SIRSALE : Integrated Video Databases Management Tools

L. Brunie[a], L. Favory[a], J.P. Gelas[b], L. Lefèvre[b], A. Mostefaoui[c], F. Nait-Abdesselam[d]

[a] Information Systems Engineering Lab.
National Institute of Applied Sciences of Lyon, 69621 Villeurbanne Cedex, France.
[Lionel.Brunie, Loic.Favory]@insa-lyon.fr

[b]RESAM laboratory - INRIA Action RESO
Ecole Normale Supérieure de Lyon - 46, allée d'Italie, 69364 LYON Cedex 07, France
Laurent.Lefevre@inria.fr

[c]Franche Comté Computer Science Lab.
4, place Tharradin, Montbéliard, France
amostefa@pu-pm.univ-fcomte.fr

[d]CITI Laboratory - INRIA Action ARES
National Institute of Applied Sciences of Lyon, 69621 Villeurbanne Cedex, France.
fnait@telecom.insa-lyon.fr

ABSTRACT

Video databases became an active field of research during the last decade. The main objective in such systems is to provide users with capabilities to friendly search, access and playback distributed stored video data in the same way as they do for traditional distributed databases. Hence, such systems need to deal with hard issues : (a) video documents generate huge volumes of data and are time sensitive (streams must be delivered at a specific bitrate), (b) contents of video data are very hard to be automatically extracted and need to be humanly annotated. To cope with these issues, many approaches have been proposed in the literature including data models, query languages, video indexing etc. In this paper, we present SIRSALE : a set of video databases management tools that allow users to manipulate video documents and streams stored in large distributed repositories. All the proposed tools are based on generic models that can be customized for specific applications using ad-hoc adaptation modules. More precisely, SIRSALE allows users to : (a) browse video documents by structures (sequences, scenes, shots) and (b) query the video database content by using a graphical tool, adapted to the nature of the target video documents. This paper also presents an annotating interface which allows archivists to describe the content of video documents. All these tools are coupled to a video player integrating remote VCR functionalities and are based on active network technology. So, we present how dedicated active services allow an optimized video transport for video streams (with Tamanoir active nodes). We then describe experiments of using SIRSALE on an archive of news video and soccer matches. The system has been demonstrated to professionals with a positive feedback. Finally, we discuss open issues and present some perspectives.

Keywords: video databases, content-based retrieval, video querying, video annotating, video transport, active networking, SIRSALE, Tamanoir.

1. INTRODUCTION

Over the past decade, distributed multimedia systems have been the object of an increasing research and development effort. One of the main targets of these works is the design and management of huge digital video libraries accessible by content by remote users. However such systems impose very hard constraints: (a) huge volumes of data (b) the complex structure of video documents whose semantic interpretation can be very subjective (c) quality of service constraints that requires a tight coupling of the metadata query system and the streaming server in order to make the global time to display as small as possible.

Applications requirements and user needs often depend on the application context. Professional users may want to find a specific piece of content (e.g., an audiovisual sequence) from a large collection within a tight

deadline whereas leisure users may want to browse the content of a video archive to get a reasonable selection. Mobile users may have limited terminals capacities (e.g., PDA) which do not support advanced user interface neither high or middle quality video streams i.e., due to the limited wireless network bandwidth, they may prefer to receive for example low quality streams.

So, distributed multimedia systems, in order to support searching and browsing video repositories, need to deal with several heterogeneities:

Context heterogeneity: in fact, number of multimedia applications impose their own a domain-specific customization. For instance, searching and browsing broadcast news is different from searching and browsing soccer archives or educational materials. So, it is essential for the success of a distributed multimedia system to cope with this domain heterogeneity.

End-user terminal heterogeneity: as mentioned above, mobile users terminals may have limited display capabilities, in addition to the wireless network bandwidth limitation. Allowing such users to access to the distributed video repositories requires that the system is able to adapt both the video streams and the user interfaces to support such users.

