views)*(\#bits per pixel)* (update rate)/(compression)}$$

Using NOC, the RGB components are treated equally, while the Alpha is treated differently. For YCrCb, the Y component has different #pixels per object and different update rate, than Cr and Cb. Cr and Cb are treated equally.

OBJECT CHARACTERISTICS

Background object

YC_rC_b - [12,10,10] bits, 4:2:2 sampling, adaptable, corresponding to [12,10,10] – [10,8,8] layers. The quantization steps can be varied as well, to obtain a compression ratio variation 100 – 50.

8 sub-objects, adaptable to 4 sub-objects

pixel size 0.5 x 0.5 mm

3 views, adaptable 3, 2, 1 view

Compression 100 – 50 x, using Mod JPEG2000

Update every 500 ms over air (5 ms in receiver)

Shape and size, rectangle 3 m wide and 2 m high

compressed data rate, bps

object size, mm2	6000000	
pixel size, mm2	0,25	
#pixels per object		24000000
#sub-objects		8
#views		3
#bits per pixel		12
update rate		2
compression ratio		100
#components, YCrCb, shape		2,2
Sub-object bitrate		4752000
Object bitrate		38016000

Note that the #components is set to 2.2. The Y component is assumed to contribute with '1.0', the Cr and Cb components to contribute with '0.5' each, and the shape mask with 0.2.

Important object - Body Object moves around

The SSOMC system tracks the object

RGBA - [16,16,16,16] bits, 4:4:4 sampling, adaptable to [10, 10, 10, 10]

8 sub-objects, adaptable to 4 sub-objects

pixel size 0.25 x 0.25 mm

6 - 3 views (adaptive)

Compression 5x, NOC

update 10 ms over air (5 ms in the receiver), adaptable to 20 ms

shooting size 2 x 1.2 m, body mask = 720000 mm

Compressed data rate, max quality and size

object size, mm2	720000	
pixel size, mm2	0,0625	
#pixels per object		11520000
#sub-objects		8
#views		6
#bits per pixel		16
update rate		100
compression ratio		5
#components		4,1
Sub-object bitrate		1,134E+10
Object bitrate		9,069E+10

With packet length 1500 bytes, the packet rate becomes 0.76 Mpps (Mega packets per second)

Very important object - Face Object

The SSOMC system tracks the object

RGBA - [18,18,18,18] bits, 4:4:4 sampling, adaptive to [12, 12, 12, 12]

16 sub-objects, adaptive to 8 sub-objects

pixel size 0.25 x 0.25 mm

18 - 3 views (adaptive)

Compression 5x, NOC

update 5 ms over air (5ms in the receiver), adaptive to 10 ms

Shooting size 400 x 300 mm, face mask = 90000 mm

Compressed data rate, max quality

object size, mm2	120000	
pixel size, mm2	0.0625	
#pixels per object		1920000
#sub-objects		16
#views		18
#bits per pixel		18
update rate		200
compression ratio		5
#components		4.1
Sub-object bitrate		6.376E+09
Object bitrate		1.02E+11

If the packet length is 1500 bytes, then the packet rate is 0.85 Mpps (Mega packets per second).

Very important object - Audio object

The quality of the audio is important, especially the delay when musicians are performing over a network. The delay should ideally not exceed 20 ms, but can in a DMP European network be about 34 ms due to propagation (The importance of the delay is planned to be further tested). The scheme used here is 7.1 stereo, no compression, scalable 24 bits resolution and 96 kHz sampling. This gives a data rate of about 20 Mbps before any error control or overhead is added. The object can be scaled down to about 7 Mbps (TBD), without loosing too much quality (48 kHz sampling, 16 bits resolution, temporarily)

EXAMPLE OF QUALITYSHAPINGPROFILE

When an object moves fast (> x meters per sec), then the ShapeProfile first get reduced spatial resolution, then temporal resolution. For slow moving objects (< x meters per sec), the temporal resolution is reduced before spatial resolution. The background object already has a moderate data rate, so reducing the quality gives only a small change of the total data rate. The quality of very important objects should not be 'touched' before it is absolutely necessary. The important objects, however, have a large quality reduction potential - when applied it will be noticeable, but not critical.

The QualityShapingProfile has the following format:

QualityShapingProfile { Id,

CompressionScheme [type, compressionRatio,

#components, (bit1, bit2,,,bitN),

(update1, update2,,,,updateN)]

#subObjects, pixelSize (x,y), #views, maskSize }

The parameters have the following description:

Table 13-1

Parameter	Type	Value	Description
Id		1	Highest quality
		2	Next highest quality
		L	Lowest quality
CompressionSceme	NOC1	1	Lossless
	ModJPEG2000	5	Lossy
#components		1-64	Number of components
bit1,,,,bitN		1-24	Number of bits per component
update1, updateN		< 200	Update rate of each component
#subObjects		2-25	The number of sub-objects
#views		1- m	The number of stereoscopic views
maskSize	Any shape, mm2		The area of the shape mask

Example of fast moving, very important object

ShapeProfile {1, [1, 5, 4, (18, 18, 18, 18), (200, 200,200, 200)],

9, (0.25, 0.25), 18, <x>}

5.6 % reducution of data rate:

Remember that AccessNodes only can drop packets, not change content of packets. If each bit layer of each sub-object is put into different packets, then the following reduction can be performed by the AccessNode, otherwise it has to be performed by the users.

ShapeProfile {2, [1, 5, 4, (17, 17, 17, 17), (200, 200,200, 200)],

9, (0.25, 0.25), 18, <x>}

,,,,,,,

22 % reduction of data rate from Id = 1:

ShapeProfile {3, [1, 5, 4, (16, 16, 16, 16), (200, 200,200, 200)],
9, (0.25, 0.25), 16, <x>}

,,,,,,

ShapeProfile {4, [1, 5, 4, (14, 14, 14, 14), (200, 200,200, 200)],
4, (0.25, 0.25), 16, <x>}

,,,,,,,

ShapeProfile {i, [1, 5, 4, (12, 12, 12, 12), (200, 200,200, 200)],
4, (0.25, 0.25), 16, <x>}

,,,,,,,

ShapeProfile {j, [1, 5, 4, (12, 12, 12, 12), (200, 200,200, 200)],
4, (0.25, 0.25), 3, <x>}

1	2	x	1	2	x
x	5	6	x	5	6
7	x	9	7	x	9
1	2	x	1	2	x
x	5	6	x	5	6
7	x	9	7	x	9

Figure 13-3. 9 sub-objects, 3, 4 and 8 dropped

Figure 13-3 shows 9 sub-objects where three sub-objects (3, 4 and 8) are dropped. The missing values x can be found by interpolation of the four surrounding pixels (the traffic is reduced by 33%).

TRAFFIC MEASUREMENTS

The QualityShaping scheme includes measurements that have to be implemented in all nodes. The measurements are performed each 5 milliseconds typically, which is small compared to maximum RTD (round trip delay, > 60 ms), and in the same range as the limits for human perception of fast changes:

We consider three traffic source AccessNodes, and a node C with an output link queuing system as shown in Figure 8-3.

In each time interval Δt (typically 10.0&-6 seconds) the QueueManagement does:

Count and record the number of arrivals ArrC, ArrCF, ArrCFa, ArrCFb, dropped packets DropC, and departures DepC. C denotes the node, F, Fa and Fb are the traffic sources (users).

Record the queue lengths contribution to Q_{1C} and Q_{2C} from all AccessNodes providing traffic to the node C. The 80 msbs of the source IP address provide unique AcessNode addresses.

Record current time, time since previous QualityShaping packet, and the current link capacity LinkCapC.

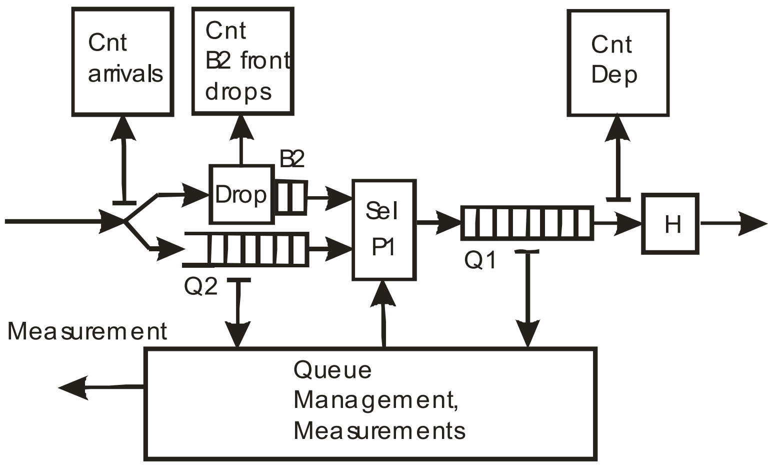


Figure 13-4. Traffic measurements in nodes

A measurement entity, Queue Management and Measurement, is activated after a number of intervals given by a variable CTick, typically 5.0&-3 seconds for normal situation, and 1.0&-3 seconds when packet drop starts, to read the counters, and to forward QualityShaping packets from node C to the access nodes loading node C.

The DEMOS code for the entity can be as follows:

```
entity class MeasureC(CTick);
real CTick;
begin
  long real PrevTime, LastSent;
  integer Help1, Help10, Help10a, Help10b,
    Help2, Help3,
    Help4, Help40, Help40a, Help40b,
    Help5, Help6;
  boolean ShortPer;
  ShortPer := false;

  Help1 := Help2 := Help3 := 0;
  Help10 := Help10a := Help10b;
  Help4 := Help5 := Help6 := 0;
  Help40 := Help40a := Help40b;
  LastSent := 0.0;
  PrevTime := 0.0;

loop:
  hold(CTick);

  if DropC > Help2 then
  begin
    ShortPer := true;
  end;

  if ShortPer then
  begin
   if LastSent >= 1.0&-3 then
   begin
     new ShapeMsg("ShapeMsg",time, time - PrevTime,
      QueLengthC, ArrC - Help4, ArrCF - Help40,
          ArrCFa - Help40a, ArrCFb - Help40b,
          DropC - Help5,DepC - Help6, LinkCapC).schedule(0.0);

     Help4 := ArrC;
     Help40 := ArrCF;
     Help40a := ArrCFa;
     Help40b := ArrCFb;
     Help5 := DropC;
     Help6 := DepC;
     LastSent := 0.0;
     PrevTime := time;
   end;
   ShortPer := false;
```

```
    end else

    if LastSent >= 5.0&-3 then
    begin
      new ShapeMsg("ShapeMsg",time, time - PrevTime,
        QueLengthC, ArrC - Help4, ArrCF - Help40,
              ArrCFa - Help40a, ArrCFb - Help40b,
              DropC - Help5,DepC - Help6, LinkCapC).schedule(0.0);
      Help4 := ArrC;
      Help40 := ArrCF;
      Help40a := ArrCFa;
      Help40b := ArrCFb;
      Help5 := DropC;
      Help6 := DepC;

      PrevTime := time;
      LastSent := 0.0;

    end;

    LastSent := LastSent + CTick;

    Help1 := ArrC;
    Help10 := ArrCF;
    Help10a := ArrCFa;
    Help10b := ArrCFb;
    Help2 := DropC;
    Help3 := DepC;

    repeat;

  end***MeasureC***;
```

ADMISSION CONTROL

Based on the measurements in the network nodes over long time, and reported back to AccessNodes, the load (say a few hundred milliseconds ahead) on nodes in a given path can be predicted (TBD). This information will be used when the negotiation of SceneProfile and QualityShapingProfile starts when establishing a multiparty DMP (see Figure 8-4). If one or more nodes in the path are heavily loaded, the request for setup of collaboration will be rejected. In this way, a minimum scene quality can be guaranteed for the ongoing collaborations.

QUALITY SHAPING AND FEEDBACK CONTROL

The network part of the Quality Shaping scheme includes two main modules, the queuing system including the Queue Management and Measurement in all nodes, see Figure 13-4, and the queuing system with the QualityShaper in AccessNodes, see Figure 13-5.