In this paper, we present the SIRSALE system, a complete indexing, searching and browsing distributed video repository system. The key idea of the SIRSALE system is to allow users, through graphical interfaces, to use enhanced searching and browsing tools, specifically tailored to support various domain data models. In practice, users have to choose the domain they want to search or browse (for example soccer matches or news archives) ; then the system automatically downloads the user interface and the data model related to that domain. So this approach allows managing video related to various domains in the same repository.

The remaining of the paper is organized as follow: section 2 discusses video indexing and browsing techniques and presents the multilevel indexing approach used in the SIRSALE system. Section 3 describes the architecture of the SIRSALE system and the structure of its main models. We then discuss the experiments we have conducted (section 4). Section 5 describes the relation between SIRSALE and video transport with Tamanoir active network technology. Finally section 6 summarizes the contribution of this paper and point out some future works.

2. VIDEO SEGMENTATION, INDEXING, AND BROWSING

Although audiovisual information is mainly in digital form, content-based video retrieval is still a very challenging task. In fact, in order to retrieve the information from a digital collection, we cannot search natively the raw data due to the nature of audiovisual data but only some kind of descriptions summarizing the contents. This issue remains an open issue even the research efforts done in this field.

To support video retrieval applications, the video must be properly modeled and indexed. Different methods have been proposed in the literature to analyze the structures of audiovisual data. Those methods could be grouped in two main approaches:

2.1. Segmentation Approach: low level

In the segmentation approach,[1] the video is divided into atomic units called *shots* using low-level features. Each shot consists of a visually continuous sequence of frames. The content of each shot is then described individually. With such segmentation, the retrievable units are low-level structures such as clips of video represented by key frames. Although such approach provides significant reduction of data redundancy, better methods of organizing multimedia data are needed in order to support true content-based video information search and retrieval capabilities. A concept structure is normally superimposed on top of the set of shots to provide necessary context information. Hence, a *scene* can be defined as a sequence of shots that focus on the same point of interest, while a *sequence* can be defined as s series of related shots and scenes that form a single, coherent unit of dramatic action.[2]

Current shot detection algorithms perform fairly well and are able to detect shot boundaries while scene detection remains, however, an open issue. Experts agree that scene detection requires content analysis at a higher level.[3]

2.2. Stratification Approach: semantic level

Instead of physically dividing the contiguous frame sequences into shots, the stratification approach focuses on segmenting the video's contextual information into a set of strata each of which describes the temporal occurrences of a simple concept such as the appearance of an anchor person in the news video. Since strata are linked to the semantic information within a video, they may overlap and thus the meaning of the video at any instance can be flexibly modeled as the union of all strata present.

The stratification approach is a context-based approach to modeling video content and therefore strongly linked to the domain to be considered. In fact, as noted before, the semantic model used to describe a specific domain (e.g., news, courses, ...) could not be re-used for another domain. Furthermore, stratification approach impose human's intentionality early during the semantic extraction stage (indexing stage).

2.3. Multilevel Approach

Multilevel models constitute an alternative for solution searching and browsing video documents by bridging the gap between low-level visual features and high level semantic concepts.[4] Indeed, as noted above, a video can be annotated in two ways: (a) a structural way which organizes the video into a set of sequences, scenes and shots. Though this structuration provides means to browse the content of the video even the lack of enhanced semantics, it does not allow advanced content-based searching and querying; (b) a semantic annotation which captures the semantic and contextual content of the video. On the contrary to the structural view, this kind of annotation allows advanced searching and querying capabilities.

In this paper, we propose data models based on such a multilevel approach (see figure 1).

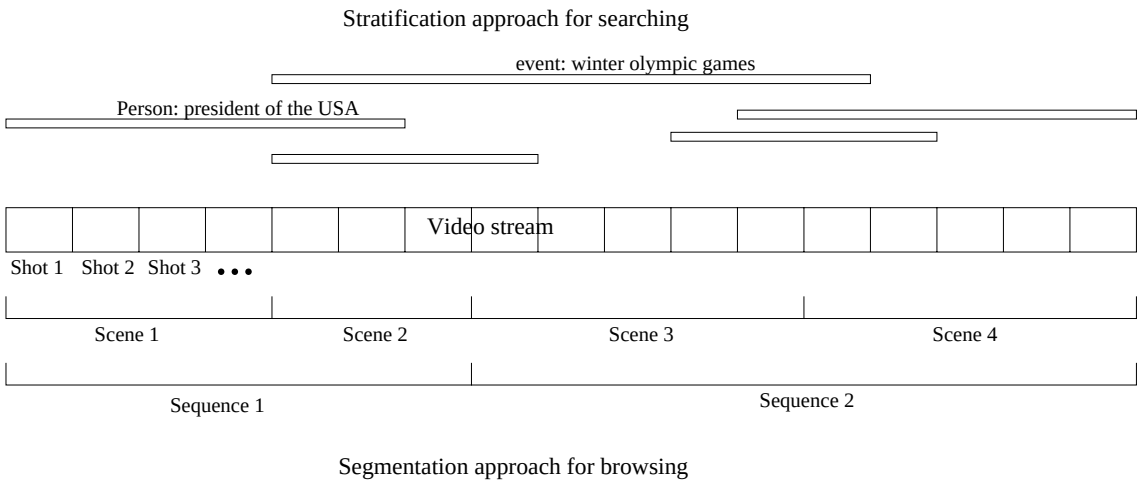

Figure 1. Multilevel view of a video stream

3. DESCRIPTION OF THE SYSTEM

This section describes the design of SIRSALE to support the whole process of video indexing, retrieval and browsing. We will discuss the data models, query processing and retrieval, video browsing, and video presentations.

3.1. System architecture

The overall architecture of the SIRSALE system is displayed in figure 2.

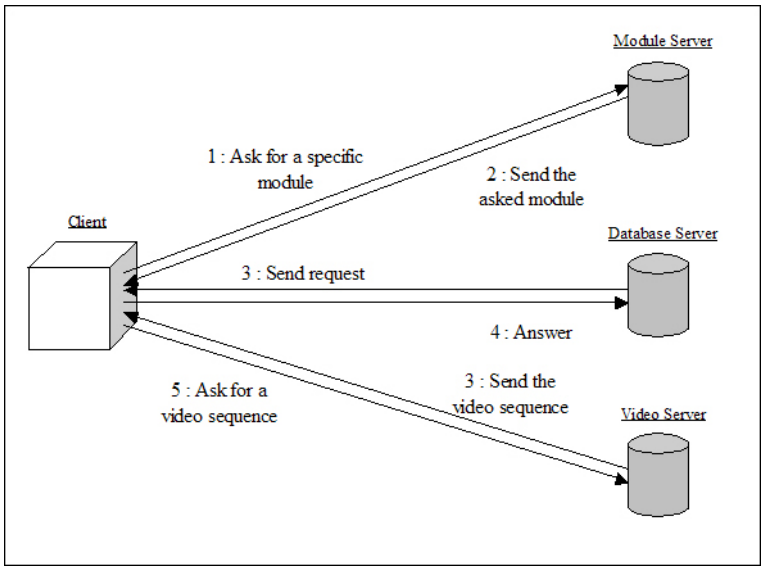

Figure 2. SIRSALE Architecture

3.1.1. Video Servers

We choose to use a video server based on the RTP protocol, that works on the UDP transport protocol, because it offers the most adapted services to transport real-time media. The RTP protocol can be used on TCP or UDP. but, UDP is more adapted to real-time data transport in long distance context. The experiences show that if some UDP packet are lost during the transport, they arrive to their target in a shortest time that TCP packet. For a NOD application, the speed of data transfer is more important than the lost of some data. The RTP protocol permits to optimize the network bit rate, because it doesn't need the use of a cache. The client reads the file as the server send it. So the client doesn't need a large storage volume and this model could be use on short storage system like PDA or mobile phone. In fact, the client ask for a video stream, the server found it, find the beginning frame and starts to send data. The client receives it as it comes and displays it on a player. We developed this video server with JAVA and, specifically, the JAVA Media Framework technology. This server provides a direct access to physical video streams and permits to access to a specified video segment. The beginning and the end of the sequences are specified to the server that sends the corresponding frames to the client. The video server also stored the video streams.