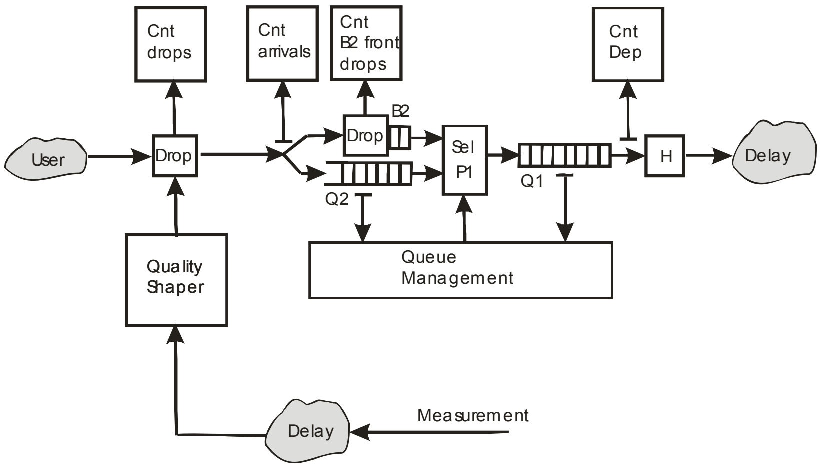


Figure 13-5. QualityShaping in AccessNodes

If we consider only one path through the network and only the path for AV packets (via B2 in Figure 13-4 and 13-5), the QualityShaping system can be modeled as a Feedback Control System. Older control models using descrete PID, P or PI regulators [Wik07], or newer models such as 'receding horizon' or 'moving horizon' models from predictive control [MAC02] can be applied.

The paper [RON05] presents a PI regulater based simulation model in DEMOS.

REFERENCES

[ATH01] S. Athuraliya et. al., REM: Active Queue Management, *IEEE Network, 15*, Issue: 3, May-June 2001.

[DUM02] Vincent Dumas, Fabrice Guillemin, and Philippe Robert. A Markovian analysis of additive-increase multiplicative-decrease algorithms Source: Adv. in Appl. Probab. 34 (2002), no. 1, 85–111

[FLO97] Floyd, V. Jacobson, Random Early Detection gateways for congestion avoidance, *IEEE ACM Transactions on Networking, 1*(4), August 1997.

[HAN03] Handley, M., Floyd, S., Pahdye, J., and Widmer, J. RFC 3448, Proposed Standard, January 2003. TCP Friendly Rate Control (TFRC): Protocol Specification.

[HOL01] C. V. Hollot, et. al., On Designing Improved Controllers for AQM Routers Supporting TCP Flows, *Proc. of IEEE Infocom,* 2001.

[HUR01] P Hurley e.al, in IEEE Network, vol 15, issue 3, May-June 2001.

[KOH04] Eddie Kohler, Mark Handley, Sally Floyd, Jitendra Padhye Datagram Congestion Control Protocol (DCCP) ICIR

[KIM03] K. B. Kim, S. H. Low, Analysis and Design of AQM based on State-Space Models for Stabilizing TCP, *Proc. of theAmerican Control Conference 2003,* Volume: 1, 4-6 June 2003.

[LIE04a] A. Lie, O. M. Aamo, L. A. Rønningen, On the use of classical control system based AQM for rate adaptive streaming media, *Proc. of 17th Nordic Teletraffic Seminar,* Norway, August 2004.

[LIE04b] Arne Lie, Aamo, L A Rønningen. OPTIMIZATION OF ACTIVE QUEUE MANAGEMENT BASED ON PROPORTIONAL CONTROL SYSTEM. CIIT, Virgin Islands, Nov 2004.

[MAC02] J M Maciejowski. Predictive Control with Constraints. Pearsons Education/ Prentice Hall 2002. ISBN 0-201-39823-0.

[RAM01] K. Ramakrishnan, S. Floyd, D. Black, The Addition of Explicit Congestion Notification (ECN) to IP, *IETF RFC3168,* September 2001.

[RON82] L A Rønningen. Input Traffic Shaping, Proceedings of the International Teletraffic Congress, Montreal 1982.

[RON05] Rønningen, Leif Arne. Adaptive Scene and Traffic Control in DMP . IADAT International Conference on Telecommunications and Computer Networks; 07.09.2005 - 09.09.2005. Published in: *Proceedings IADAT-tcn2005*; 2005

[WIK07] http://en.wikipedia.org/wiki/PID_controller

[TUR86] Turner, J.S. 1986. New directions in communications (or which way to the information age?). IEEE Commun. Mag. 25, 10: 8-15.

PART 15: TRANSIENT TRAFFIC FROM DMP SCENES

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

INTRODUCTION

This part treats traffic generation in DMP collaborations. DMP includes virtual collaborations in e.g., games, business, concerts, opera, theatre, education, telemedicine, and future television, and this developing technology could be in practical use before 2017. Certain games have severe delay and temporal resolution requirements, while surgery requires very high spatial resolution. More details can be found in Parts 1 – 14.

STOCHASTIC PROCESSES

The audiovisual traffic sources of DMP appear to be non-linear and transient. This means that established stochastic process theory assuming stationary processes cannot be applied. See for example [JON01] for an introduction to stochastic processes. However, aggregated control packet streams may be modeled using established queuing theory [IVE07], assuming stationary processes. Only audiovisual traffic will be treated in this Part.

Following each audiovisual packet through the network is of less interest, since the minimum and maximum user-to-user delays (jitter) are given by the path through the network. Assuming the maximum jitter is 10 milliseconds, the variation is not perceivable on the display. Of interest is the distribution of drop occurrences vs time interval vs drop rate, see paper [RON07]. The perception of video resolution following such drop distributions will be tested in the Caruso lab in the autumn 2007.

Seen from the sources, the packet rate typically can be constant for a long time, and then instantly step to another constant value. For a packet stream from a sub-object, the time between each rate step shows random nature, but the distribution is presently unknown. Manual inspection of some cases suggests that the uniform pdf (probability density functions) can be used as an approximation. Then, if a large number of independent streams are merged, one can argue that the negative exponential distribution is a reasonable approximation to the (unknown) distribution of time between change of packet rate (See Palm-Kintchine limit theorem [CIN72]).

TRAFFIC GENERATION

Several traffic sources are described here, the 'Virtual Dinner', the 'Virtual Song Lesson', an Existing Movie, a Futuristic Movie which is a distributed game with users participating in real time as actors in the movie, and a more general StepRate Generator. The sources have transient and non-linear behaviors, and therefore stochastic theory that require stationary processes, cannot be applied. However, we are not generally interested in mean values, but rather transient slopes and their variations, and extreme values and their duration. Sources have strong internal dependencies in their packet streams, but are in some cases independent of each other. However, collaborations are example where the sources are highly dependent.

Typical for the sources is the extreme variability of packet generation rates, which can be regarded constant in time intervals, where the time intervals are approximately uniformly distributed. The packet rate can vary from a few kilo packets per second (kpps) to several Mega packets per second (Mpps) (the packet length is 1500 bytes). The sources have different characteristics as described below.

In order to obtain near-natural feel of the content, an important object (human body, etc) should meet the following (transmitted over network) requirements

- YC_rC_b – [16, 12, 12] bits resolution, 4:2:2 sampling, adaptable [12,10,10] to [16,12,12]
- 9 sub-objects per object, pixel size 0.25 x 0.25 mm, 1 to 8 sub-objects are droppable
- 5 views, adaptive 1 to 5 views (determined by the number of viewers at receiving end)
- Compression, 5 times, lossless NOC scheme
- 5 ms temporal update over network, adaptive 5 to 10 ms

If the size of the important object is 600 x 900 mm, and the packet size is 12 kbits, the compressed data rate from the object is about 2 Mpps (25 Gbps). However, the scene resolution is adaptive and may be scaled down so the important object generates less than 20 kpps for rare and short periods, with substantial quality reduction (admission control assures that periods with lower packet rate than say 500 kpps are shorter than say 500 ms, TBD). Note that the adaptive scaling of the scene takes place in the network access node, regulated by a contract between the network owner (service provider) and the user (users that somehow try to use the network in an unfair way are stopped by the access node).

THE SONG LESSON GENERATOR

As an example of a virtual song lesson we may think of a song teacher living in Padova giving song lessons to a student in Trondheim over a network. The main requirement to the collaboration is the stereoscopic multiview near-natural feel. This includes an end-to-end delay minimum 22 ms (the propagation delay), maximum 34 ms (queuing time added). Experiments have indicated [CHA04] that audio delays of 11-12 ms are ideal. The backgrounds of the scenes are less important in this case, while the quality of the display of faces and the sound reproduction are critical. The bodies of the singers are also important. Note that as these requirements may seem extreme (hardly realizable) today, they are expected to be normal in say 10 years.

SCENE OBJECT CHARACTERISTICS

The case described below is based on a video shot of a simulated DMP song lesson in the Caruso Lab [RON04]. A standard DV camera was used, and the scenes were off-line analyzed by manual inspection. Dominique Guyot is the song teacher, while the author is the song student. The song lesson scene does not have fast movements, but it is important that the temporal and spatial resolution is high for the whole body of both the teacher and the student (TBD). The audio (including guaranteed delay) is also an important object. The background object is less important. The data rate changes dramatically in steps, but is stable for minutes.

The following requirements have been assumed for the scene objects:

Background object

YC_rC_b - [12,10,10] bits, 4:2:2 sampling

9 sub-objects, pixel size 0.5 x 0.5 mm

3 views (adaptive)

Compression 100x, JPEG2000

Update every 500 ms over air (5 ms in receiver)

size 2400 x 1200 mm

Uncompressed 1.5 Gbps, compressed 15 Mbps

Body Object moves sideways

The 'system' tracks the object

YC_rC_b - [16,12,12] bits, 4:2:2 sampling

9 sub-objects, pixel size 0.25 x 0.25 mm

3 views (adaptive)

Compression 5x, NOC

update 10 ms over air (5 ms in the receiver)

Size 600 x 900 mm

Compressed 7.25 Gbps

Face Object

The 'system' tracks the object

YC_rC_b - [16,12,12] bits, 4:2:2 sampling

16 sub-objects, pixel size 0.25 x 0.25 mm

3 views (adaptive)

Compression 5x, NOC

update 5 ms over air (5ms in the receiver)

Size 200 x 300 mm

Compressed 3.22 Gbps

Audio object

The quality of the audio is important, especially the delay when two singers are performing a duet over a network. The delay should ideally not exceed 20 ms, but is 24-34 ms in our case. (The importance of the delay is planned to be further tested). The scheme used here is stereo, no compression, and scalable 20 bits resolution and 96 kHz sampling. This gives a data rate of 3.84 Mbps before any error control or overhead is added.

ADAPTIVE RESOLUTION & VARYING TRAFFIC

When the traffic in the network varies and nodes are saturated, the Quality Shaping algorithm assures that packets are dropped in a controlled manner, and that traffic sources after a delay are scaled according to the traffic load. In the song lesson case, the data rate of the background object is relatively low, so a reduction counts less. The sound and the face objects are so important that reducing their quality should be the last thing to do. However, the body object can be made adaptable, and the perceived quality variation is probably hardly noticeable (this has not been tested yet). All object parameters listed can in principle be varied, but the network can only perform controlled dropping of sub-objects. Users can vary all parameters. If the number of sub-objects is reduced from 9 to 5, and the update time 'over the air' is increased from 10 to 20 ms, the data rate is reduced from 7 Gbps to 2 Gbps. The total data rate is then reduced from about 11 Gbps to 6 Gbps. The duration of the overload situation may vary from tenths of milliseconds to seconds, depending on the characteristics of the traffic mix in the network, and by the selected Quality Shaping parameters. The reduction of spatial and temporal resolution of the body object will be noticeable only when the eyes of the singers focus on the bodies, not else.