3.1.2. Metadata Servers

The database server contains the video metadata, like the streams length and the id of the video server used to send video streams. It also manages the low and high level annotations on the video streams. In order to enhance the fault-tolerance and the scalability of SIRSALE we have chosen to use one database per topic of interest (e.g. news, football, education...). Metadata bases can be distributed over several servers.

3.1.3. Module Servers

SIRSALE is based on a set of generic modules like the streaming or metadata servers and ad hoc modules that can be developed and integrated within SIRSALE in order to deal with the specific features of each kind of application and each topic of interest. For instance, a query interface for basic users for a database of football videos is fundamentally different from a query interface dedicated to professional users manipulating video repositories for editing purposes.

In practice, for a specific topic of interest, we group all the functionalities related to annotation activities together into one application package that is stored on the module server. The user interface provides low-level indexing, streaming and display functionalities since they do not depend on the video semantics. However the basic interface does not include any high level (semantic) functionalities since they are dependent on the topic of interest and the application features. So, after the user has defined his/her target topic of interest, it sends a request to the module server in order to receive the pertinent application package. This latter is then plugged into the interface so has the user can deal with a fully functional interface.

This approach combines several advantages. First, it allows providing a lightweight interface that can be adapted to the specific needs of the user. Second, it allows limiting the number of connections to the server. Indeed, all the events and the operations such as checking a mouse click, a button press, or requesting the databases for video sequences, are managed by the client interface. Finally, it makes the update and addition of packages, and by consequence the update of the user client, very easy : the administrator is only required to put the new packages in a specified directory to make all modifications accessible to the client interface.

3.1.4. Query Interface

The query interface allows two types of queries. The first part of the query interface allows browsing the video streams using the structured modeling approach. The second part provides the user with graphical tools that he/she can use to build complex queries based on the stratification of the video . The stream browser is represented by a navigation tree (fig. 4.1.3, bottom left). Each node of the tree is used to represent a structural indexing level. First, it specifies the compound units, followed by the sequences then by the scenes and the shots. The tree representation offers a friendly interface to quickly visualize and browse the different video documents stored on the servers. But, this browser can not be used to search specific information in all of the documents. To do that we propose to use a semantic query construction interface (fig. 4.1.3, top left). This semantic query interface is composed of two parts. In the first part, we provide an access to the annotation tools that depends on the video content. Actually, this part of the interrogation interface is not the same if we use a news research tool or a football research tool. On the second part, we allow the user to graphically construct the queries.

3.1.5. Annotation Interface

To define the video sequences, a user needs a dedicated player, with complete VCR functionality used to create the time sequence, and a set of objects used to annotate the sequence. Video data are downloaded to the user machine before they can be read by the player. We chose this approach because the definition of the time sequence must be very precise, so the video quality and bit rate must be as high as possible. By using a "local player", the accuracy of the time definition is only limited by the hardware performances.

4. PROTOTYPE AND EXPERIMENTS

We have developed a fully functional system that implements solutions described above. This system has been developed in Java, Java RMI and MySQL in order to be portable to various operating systems like Windows and Linux.

Two main experiments have been conducted on two different semantic contexts: TV news and soccer video. These two contexts show different constraints and use different semantic objects.

4.1. News Management

The news manager is used to index and retrieve news sequences. Currently, the query result of this kind of manager is not directly linked to the video information. Thus, the retrieval of news support could be very complex and time consuming. But, most users of this kind of information are television reporters who need a simple and quick retrieval tools. Hence, the first module we choose to develop is dedicated to this work: give a complete tool to retrieve and directly visualize news information.

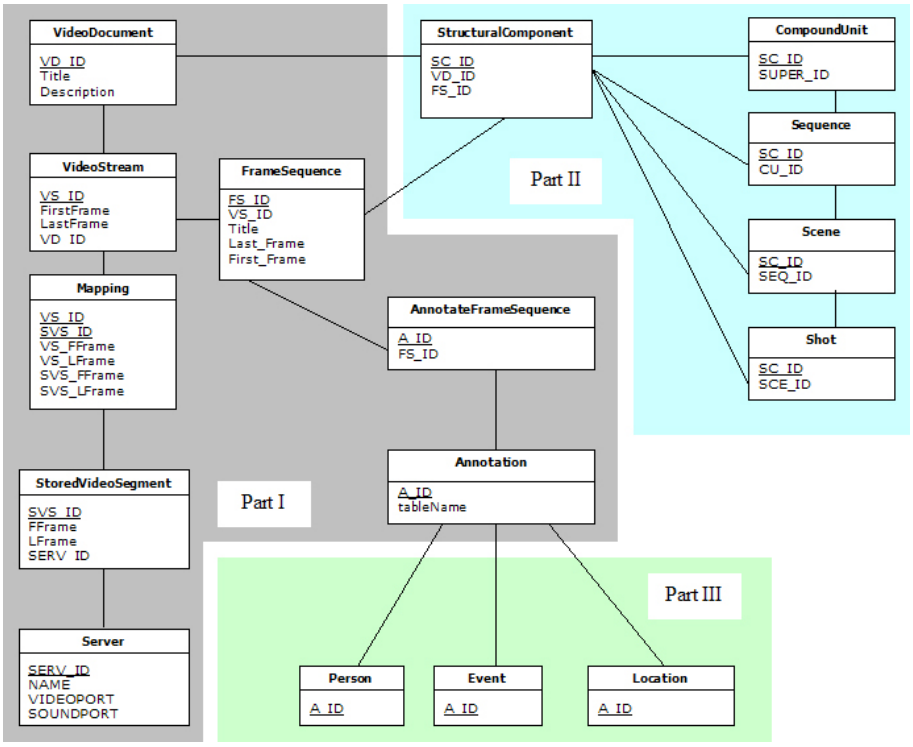

Figure 3. SIRSALE Data Model

4.1.1. Data model

Figure4.1.1, shows that our model has three different parts. The first part is used to get the physical description of each video document. It is used to retrieve and display the stored video segments in the video servers. The second part is used to structurally cut the video document. This describes all the different structural levels of the cutting process. These two parts are common to all the video semantic content and will be the same in all the different module databases. The third part is used to semantically annotate the video documents. As noted above, such annotations depend on the topic of interest. In the context of a news manager, we decided to annotate the persons, the events and the locations of each sequence.

4.1.2. Annotating

Figure 4.1.2, shows the annotation interface. It has three parts. On the left, the user selects the video documents and its video sequences he/she is interested in. On the right, the user can define the annotation entities (person, location...) which are concerned using a selection list. In the middle, he/she can view the video document. To get a full control on the video, VCR functions like play, pause, stop, fast and low forward and fast and low reward are provided.

4.1.3. Querying

Figure 4.1.3, shows the query interface. On the bottom-left corner, we see the video document browser with the different parts of the structural cutting. On the top left corner one can see the request constructor with the various annotation entities. On the bottom right corner, we find the result window and on the top right corner, the video player.