TYPICAL TRAFFIC GENERATED BY THE SONG TEACHER SCENE

The traffic generated by the song teacher's scene is first assumed to vary with full resolution of all objects, and then with a quality reduction for the body object (down to 2 Gbps) lasting for twenty seconds.

Sequence with temporal and spatial resolution reduction:

Time (min:sec)	Behavior
00:00	Student enters his room
00:20	Teacher enters her room, orders the 'System' to establish collaboration
00:25	Collaboration set-up

00:30	‘System’ has identified and tracks three objects in each room, background, body, face
00:40	Student and teacher ‘say hello’
01:00	Teacher demonstrates technical exercises
01:05	The Quality Shaping algorithm reduces the teacher’s body object resolution (data rate down to 2 Gbps)
01:25	The Quality Shaping algorithm increases the teacher’s body object resolution to maximum
02:00	Student practices exercises

,,

20:00	Teacher and student leave their rooms
30:00	Teacher and student enter their rooms
35:00	Student sings a song, the teacher’s body resolution is reduced
45:00	Collaboration released

A not-to-scale approximated diagram for the traffic generated by the teacher’s scene during this sequence is shown in Figure 15-1. Note that the generated traffic pattern is characterized by a minimum transmission data rate of about 20 Mbps for five minutes, while the maximum data rate is 10.5 Gbps for minutes. For two periods the data rate is 5.1 Gbps.

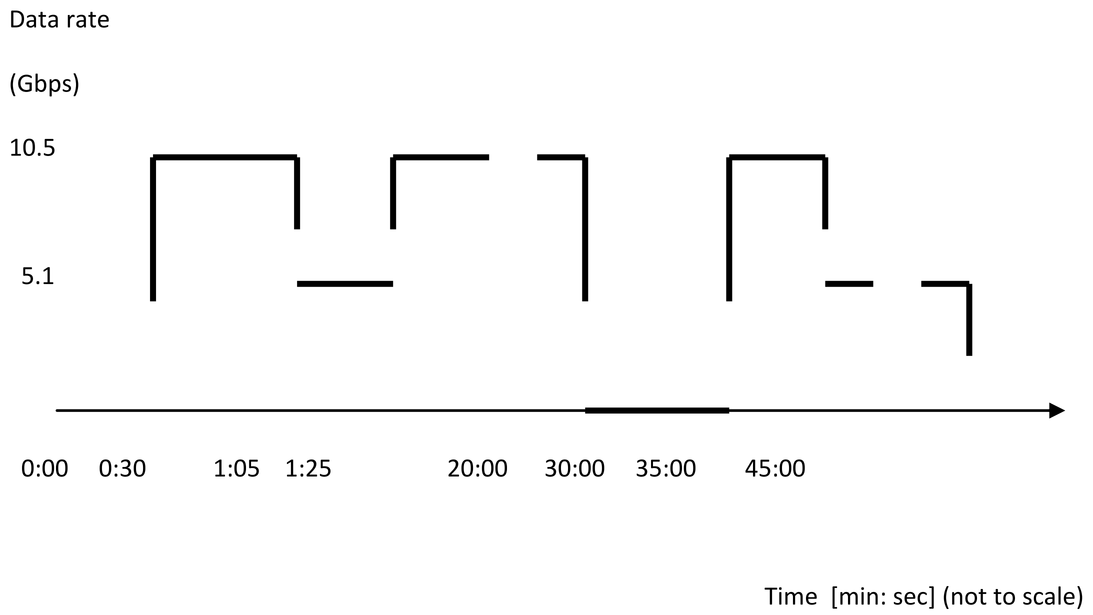


Figure 15-1. The data rate of the stream transmitted from the teacher’s scene

Figure 15-2 shows the packet rate [Mpps] vs a correct time scale.

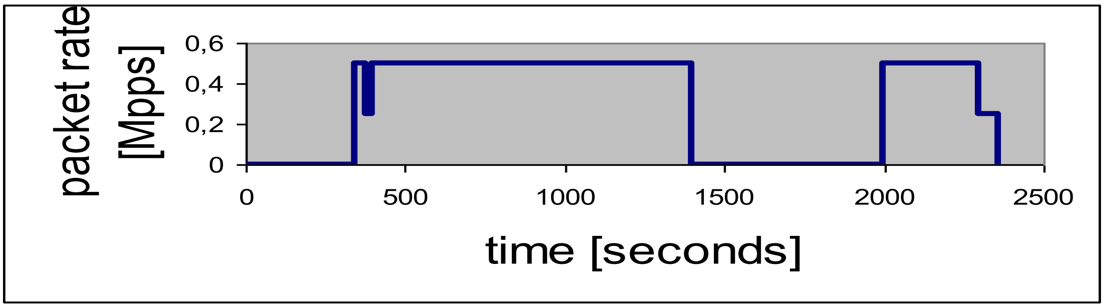


Figure 15-2. The data rate of the stream transmitted from the teacher's scene (to scale). [Mpps]

THE VIRTUAL DINNER GENERATOR

In the Virtual dinner case, the packet rate is constant for minutes, but changes rapidly for short periods of 10-15 seconds when a person stands up and leaves the room, and when he comes back and sits down. The Virtual Dinner model was first presented in [RON04b].

It is always hard to predict what user requirements will be ten years ahead (history shows), but the perceived naturalness should be dramatically improved compared to existing systems (HTDV in 2D, RAVE in 3D). The Hems Lab in Part 5 indicates what the requirements could be.

THE VIRTUAL DINNER

The Virtual Dinner service allows colleagues, friends, family members to have dinner together, even if they are not in the same city. This is a type of enhanced video conference, where the requirements of naturalness are severe. Consider a situation where three researchers have dinner and discuss research problems. One researcher lives in Trondheim, Norway, while the other two live in Padova, Italy and Poznan, Poland. Generally, there can be more than one person in all places. Persons may come in and go out of the scene, or walk around the room, but will most of the time sit or stand, talking. The faces and hands of the persons talking are regarded important objects. From time to time, the whole body may be an important object. The food and plates are also important objects. There may be other important objects as well, a painting on the wall, a dog, etc. The dining rooms are background scenes.

The behavior of the scenes may be as follows: Researcher A in Trondheim comes into his dining room, sits on the sofa and requests a Virtual Dinner with researcher B in Padova, also sitting on his sofa. The system finds two faces and a plate with food researcher B. Researcher A walk out for a plate of food , comes back, and sits down. The system finds the plate and the food. They start eating and talking. Researcher B stands up and walks sideways in the room. After a few minutes they need to talk to researcher C in Poznan, and set up a three party DMP. After eating, B leaves the room. A asks the system to disconnect B. Researcher C has to leave home and go to his office. But he wants to continue the session with A while traveling to the office.

CHARACTERISTICS OF OBJECTS OF THE VIRTUAL DINNER

When the researchers are at home the objects of the scenes can be characterized as

follows:

Background objects

YUV - [12,10,10] bits, 4:2:2 sampling

9 sub-objects, pixel size 0.25 x 0.25 mm

3 views (adaptive)

Compression 100x, JPEG2000

Update every 500 ms over air (5 ms in receiver)

size 3000 x 2000 mm

Uncompressed 12.7 Gbps, compressed 127 Mbps

Body Object1 approaches fast towards dedicated cameras

The 'system' tracks the object

YUV - [16,12,12] bits, 4:2:2 sampling

16 sub-objects, pixel size 0.25 x 0.25 mm

3 views (adaptive)

Compression 5x, NOC, on sub-objects

update 5 ms over air

Size 200 x 600 mm, increasing to 600 x 1800 mm

Compressed min. 6.45 Gbps, Compressed max. 58 Gbps

Body Object2 moves sideways

The 'system' tracks the object

YUV - [16,12,12] bits, 4:2:2 sampling

16 sub-objects, pixel size 0.25 x 0.25 mm

3 views (adaptive)

Compression 5x, NOC, on sub-objects

update 5 ms over air

Size 300 x 900 mm

Compressed 14.5 Gbps

Face Object

The ’system’ tracks the object

YUV - [16,12,12] bits, 4:2:2 sampling

16 sub-objects, pixel size 0.25 x 0.25 mm

3 views (adaptive)

Compression 5x, NOC, on sub-objects

update 5 ms over air

Size 200 x 300 mm

Compressed 3.22 Gbps

Hands and arm Object

The ’system’ tracks the object

YUV - [16,12,12] bits, 4:2:2 sampling

16 sub-objects, pixel size 0.25 x 0.25 mm

3 views (adaptive)

Compression 5x, NOC, on sub-objects

update 5 ms over air

Size 200 x 300 mm

Compressed 3.22 Gbps

Plate and food Object

The 'system' tracks the object

YUV - [16,12,12] bits, 4:2:2 sampling

16 sub-objects, pixel size 0.25 x 0.25 mm

3 views (adaptive)

Compression 5x, NOC, on sub-objects

update 10 ms over air

Size 200 x 300 mm

Compressed 1.61 Gbps

Characteristics of objects of the Virtual Dinner, mobile researcher

When the researcher C goes to his office his display shows the objects of the scene as follows:

Background object

YUV - [10,10,10] bits, 4:2:2 sampling

4 sub-objects, pixel size 0.25 x 0.25 mm

3 views (adaptive)

Compression 100x, JPEG2000

Update every 100 ms over air (10 ms in receiver)

size 500 x 280 mm

Uncompressed 1.344 Gbps, compressed 13.44 Mbps

Body Object1 approaches fast towards dedicated cameras

The 'system' tracks the object

YUV - [16,12,12] bits, 4:2:2 sampling

16 sub-objects, pixel size 0.25 x 0.25 mm

3 views (adaptive)

Compression 5x, NOC, on sub-objects

update 10 ms over air

Size 20 x 50 mm, increasing to 200 x 280 mm

Compressed min. 26.9 Mbps, Compressed max. 1.5 Gbps

Face Object

The ’system’ tracks the objects, shoots with dedicated cameras

YUV, [16,12,12] bits, 4:2:2 sampling

16 sub-objects, pixel size 0.25 x 0.25

3 views (adaptive)

Compression 5x, NOC, on sub-objects

update 10 ms over air

Size 100 x 100 mm

Compressed 269 Mbps

TRAFFIC GENERATED BY RESEARCHER A, SENT TO RESEARCHER B

Researchers A, B and C generate different traffic. Researcher A generates about 7 Gbps to start with. When he gets on his feet and walks out of the room, the date rate increases rapidly to a maximum value of close to 60 Gbps, and then decreases down to the background object traffic, which is 127 Mbps. When he returns with his plate, the data rate increases to maximum, and when he sits down the face, arms & hands, and plate & food dominates the data rate, about 8 Gbps.

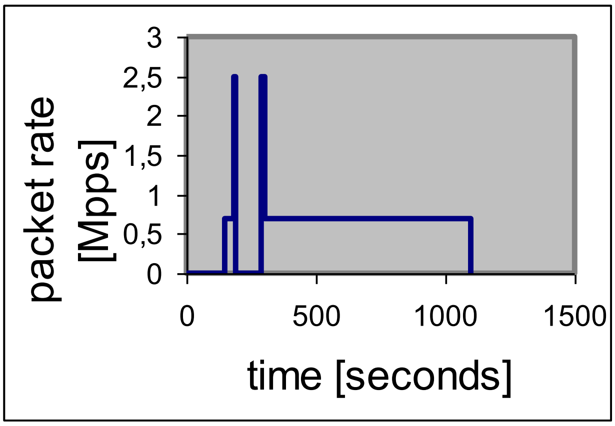


Figure 15-3. The Virtual Dinner

AN EXISTING MOVIE GENERATOR

This generator is based on inspection of parts of an existing movie, the Lord of the Rings, Two Towers [LOR01], where Gandalf is fighting the Balrog. Since an object oriented, adaptive scene version of the movie does not exist, the modeling is based on assumptions about which objects are important, their needed adaptive spatial and temporal resolution, the number of views, etc, similar to as shown for the Song Lesson case.

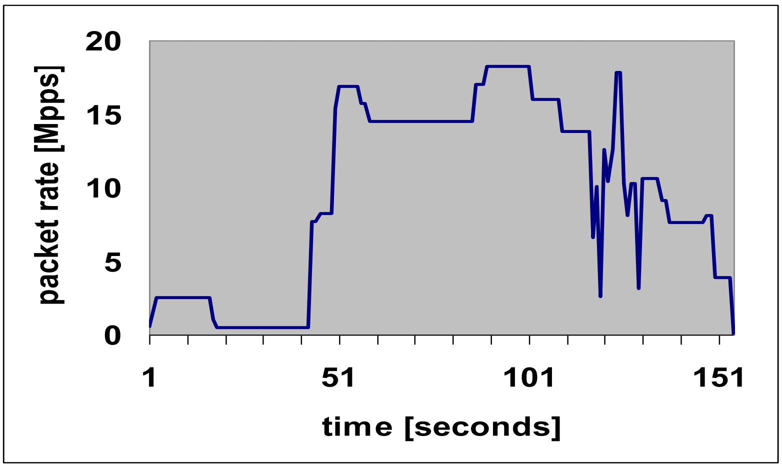


Figure 15-4. Packet rate, existing movie.

THE STEPRATE GENERATOR

To simplify the modeling of futuristic interactive movie traffic sources and other sources, a generator called StepRate generator is introduced her. Figure 15-5 shows how a simple random rate slope can be modeled by drawing four points and three line segments in the time-rate plane. The start and end points of each line segment are drawn from 2-dimentional uniform distributions.

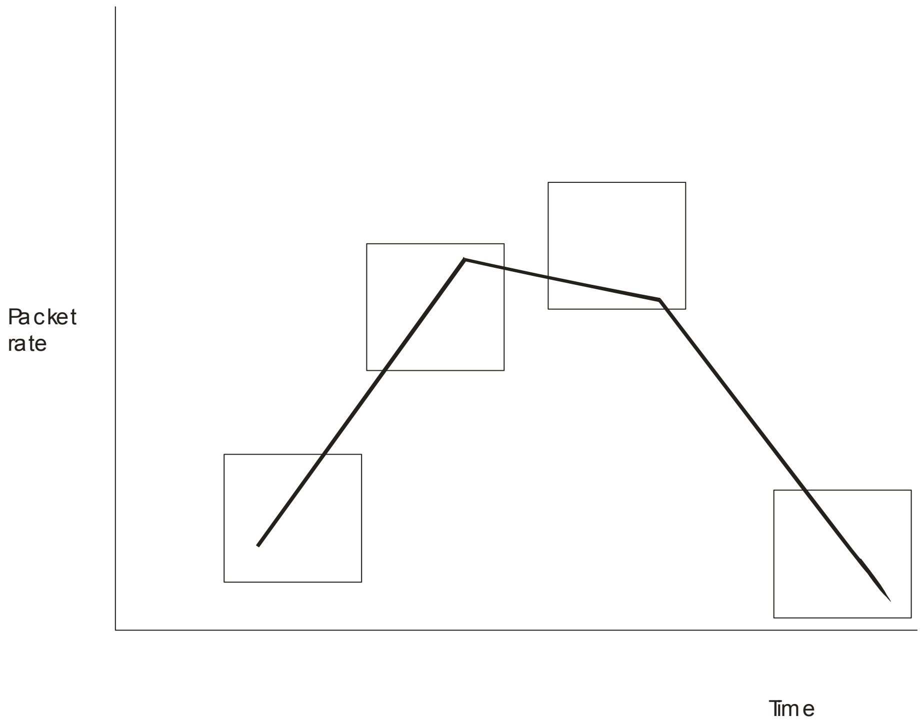


Figure 15-5. Random 4-line slope. The squares

By generating several objects, several slopes can be concatenated to form any slope as a continuous curve of line-segments.

Constant rates in short time-steps are used as an approximation to the (random) lines. The number of time-steps is input to the mode. The random, transient lines are obtained by using samples from four uniformly distributed squares (input), as shown in the figure.

Inputs to the program are written directly into the Simula/demos code, since the program is not much used, it is short and understandable, and is very fast to compile and run. The complete Simula/Demos program code is shown in Appendix 15-A.

```
entity class StepRate( k,        t1a,t1b,r1a,r1b, NTSteps12,
                                 t2a,t2b,r2a,r2b, NTSteps23,
                                 t3a,t3b,r3a,r3b, NTSteps34,
                                 t4a,t4b,r4a,r4b);
```

k - the number of this generator;

NTSteps** - number of time steps used to approximate the straight line between points * and *, integer value > 0;

t1a,t1b – the min and max values for an uniform distribution of time-rate point (t1,r1).

r1a,r1b – the min and max values for an uniform distribution of time-rate point (t1,r1).

NTsteps12 – the number of time steps between t1 and t2

Etc,,,

Note that the first rate step from r1 occurs at time t1 and the new value lasts until the next time step;

The last rate step to r2 occurs at the time step before t2, and the value lasts until t2;

To obtain a non-interrupted slope when applying several generator object serially, the value of t1 must be the same as t4 of the previous object;

It is the user’s responsibility to choose t-values so that negative time steps are always avoided;

Likewise, user must avoid negative absolute rates;

(Number :- new randint("number",1,128); Route streams to one of 128 links;) (Not usable yet)

ratechange :- new uniform("rate change",rch1,rch2); Draw the time from object generation to activation of the first generator object;

The time unit is second (s), and the rate unit is packets per second (-pps);

Example of *Input*:

```
new StepRate("StepRate(1) ",1,
            2.0,    2.0,    0.1&6,  0.1&6, 4,
```

6.0, 6.0, 3.3&6, 3.3&6, 2,

8.0, 8.0, 0.2&6, 0.2&6, 1,

12.0, 12.0, 2.7&6, 2.7&6).schedule(0.0);

Number of repetitions: 2

Activation time: rch1 = rch2 = 2

Output to one file, outf1d, by: Number :- new randint("number",1,1);

Output file outf1d:

Time	Packet rate
2.000000000&+000	9.0&+005
3.000000000&+000	1.7&+006
4.000000000&+000	2.5&+006
5.000000000&+000	3.3&+006
6.000000000&+000	1.8&+006
7.000000000&+000	2.0&+005
8.000000000&+000	2.7&+006
1.400000000&+001	9.0&+005
1.500000000&+001	1.7&+006
1.600000000&+001	2.5&+006
1.700000000&+001	3.3&+006
1.800000000&+001	1.8&+006
1.900000000&+001	2.0&+005
2.000000000&+001	2.7&+006
2.600000000&+001	9.0&+005
2.700000000&+001	1.7&+006
2.800000000&+001	2.5&+006

2.900000000&+001   3.3&+006

Output file outf2.txt:

(no records)

A FUTURISTIC INTERACTIVE MOVIE GENERATOR

This generator applies the StepRate generator. A large number of objects are assumed to behave independently as suggested in Figure 15-6, where an object (e.g. a cowboy) comes into the scene towards the camera in one second, and disappear from the camera in six seconds. Several object streams runs in parallel, and the time between each object is drawn from a uniform distribution, as shown in Appendix 15-A.

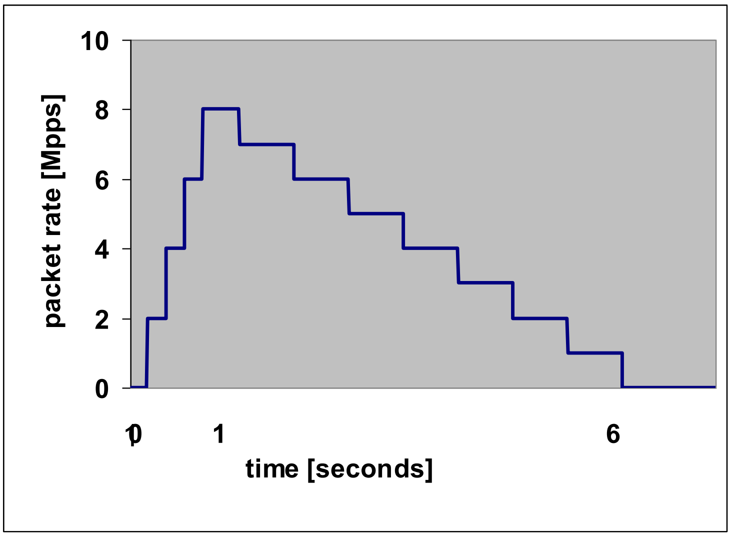


Figure 15-6. Packet rate from an object coming into the scene in 1 second, and disappearing in 6 seconds

Merged streams

Figure 15-7 shows the packet rate from 50 of each of the 4 generators as described above. Details are shown in Appendix 15-A. The merged stream packet rate shows transient slope, varying from about 60 Mpps and 260 Mpps.

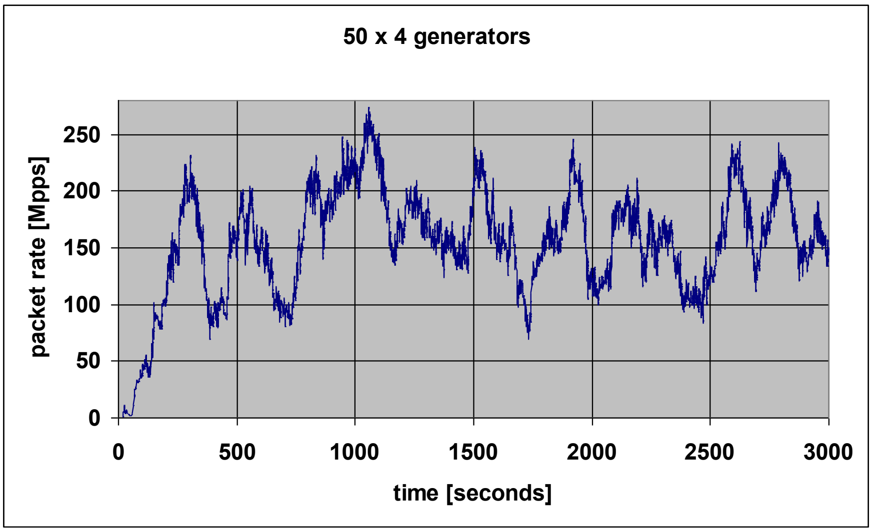


Figure 15-7. Merged packets streams, Song Lesson, Virtual Dinner, Existing Movie, Future Movie

REFERENCES

[CHA04] C Chafe, M Gurevich, G Leslie, S Tyan. Effect of Time Delay on Ensemble Accuracy. Proc. of the International Symposium on Musical Acoustics, March 20004. Nara, Japan.

[CIN72] Cinlar, E Superposition of pont processes. In stochastic point processes: Statistical analysis, theory, and applications. Ed. P A W Lewis, 549-606. New York: Wily Interscience.

[GAJ03] A Gajrat, A Hordijk and A Ridder. “Large-deviation analysis of the fluid approximation for a controllable tandem queue”. The Annals of Applied Probabiliy 2003, Vol. 13, No. 4, pp1423-1448.

[IVE07] V Bæk Iversen. Teletraffic Engineering Handbook 2007 http://oldwww.com.dtu.dk/education/34340/telenook.pdf

[JON01] P W Jones, P Smith. Stochastic Processes, An introduction. Arnold, London 2001. ISBN 0 340 80654 0

[LIE03] Lie, L. A. Rønningen, Distributed Multimedia Plays with QoS guaranties over IP, *Proc. of IEEE Wedelmusic’03,* 14–17 Sept., Leeds UK, 2003

.

[LIE04a] A. Lie, O. M. Aamo, L. A. Rønningen, On the use of classical control system based AQM for rate adaptive streaming media, *Proc. of 17th Nordic Teletraffic Seminar,* Norway, August 2004.