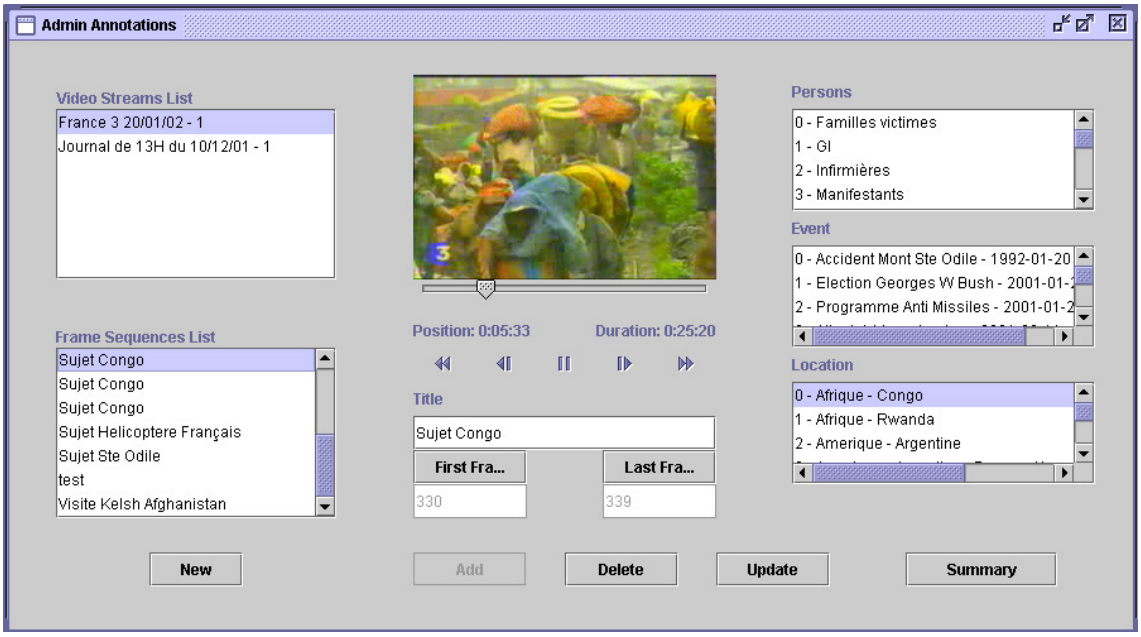

Figure 4. Annotation Interface

4.2. Soccer archives

Soccer archives can interest both "basic users" and professionals. Thus soccer clubs use this kind of archive to help training the players, since video films are currently used to visualize the game play of any opponent team. The video helps the manager to understand a specific action or just visualize the game play of a player.

4.2.1. Data model

The soccer model 4.2.1 has also three parts. The two models (news and soccer) use the same two first parts, but the semantic part changes because users do not need the same information. So, the soccer manager tool allows specifying information about players, teams, events, referees, stadiums. . . .

4.2.2. Querying

Figure 4.2.2, shows that the soccer query interface is nearly the same as that of the news interface, except the query constructor. Oppositely, the annotation entities are very different, adapted to a soccer-indexing tool.

5. ACTIVE NETWORKING FOR VIDEO DATABASE MANAGEMENT FRAMEWORKS

The integration of new and standard technologies into the shared network infrastructure has become a challenging task, and the growing interest in the active networking field[5] might be seen as a natural consequence. In "active" networking vision, routers or any network equipments (like gateway or proxy) within the network can perform computations on user data in transit, and end users can modify the behavior of the network by supplying programs, called *services*, that perform these computations. This kind of routers are called *active nodes* (or *active routers*), and show a greater flexibility towards the deployment of new functionalities, more adapted to the architecture, the users and the service providers' requirements.