[LIE04b] Arne Lie, Aamo, L A Rønningen. OPTIMIZATION OF ACTIVE QUEUE MANAGEMENT BASED ON PROPORTIONAL CONTROL SYSTEM. CIIT, Virgin Islands, Nov 2004.

[LOR01] Lord of rings, Two Towers. Movie from 2001.

[RON02] L. A. Rønningen. 'Manual inspection of scenes from the movie Two Towers', Berkeley spring 2002. (personal notes, not published)

[RON02a] L. A. Rønningen, A. Lie. "Performance Control Of High-Capacity IP Networks For Collaborative Virtual Environments", published at IBC 2002 Conference Programme.

[RON02b] L. A. Rønningen, A. Lie. "Transient Behaviour of an Adaptive Traffic Control Scheme", published at EUNICE workshop, Trondheim 2002.

[RON04a] L. A. Rønningen, Dominique Guyot. DV film of 'A Virtual Song Lesson', shot of ted DMP in the Caruso Lab, Item, NTNU 2004 (the DV tape exists, but is not published)

[RON04b] L. A. Rønningen. " DMP - Distributed Multimedia Plays. Virtual Dinner". Technical report, Item, NTNU 2004, see http://www.item.ntnu.no/~leifarne

[RON05] L. A. Rønningen. Adaptive Scene and Traffic Control in DMP. IADAT tcn-2005, Portsmouth, September 2005

[RON07] L A Rønningen. A protocol stack for futuristic multimedia: Paper 2007

```
!******************************************************************************;
!************DMP traffic generator, version FNode5h********************;
!************Designer: LAR, Nov-Dec 2006, March 2007*************************;
!** Description: The Step NewRate Generator ***************;
!**Song lesson, Virtual Dinner, Movie *******************;
!**Change from earlier versions: here absolute packet NewRates are used,;
!**and instead of timeStepSize, #steps are used as inputs*****;
!**The time for activation of the generator is drawn from a uniform;
!**distribution with inputs rch1 and rch2;


EXTERNAL CLASS demos="c:/demos/demos.atr";
demos begin

        ref (outfile) array outf(1:2);
        ref (histogram) InterRateTime;
        ref (rdist) Ratechange;
        ref (idist) Number;
        integer j;
        long real simtime;
        real rch,TotRate;

        boolean procedure and2(a,b); name a,b; boolean a,b;
        and2 := if a then b else false;


!*********************************************************************************;
!**************************StepNewRate generator ******************************;
!*********************************************************************************;
!*                                  (t3a,t3b);
!*                        (t2a,r2a)        (t3b,r3b);
!*      Rate     |           (t2b,r2b)          *;
!*              |            *;
!*              | ;
!*              |                               *;
!*              |        *                   (t4a,r4a);
!*              |      (t1a,r1a)                (t4b,r4b);
!*              |      (t1b,r1b);
!*              |________________________________________________  ;
!;
!                                                          time;
!;
!**  The StepRate generator provides random, transient slopes in the time-Rate plane;
```

```
!**  Each entity object geneRates three stright line pieces between four points, as shown in the
Figure;
!**  Constant Rates in short  timesteps are used as an approx. to the (random) lines;
!**  Random, transient lines are obtained by using samples from four uniformly distributed squares;
!**  in the time-Rate plane, (t,r);
!**;
!**  Inputs:;
!**  k - the number of this generator;
!**  NTSteps** - number of time steps used to approximate the stright line between points * and *,;
!**  integer value > 0;
!**  Note that the first Rate step from r1 occurs at time t1 and the new value lasts until;
!**  the next time step.;
!**  The last Rate step to r2 occurs at the time step before t2, and the value lasts until t2;
!**  Times t1a,,,,t4b must be increasing.;
!**  To obtain a non-interrupted slope, the value of t1 is the same as t4 of the previous object;
!** ;
!**  It is the users responsibility to choose t-values so that negative time steps are always avoided;
!**  Likewise, user must avoid negative absolute Rates;
!*****************************************************************************
******************;
       entity class StepRate(k,
                             t1a,t1b,r1a,r1b, NTSteps12,
                             t2a,t2b,r2a,r2b, NTSteps23,
                             t3a,t3b,r3a,r3b, NTSteps34,
                             t4a,t4b,r4a,r4b);
       integer k, NTSteps12, NTSteps23, NTSteps34;
       long real  t1a,t1b,r1a,r1b,t2a,t2b,r2a,r2b,
                      t3a,t3b,r3a,r3b,t4a,t4b,r4a,r4b;


       begin
        integer i,j,n;
        long real t1,r1,t2,r2,t3,r3,t4,r4, RateStepSize, TimeStepSize,th;
        real NewRate, PrevRate;
        ref (rdist) Unit1, Unir1,Unit2,Unir2, Unit3,Unir3,Unit4,Unir4;

       n := 10;
       NewRate := 0.0; PrevRate := 0;

        Unit1 :- new uniform("Uni-t1",t1a,t1b);
        Unit2 :- new uniform("Uni-t2",t2a,t2b);
        Unit3 :- new uniform("Uni-t3",t3a,t3b);
        Unit4 :- new uniform("Uni-t4",t4a,t4b);
        Unir1 :- new uniform("Uni-r1",r1a,r1b);
        Unir2 :- new uniform("Uni-r2",r2a,r2b);
        Unir3 :- new uniform("Uni-r3",r3a,r3b);
        Unir4 :- new uniform("Uni-r4",r4a,r4b);

loop:
        th := Ratechange.sample;
        j := Number.sample;
        if j > 1 then j := 2;
```

```
            hold(th);

            t1 := Unit1.sample;
            t2 := Unit2.sample;
            t3 := Unit3.sample;
            t4 := Unit4.sample;
            r1 := Unir1.sample;
            r2 := Unir2.sample;
            r3 := Unir3.sample;
            r4 := Unir4.sample;

!**time steps between t1 and t2;

            RateStepSize := (r2-r1)/NTSteps12;
            TimeStepSize := (t2-t1)/NTSteps12;

            NewRate := r1;
            for i := 1 step 1 until NTSteps12 do
            begin
              NewRate := NewRate + RateStepSize;
              TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
              outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15);
              outf(j).outimage;
              hold(TimeStepSize);
            end;


            NewRate := 0.0;

!**time steps between t2 and t3;

            RateStepSize := (r3-r2)/NTSteps23;
            TimeStepSize := (t3-t2)/NTSteps23;

            NewRate := r2;
            for i := 1 step 1 until NTSteps23 do
            begin
              NewRate := NewRate + RateStepSize;
          TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
              outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15);
              outf(j).outimage;
              hold(TimeStepSize);
            end;


            NewRate := 0.0;

!**time steps between t3 and t4;

            RateStepSize := (r4-r3)/NTSteps34;
            TimeStepSize := (t4-t3)/NTSteps34;
```

```
TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
     NewRate := r3;
     for i := 1 step 1 until NTSteps34 do
     begin
       NewRate := NewRate + RateStepSize;

       outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15);
       outf(j).outimage;
       hold(TimeStepSize);
     end;


     NewRate := 0.0;


    n := n - 1;
    if n > 0 then repeat;

   end*********;





!*************************************************************************;
!********************** song lesson generator*******************************;
!*************************************************************************;
         entity class SongLessonGen(k);
         integer k;
         begin

          real t2, PrevRate,NewRate; integer i,j;
          i := 3; PrevRate := 0.0;
loop:
          t2 := Ratechange.sample;
          j := Number.sample;
         if j > 1 then j := 2;

         hold(t2);
         NewRate := 0.5; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
         outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
         hold(35.0);

         NewRate := 0.25; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
         outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
         hold(20.0);

         NewRate := 0.5; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
         outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
         hold(1000.0);

         NewRate := 0.0; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
```

```
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(680.0);

        NewRate := 0.5; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(300.0);

        NewRate := 0.25; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(60.0);

        NewRate := 0.0; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;

         i := i - 1;
         if i > 0 then repeat;
        end***SongLessonGen***;




!***************************************************************************;
!********************** virtual dinner generator****************************;
!***********************route streams to two files, j ******************;
        entity class VirtualDinnerGen(k);
        integer k;
        begin
        real t2, PrevRate, NewRate; integer i,j;
        i := 3; PrevRate := 0.0;
loop:
        t2 := Ratechange.sample;
        j := Number.sample;
        if j > 1 then j := 2;

        hold(t2);

        NewRate := 0.7; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(30.0);

        NewRate := 2.5; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(10.0);

        NewRate := 0.0; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(100.0);

        NewRate := 2.5; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
```

```
        hold(10.0);

        NewRate := 0.7; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(1000.0);

        NewRate := 0.0; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;

         i := i - 1;
         if i > 0 then repeat;

        end***n***;




!***************************************************************************;
!**************Movie generator, The foundation of stone****************;
!**********************route streams to two files, j ******************;
        entity class MovGen(k);
        integer k;
        begin
        real t2, NewRate,PrevRate; integer i,j;
        i := 12;
loop:
        t2 := Ratechange.sample;
        j := Number.sample;
        if j > 1 then j := 2;

        hold(t2);

        NewRate := 0.5; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(1.0);

        NewRate := 1.0; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(1.0);

        NewRate := 2.5; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(15.0);

        NewRate := 1.0; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(1.0);

        NewRate := 0.5; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
        hold(25.0);
```

```
NewRate := 7.7; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(2.0);

NewRate := 8.2; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(4.0);

NewRate := 15.4; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(1.0);

NewRate := 16.9; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(6.0);

NewRate := 15.7; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(2.0);

NewRate := 14.5; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(28.0);

NewRate := 17.0; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(3.0);

NewRate := 18.2; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(12.0);

NewRate := 16.0; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(8.0);

NewRate := 13.8; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(8.0);

NewRate := 6.6; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(1.0);

NewRate := 10.1; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(1.0);

NewRate := 2.6; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(1.0);
```

```
NewRate := 12.6; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(1.0);

NewRate := 10.4; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(1.0);

NewRate := 12.6; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(1.0);

NewRate := 17.8; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(2.0);

NewRate := 10.3; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(1.0);

NewRate := 8.1; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(1.0);

NewRate := 10.3; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(2.0);

NewRate := 3.1; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(1.0);

NewRate := 10.6; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(5.0);

NewRate := 9.1; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(2.0);

NewRate := 7.6; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(10.0);

NewRate := 8.1; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(2.0);

NewRate := 3.9; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;
hold(5.0);
```

```
        NewRate := 0.0; TotRate := TotRate - PrevRate + NewRate; PrevRate := NewRate;
        outf(j).outreal(time,10,22); outf(j).outreal(TotRate,3,15); outf(j).outimage;


         i := i - 1;
         if i > 0 then repeat;

        end***n***;




        outf(1) :- new outfile("outf1d.txt");
        outf(1).open(blanks(120));
        outf(1).outtext("Resultater forsøk");
        outf(1).outimage;




!*******one second is used as time unit;
!*******Rates in packets per second;

        simtime:=inreal;

        outf(2) :- new outfile("outf2.txt");
        outf(2).open(blanks(120));
        outf(2).outtext("Resultater forsøk");
        outf(2).outimage;

        TotRate := 0.0;

        rch := 1000.0;
!       outf :- outf(1);

!       trace;

!       notrace;


!******route streams to one of 128 links;

        Number :- new randint("number",1,1);

!******draw the time from object generation to activation of object;
        Ratechange :- new uniform("Rate change",0.5,rch);
```

```
        for j := 1 step 1 until 50 do
        new StepRate("StepRate(1) ",1,
                1.0,      1.0,      0.0,   0.0,   4,
                2.0,                2.0,                8.0,      8.0,   8,
                8.0,                8.0,                0.0,      0.0,   10,
                12.0,               12.0,               0.0,      0.0).schedule(0.0);

!       new StepRate("StepRate(2) ",2,1,            ;
!                       0,      0,      0.0,   0.0,   ;
!                       2,   2, 150.0&6, 150.0&6,       ;
!                       5,   5, 150.0&6, 150.0&6,       ;
!                       7,   7,  0.0,   0.0).schedule(0.0);

!      InterRateTime :- new histogram("TotRate",0, 200, 20);



       for j := 1 step 1 until 50 do
       new SongLessonGen("SongLessonGen",j).schedule(0.0);

       for j := 1 step 1 until  50 do
       new VirtualDinnerGen("VirtualDinnerGen1",j).schedule(0.0);

       for j := 1 step 1 until 50 do
       new MovGen("MovGen",j).schedule(0.0);

       hold(simtime);
       outf(1).close;
       outf(2).close;

end*****demos***;
```

PART 16: MODELING AND SIMULATION OF THE DROPPING SCHEME

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

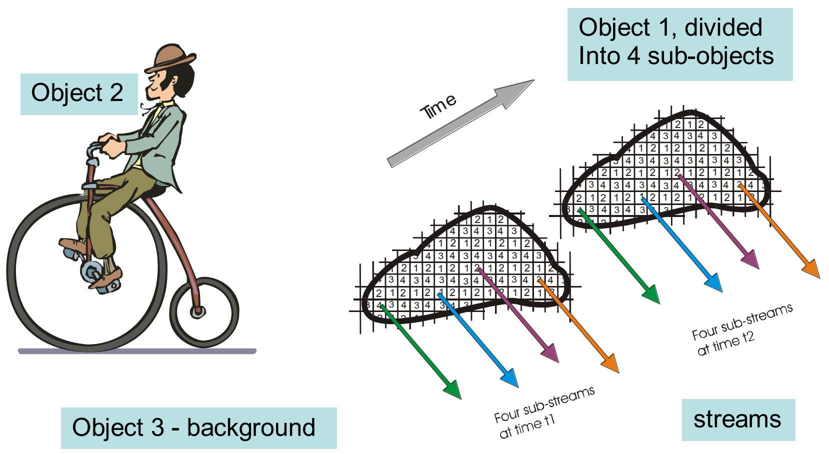
NODE OUTPUT QUEUE MODEL

DROPPING AND PRIORITIZING

Presented at DEI, the Unviversity of Padava, 10nov06

by L A Rønningen

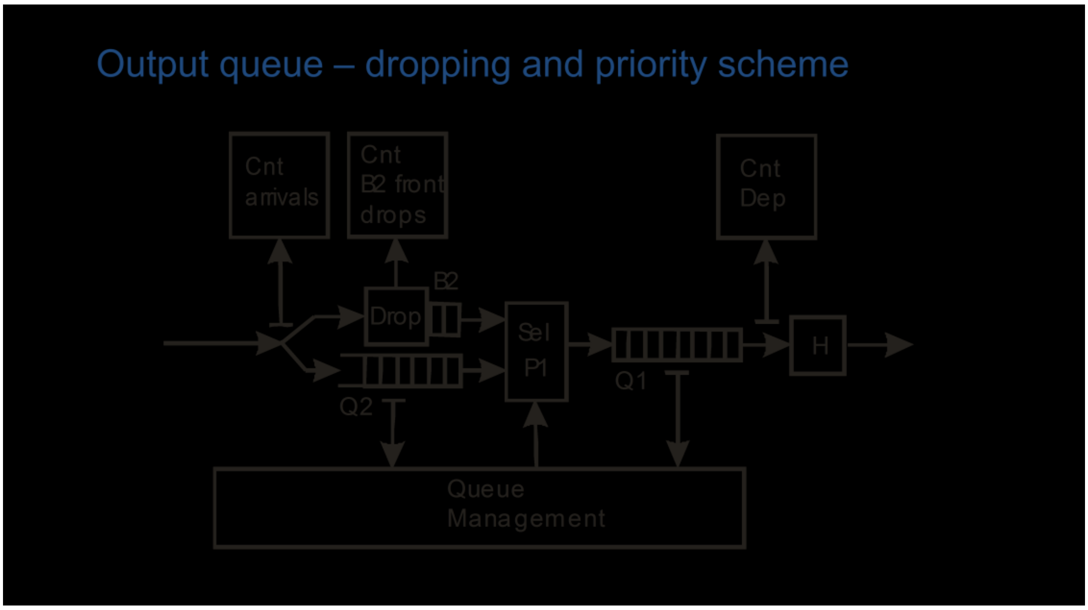
Object oriented scene with sub-objects



Dropping strategy

- If objects are moving fast, drop packets carrying spatial pixels
- If objects are nearly static, drop packets carrying temporal pixels

- Note that the sub-object approach can be used with any compression/coding scheme

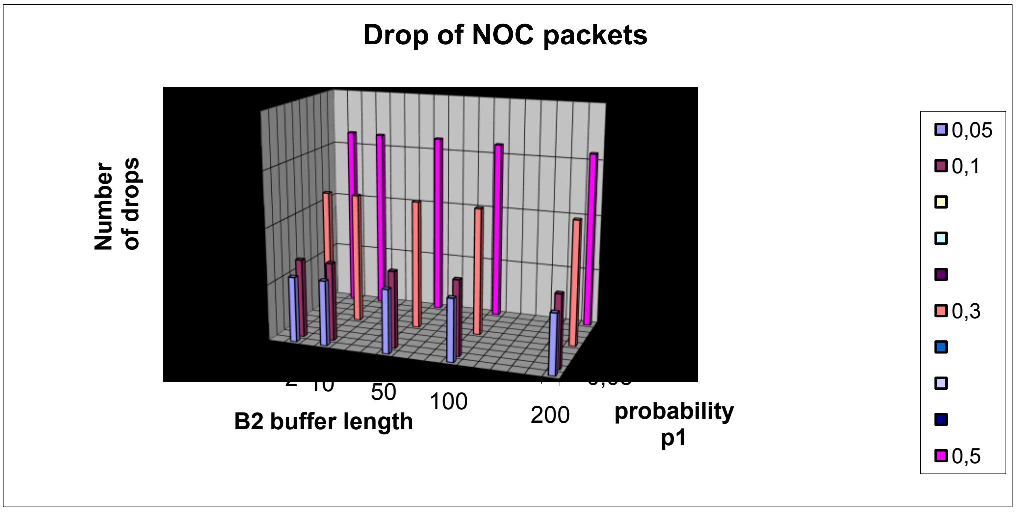


Specification & behavior

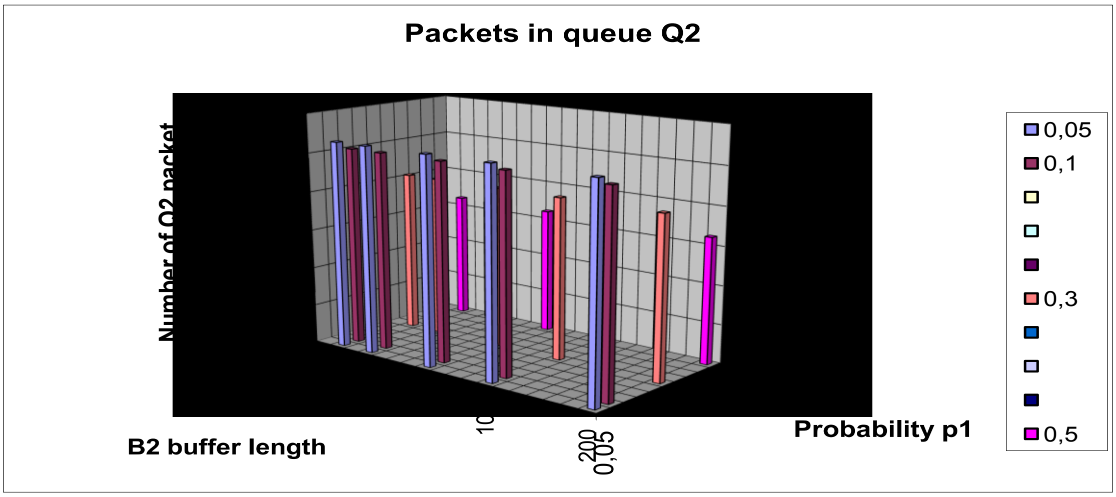
- Two classes of packets
  - NOC – controlled drop, abs. maximum delay 1 ms, 9 sub-objects, enter B2
  - CTRL – no drop, can be delayed, enter Q2
  - Packet rate, 1200 + 1200 packets per ms (from uniform distributions)
- Link capacity, 1000 packets per ms
- Q1, queue, max 1000 packets, delay 1 ms
- Q2, queue, unlimited length
- B2, buffer, max 50 packets
  - If buffer length 0 - 12 join B2
  - If buffer length 13 - 18 drop packets from sub-object 8
  - If buffer length 19 - 24 drop packets from sub-object 8 and 7
  - and so on,,,,,,,
- Sel, selects packets from B2 and Q2, and forwards to Q1
- p1, prob. of selecting packet from Q2
- 1- p1, prob. of selecting packet from B2

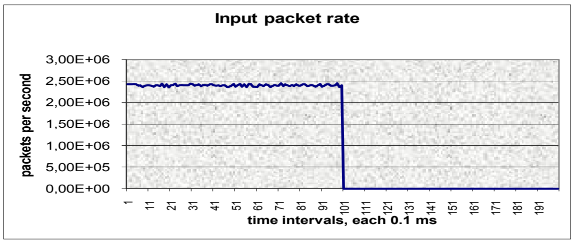
Drop of NOC packets,

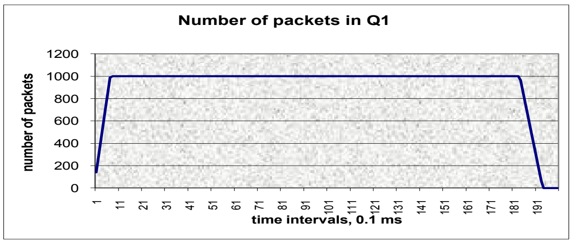
B2 and Q1 full, Q2 increasing

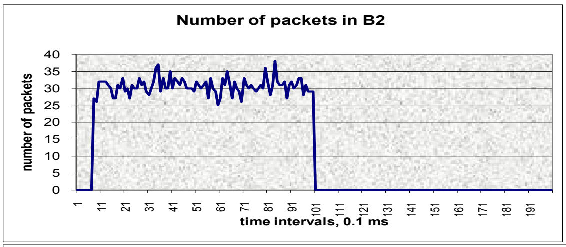


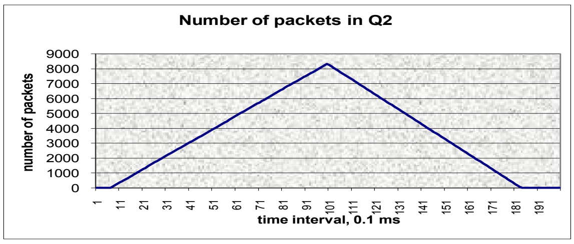
B2 buffer lenght = 50, and Prob p1 = 0.3 used in following runs

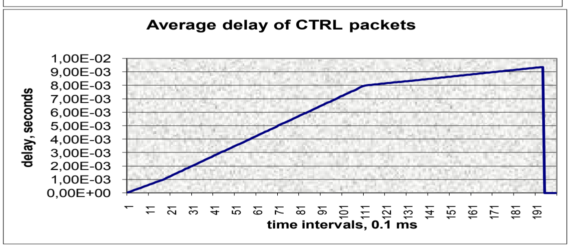

Packets in queue Q2
Number of Q2 packet
B2 buffer length
10
200
0,05
Probability p1
0,05
0,1
0,3
0,5