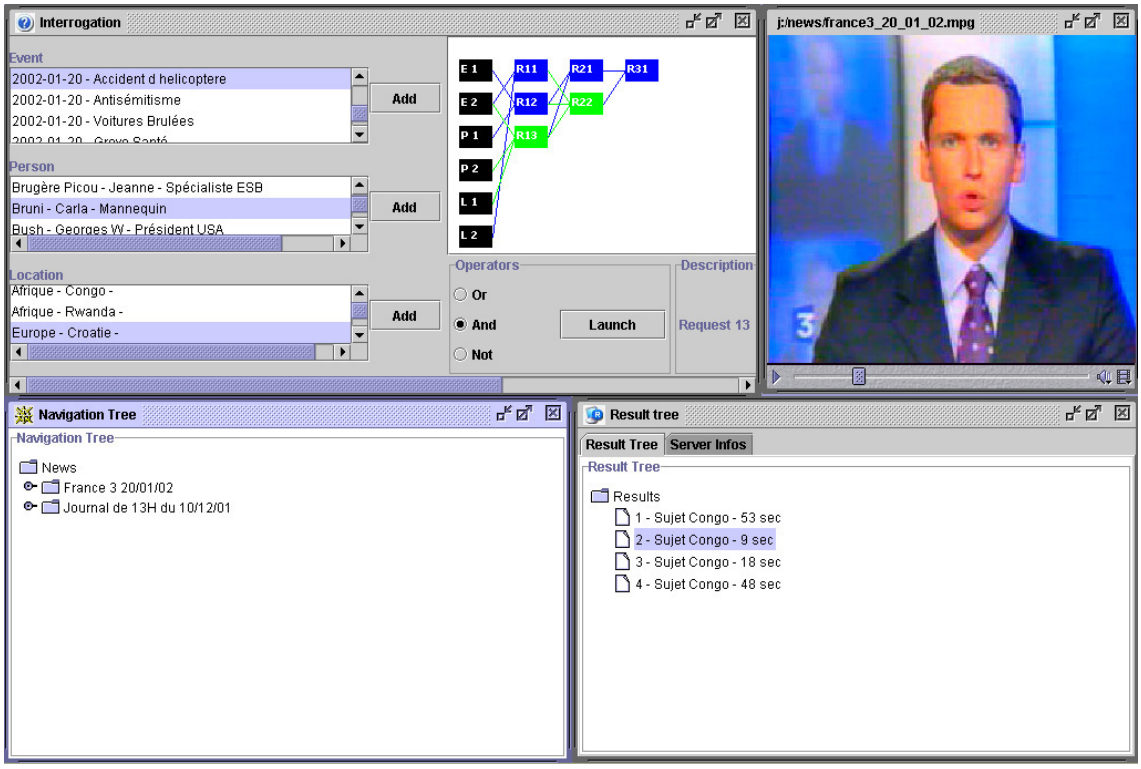

Figure 5. News Query Interface

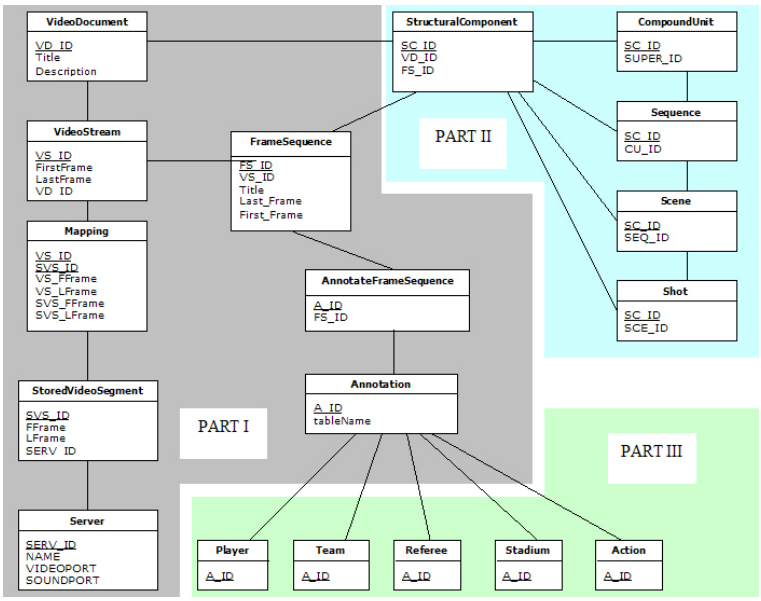

Figure 6. Soccer annotation model

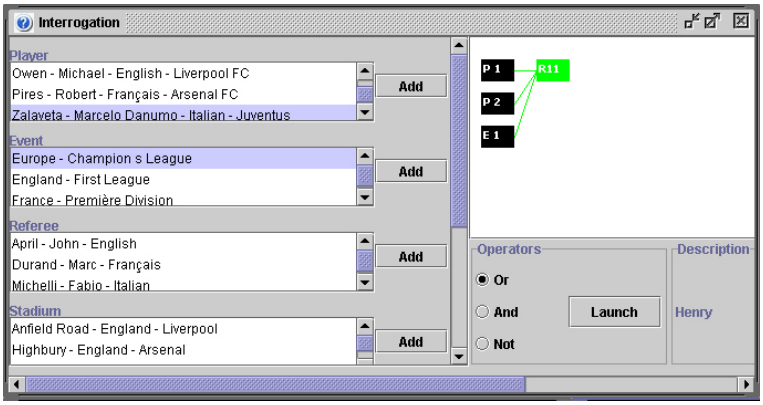

Figure 7. Soccer Query Interface

5.1. The Tamanoir architecture

The Tamanoir[6,7] architecture design does not interphere with the core network, mainly to guarantee higher performance results, and it's deployed only on the network periphery.

Tamanoir Active Nodes (TAN) provide persistent active routers which are able to handle different applications and various data stream at the same time. The two main transport protocol (TCP and UDP) are supported by the TAN for carrying data. We use the ANEP (Active Network Encapsulated Protocol)[8] format to send data over active networks.

The injection of new functionalities, called services, is independent from data streams : services are deployed on demand when streams reach an active node which doesn't hold the required service. There are two ways for service deployment: with a *service broker* (or service repository), where TANs send all requests for downloading required services, and without, in which case the TAN queries the active node that sent the stream for the service. Using a service broker introduces a single point of failure, but at the same time can be used to to check what services are spread all over the TANs in the Internet. When the service is installed in memory, it is ready to process the stream. It is worth noticing that a stream can cross equally a classical router, obviously, without any processing actions.