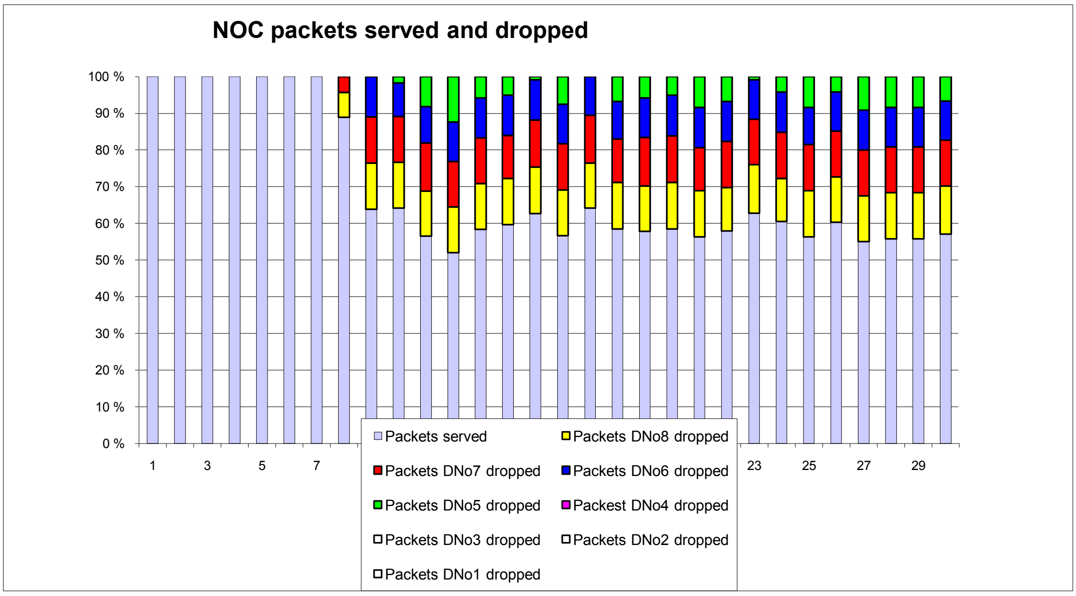

Input packet rate
packets per second
3,00E+06
2,50E+06
2,00E+06
1,50E+06
1,00E+06
5,00E+05
0,00E+00
time intervals, each 0.1 ms


Number of packets in Q1
number of packets
1200
1000
800
600
400
200
0
time intervals, 0.1 ms


Number of packets in B2
number of packets
40
35
30
25
20
15
10
5
0
time intervals, 0.1 ms


Number of packets in Q2
number of packets
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
time interval, 0.1 ms


Average delay of CTRL packets
delay, seconds
1,00E-02
9,00E-03
8,00E-03
7,00E-03
6,00E-03
5,00E-03
4,00E-03
3,00E-03
2,00E-03
1,00E-03
0,00E+00
time interval, 0.1 ms


NOC packets served and dropped
100 %
90 %
80 %
70 %
60 %
50 %
40 %
30 %
20 %
10 %
0 %
1
3
5
7
23
25
27
29
Packets served
Packets DNo8 dropped
Packets DNo7 dropped
Packets DNo6 dropped
Packets DNo5 dropped
Packest DNo4 dropped
Packets DNo3 dropped
Packets DNo2 dropped
Packets DNo1 dropped

APPENDIX A. DEMOS SIMULATION PROGRAM

```
!*************************************************************;
!*                                                            *;
!*      DMP Node output link queueing, (version q1q2e)        *;
!*      adaptive B2 dropping mechanism is implemented         *;
!*      Designer: LAR, Nov-Dec 2006                           *;
!*                                                            *;
!*************************************************************;


!* Make sure this path points to the location of your 'demos.atr' file. *;
!* This is the path if cim is installed on C:/                           *;

EXTERNAL CLASS demos="c:/cim/demos/demos.atr";


demos begin

      ref (outfile) outf1, outf2;

      ref (waitq) Q1, Q2, B2;

      ref (condq) cq;

      ref (CTRL) PCTRL;

      ref (NOC) PNOC;

      ref (histogram) TotTimeCTRL, TotTimeNOC;

      ref (count) CntB1DrNOC8, CntB1DrNOC7, CntB1DrNOC6,
        CntB1DrNOC5, CntB1DrNOC4, CntB1DrNOC3,
        CntB1DrNOC2, CntB1DrNOC1;

      ref (rdist) nextCTRL, nextNOC, nextNOC2, xdraw;

      long real simTime, ht, Q1m,Q1n, changeTimeNOC1,
            changeTimeNOC2, changeTimeNOC3, changeTimeCTRL;

      real p1,tx;

      integer ix, buflength,CntQ1DrNOC;

      boolean procedure and2(a,b); name a,b; boolean a,b;
      and2 := if a then b else false;

      boolean procedure or2(a,b); name a,b; boolean a,b;
      or2 := if a then true else b;


!*************************************************************;
!*      Control packet generation                             *;
!*************************************************************;
```

```
     entity class CTRL;
     begin
       ref (packetCTRL) pk1;
       integer SN;

       SN := 1;
       while time < changeTimeCTRL do
       begin
         hold(nextCTRL.sample);
         new packetCTRL("packetCTRL",1,SN, 50&3).schedule(0.0);
         SN := SN +1;
       end;

     end***CTRL***;



!*************************************************************;
!*       NOC packet generation                               *;
!*************************************************************;

     entity class NOC;
     begin
       ref (packetNOC) pk2;
       integer SN;

       SN := 1;
       while time < changeTimeNOC1 do
       begin
         hold(nextNOC.sample);
         new packetNOC("packetNOC",2,SN,750&3).schedule(0.0);
         if SN < 9 then
             begin
               SN := SN + 1;
             end else
             begin
               SN := 1;
             end;
       end;

     end***NOC***;



!*************************************************************;
!*       Control Packet                                      *:
!*************************************************************;

     entity class packetCTRL(PT, SeqNo,Rate);
     integer PT, SeqNo;
     real Rate;

     begin
       long real t1;
       t1 := time;

       cq.signal;
```

```
  Q2.wait;

  Q1.wait;

  if PT = 1 then
  begin
    TotTimeCTRL.update(time - t1);
  end;
end***packet***;



!************************************************************;
!*        NOC Packet                                        *;
!*                                                          *;
!* (ex. 7/4 = 1,75 || 7//4 = 1 )                            *;
!************************************************************;

entity class packetNOC(PT, SeqNo,Rate);
integer PT, SeqNo;
real Rate;


begin
  integer l1;
  long real t1;
  real help1;
  t1 := time;


  if B2.length <= 12 then
  begin
    cq.signal;
    B2.wait;
  end else

  if and2(B2.length > 12, B2.length <= 18) then
  begin
    if SeqNo/8 eq SeqNo//8 then
    begin
      CntB1DrNOC8.update(1);
      goto Lterm;
    end else
    begin
      cq.signal;
      B2.wait;
    end;
  end else

  if and2(B2.length > 18, B2.length <= 24) then
  begin
    if SeqNo/8 eq SeqNo//8 then
    begin
      CntB1DrNOC8.update(1);
      goto Lterm;
    end else
    if SeqNo/7 eq SeqNo//7 then
```

```
  begin
    CntB1DrNOC7.update(1);
    goto Lterm;
  end else
  begin
    cq.signal;
    B2.wait;
  end;
end else

if and2(B2.length > 24, B2.length <= 30) then
begin
  if SeqNo/8 eq SeqNo//8 then
  begin
    CntB1DrNOC8.update(1);
    goto Lterm;
  end else
  if SeqNo/7 eq SeqNo//7 then
  begin
    CntB1DrNOC7.update(1);
    goto Lterm;
  end else
  if SeqNo/6 eq SeqNo//6 then
  begin
    CntB1DrNOC6.update(1);
    goto Lterm;
  end else
  begin
    cq.signal;
    B2.wait;
  end;
end else

if and2(B2.length > 30, B2.length <= 36) then
begin
  if SeqNo/8 eq SeqNo//8 then
  begin
    CntB1DrNOC8.update(1);
    goto Lterm;
  end else
  if SeqNo/7 eq SeqNo//7 then
  begin
    CntB1DrNOC7.update(1);
    goto Lterm;
  end else
  if SeqNo/6 eq SeqNo//6 then
  begin
    CntB1DrNOC6.update(1);
    goto Lterm;
  end else
  if SeqNo/5 eq SeqNo//5 then
  begin
    CntB1DrNOC5.update(1);
    goto Lterm;
  end else
  begin
    cq.signal;
```

```
    B2.wait;
  end;
end else

if and2(B2.length > 36, B2.length <= 40) then
begin
  if SeqNo/8 eq SeqNo//8 then
  begin
    CntB1DrNOC8.update(1);
    goto Lterm;
  end else
  if SeqNo/7 eq SeqNo//7 then
  begin
    CntB1DrNOC7.update(1);
    goto Lterm;
  end else
  if SeqNo/6 eq SeqNo//6 then
  begin
    CntB1DrNOC6.update(1);
    goto Lterm;
  end else
  if SeqNo/5 eq SeqNo//5 then
  begin
    CntB1DrNOC5.update(1);
    goto Lterm;
  end else
  if SeqNo/4 eq SeqNo//4 then
  begin
    CntB1DrNOC4.update(1);
    goto Lterm;
  end else
  begin
    cq.signal;
    B2.wait;
  end;
end else

if and2(B2.length > 40, B2.length <= 46) then
begin
  if SeqNo/8 eq SeqNo//8 then
  begin
    CntB1DrNOC8.update(1);
    goto Lterm;
  end else
  if SeqNo/7 eq SeqNo//7 then
  begin
    CntB1DrNOC7.update(1);
    goto Lterm;
  end else
  if SeqNo/6 eq SeqNo//6 then
  begin
    CntB1DrNOC6.update(1);
    goto Lterm;
  end else
  if SeqNo/5 eq SeqNo//5 then
  begin
    CntB1DrNOC5.update(1);
```

```
    goto Lterm;
  end else
  if SeqNo/4 eq SeqNo//4 then
  begin
    CntB1DrNOC4.update(1);
    goto Lterm;
  end else
  if SeqNo/3 eq SeqNo//3 then
  begin
    CntB1DrNOC3.update(1);
    goto Lterm;
  end else
  begin
    cq.signal;
    B2.wait;
  end;
end else

if and2(B2.length > 46, B2.length <= 50) then
begin
  if SeqNo/8 eq SeqNo//8 then
  begin
    CntB1DrNOC8.update(1);
    goto Lterm;
  end else
  if SeqNo/7 eq SeqNo//7 then
  begin
    CntB1DrNOC7.update(1);
    goto Lterm;
  end else
  if SeqNo/6 eq SeqNo//6 then
  begin
    CntB1DrNOC6.update(1);
    goto Lterm;
  end else
  if SeqNo/5 eq SeqNo//5 then
  begin
    CntB1DrNOC5.update(1);
    goto Lterm;
  end else
  if SeqNo/4 eq SeqNo//4 then
  begin
    CntB1DrNOC4.update(1);
    goto Lterm;
  end else
  if SeqNo/3 eq SeqNo//3 then
  begin
    CntB1DrNOC3.update(1);
    goto Lterm;
  end else
  if SeqNo/2 eq SeqNo//2 then
  begin
    CntB1DrNOC2.update(1);
    goto Lterm;
  end else
  begin
    cq.signal;
```

```
      B2.wait;
     end;
    end else goto Lterm;

    Q1.wait;


Lterm:
    if PT = 2 then
    begin
     TotTimeNOC.update(time - t1);
    end;

   end***packet***;



!*************************************************************;
!*        Box server                                           *;
!*************************************************************;

   entity class Box;
   begin
    ref (entity) ep;
    real p;

loop:
    cq.waituntil(or2(
    and2(Q2.length > 0,Q1.length < Q1n),
    and2(B2.length > 0,Q1.length < Q1n)));

    begin
     p := xdraw.sample;
     ep :- none;


     if p > p1 then
     begin
      if B2.length > 0 then ep :- B2.coopt;
     end else
     begin
      if Q2.length > 0 then ep :- Q2.coopt;
     end;

     if ep =/= none then
     begin
       ep.schedule(now);
     end;
    end;

    repeat;
   end***Box***;
```

```
!*************************************************************;
!*      Output link server                                   *;
!*************************************************************;

        entity class H;
        begin
          ref (entity) ep;
loop:
          ep :- Q1.coopt;
          cq.signal;
          hold(ht);
          ep.schedule(0.0);
          repeat;
        end***H***;



!*************************************************************;
!*      Simulation:                                          *;
!*                                                           *;
!*      one second is used as time unit                      *;
!*      rates in packets per second                          *;
!*************************************************************;

        CntB1DrNOC8 :- new count("CntB1DrNOC8");
        CntB1DrNOC7 :- new count("CntB1DrNOC7");
        CntB1DrNOC6 :- new count("CntB1DrNOC6");
        CntB1DrNOC5 :- new count("CntB1DrNOC5");
        CntB1DrNOC4 :- new count("CntB1DrNOC4");
        CntB1DrNOC3 :- new count("CntB1DrNOC3");
        CntB1DrNOC2 :- new count("CntB1DrNOC2");
        CntB1DrNOC1 :- new count("CntB1DrNOC1");


        nextCTRL :- new uniform("nextCTRL", 0.8/(1.2&6), 1.2/(1.2&6));
        nextNOC :- new uniform("nextNOC", 0.8/(1.2&6), 1.2/(1.2&6));
        nextNOC2 :- new uniform("nextNOC2", 2.0, 2.0);
        xdraw :- new uniform("xdraw",0.0, 1.0);

        Q1 :- new waitq("Q1");
        Q2 :- new waitq("Q2");
        B2 :- new waitq("B2");

        cq :-  new condq("cq");

        TotTimeCTRL :- new histogram("TotTimeCTRL",1&-6,11&-3,11);
        TotTimeNOC :- new histogram("TotTimeNOC",1&-6, 1.1&-3,11);

        buflength := 50;
        p1 := 0.3;

        ht := 1&-6;
        Q1m :=  1.1&3;
        Q1n  :=  1.0&3;
```

```
outf1 :- new outfile("queuemodel_sim.txt");
outf1.open(blanks(120));


PCTRL :- new CTRL("CTRL");
PCTRL.schedule(0.0);
PNOC :- new NOC("NOC");
PNOC.schedule(0.0);

new Box("Box").schedule(0.0);
new H("H").schedule(0.0);

changeTimeNOC1 := 1&-2;
changeTimeNOC2 := 0.5;
changeTimeNOC3 := 1.0;

changeTimeCTRL := 1&-2;


outf2 :- new outfile("queuemodel_drop.txt");
outf2.open(blanks(120));
outf2.outtext("OK forsøk");
outf2.outimage;


outf :- outf1;



!****************************************************************;
!*        Output files:                                          *;
!*                                                               *;
!*        one second is used as time unit                        *;
!*        rates in packets per second                            *;
!****************************************************************;

notrace;
noreport;


begin

  outf1.outimage;
  outf1.outtext("**********************************");
  outf1.outtext("**********************************");
  outf1.outimage;
  outf1.outtext("**********************************");
  outf1.outtext("**********************************");
  outf1.outimage;
  outf1.outtext("**********************************");
  outf1.outtext("**********************************");
  outf1.outimage;
  outf1.outtext("p1 =   ");
  outf1.outfix(p1,2,13);
  outf1.outimage;
  outf1.outtext("buflength =   ");
  outf1.outint(buflength,4);
```

```
        outf1.outimage;


        for ix := 1 step 1 until 200 do
        begin
          hold(1.0&-4);
          outf1.outint(nextCTRL.obs + nextNOC.obs,7);
          if TotTimeCTRL.Myt.Obs > 0 then
          outf1.outreal(TotTimeCTRL.Myt.Sum/TotTimeCTRL.Myt.Obs,4,12);
          outf1.outreal(TotTimeCTRL.Myt.Max,4,12);
          if TotTimeNOC.Myt.Obs > 0 then
          begin
          outf1.outreal(TotTimeNOC.Myt.Sum/TotTimeNOC.Myt.Obs,4,12);
          end else outf1.outint(0, 12);
          outf1.outreal(TotTimeNOC.Myt.Max,4,12);
          outf1.outint(Q1.LENGTH,5);
          outf1.outint(Q2.LENGTH,6);
          outf1.outint(B2.LENGTH,4);
          outf1.outimage;

          outf2.outint(nextNOC.obs,7);
          outf2.outint(CntB1DrNOC8.obs,4);
          outf2.outint(CntB1DrNOC7.obs,4);
          outf2.outint(CntB1DrNOC6.obs,4);
          outf2.outint(CntB1DrNOC5.obs,4);
          outf2.outint(CntB1DrNOC4.obs,4);
          outf2.outint(CntB1DrNOC3.obs,4);
          outf2.outint(CntB1DrNOC2.obs,4);
          outf2.outint(CntB1DrNOC1.obs,4);
          outf2.outimage;

          reset;
        end;
       end;


!      outf1.close;
       outf2.close;

end***demos***;
```

PART 17: MOBILITY IN DMP

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

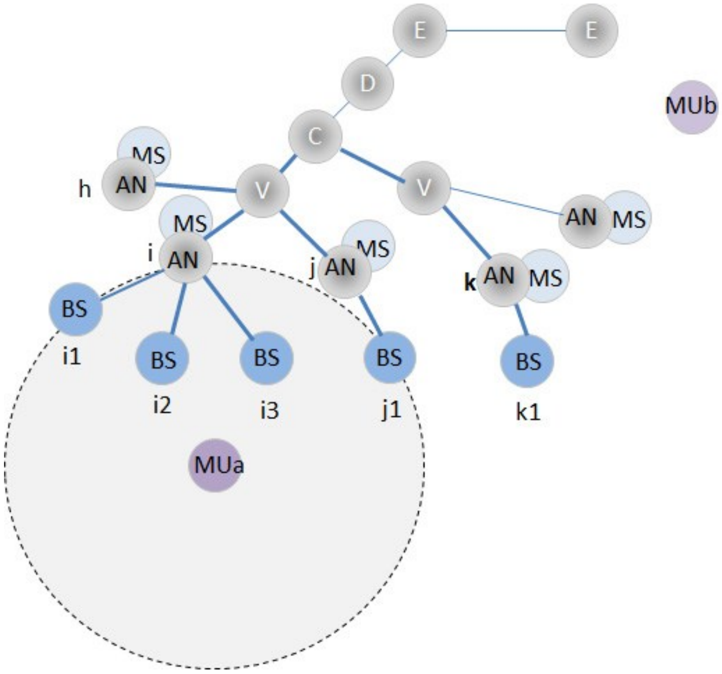
MOBILE USERS, MANAGEMENT AND HANDOVER

Base Stations, BSs, are normally connected to Access Nodes, ANs, of DMP. A Mobile Server, MS, also connected to AN (or maybe to a Village Node) manages the BSs and the Mobile Users, MUs, having an MS as the home location or visiting location. An MS also communicates with MSs under other ANs. Figure 17-1 depicts the situation where MUa, which has its home location at ANh but now is approaching ANi, is switched on. Localization and seem-less handover are required.

MUs normally are within the cell range of several BSs. In this case it is assumed that MUa after switched on can communicate with four BSs, three at ANi, and one at ANj. MUa measures the electromagnetic field strength of the four BSs and selects the strongest as ‘active’, which in this case it is BSi2. The other three BSs are selected as ‘stand-by’. MUa sends its ID to all four BSs, which forward to their respective MS. The ‘active’ MS then register the MUa ID, allocates an IP address to it, and sends to ANh/MSh which is the home server, so that calls for MUa can be re-routed. The other participating MSj register MUa as ‘discovered’ and assign IP address.

After a time the user/MUa wants to set up a collaborative session with another MUb (or fixed user). This is done in the normal way by negotiating SceneProfile and QualityShapingProfile. After a short time the session commences. Immediately, MUa sends the Collaboration Status to ANj/MSj

MUa is moving, and at a certain position BSi3 becomes stronger than BSi2. MUa then sends a request to MSi which in turn informs MSh and MSb about the new IP address of MUa. From now on the new IP address is used, and the new status is registered in MSh, MSa and MSb.



Legend: BS – Base Station,   MU – Mobile wireless User,   AN – Access Nodes, MS – Mobile Server

Figure 17-1. DMP network with Base Stations and Mobile Users

THE DMP ARCHITECTURE

Technical Report on Futuristic Architecture and Technology (2007), 2011

Leif Arne Rønningen, professor

Norwegian University of Science and Technology, Dept. of Telematics.
O. S. Bragstads plass 2B

N-7491 Trondheim, Norway
tel. 47 9003 0473

leifarne@item.ntnu.no

XXXXXXXXXXXXX

Since the AppTraNetLFC protocol uses IP v6 'as is', standard IP v6 packets can be sent through DMP networks as service class E packets. And vice versa, AppTraNetLFC packets of service class E can be sent through existing Internet. In the latter case, if for example sequence guarantee is needed, the TCP protocol can be applied end-to-end in order to enclose the the AppTraNetLFC packets. This means that a route from one user to another can go through a mix of Internets and DMP networks. The two cases of interconnection of Internet and DMP in a node are shown in Figures 18-1. DMP Collaboration Spaces can use the old Internet as is, but the controlled dropping of packets occurs only in the user equipment. If most 'heavy' users do the same dropping the delay through the old Internet could be close to constant.

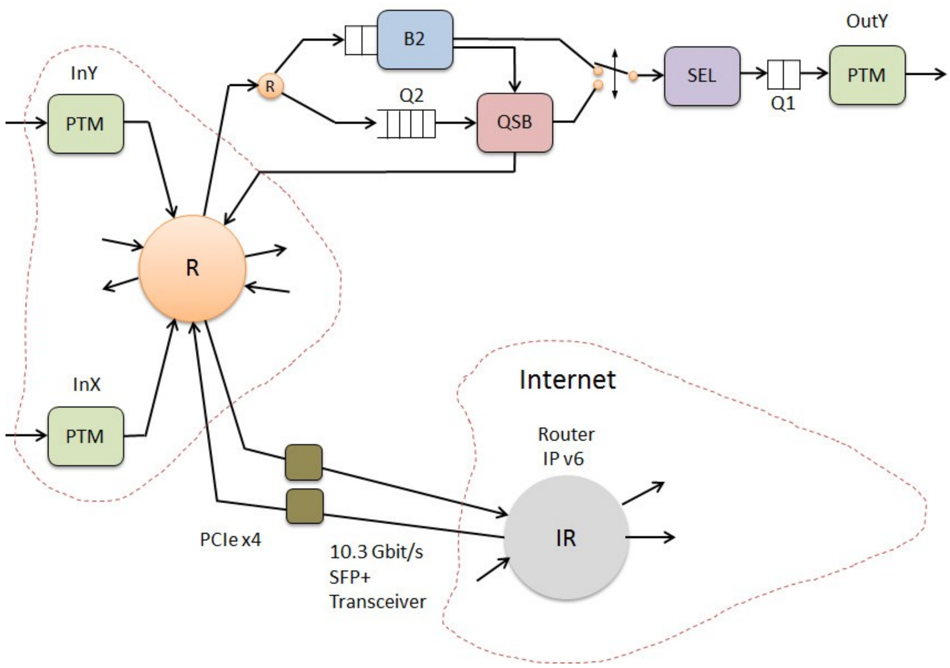


Figure 18-1. Interconnection of DMP and old Internet using a Gateway.

The allocation of IPv6 address bits to nodes and users are not the same in DMP and Internet. This can in principle be solved in two ways, either using the same allocation for both systems, or using a double set of source and destination addresses. The latter occupies 32 bytes more in the protocol header, and the address has to be changed in the Gateway module between the two routers as shown in Figure 18-1.