Figure 8 describes a TAN. The main part, called *TAMANOIRd*, redirects packets towards the adapted service in function of a hash key contained in the packet's header. New services are plugged in it dynamically. The second part, called Active Node Manager (ANM), is dedicated (1) to send services request to another TAN (2) to update its routing table.

For the implementation process of the Tamanoir execution environment we choose the JAVA language, because it provides great flexibility, typical of an Object-Oriented language, and is shipped with standard library. Moreover, recent JVM releases ($\geq$ 1.3.x) give excellent performance for the mainstream hardware architecture (i.e., x86), mainly due to the improvements in Just-In-Time (JIT) compilation techniques.

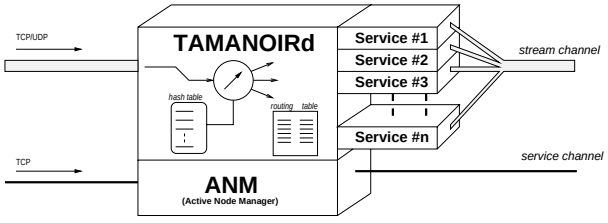

Figure 8. A Tamanoir Active Node (TAN)

5.2. Active Services Tool-box for Data Streaming and Multimedia

Tamanoir offers various network and high level services to multimedia and data streaming applications (Figure 9). These services, dynamically deployed, allow users and operators to manage multimedia streams.

Active available services can be classified into various categories (see fig. 9) : transport services (multi-protocols, multicast, QoS...), network services (content based routing, dynamic network management, monitoring...) and stream services (transcoding, compression on the fly, multi-codes...).

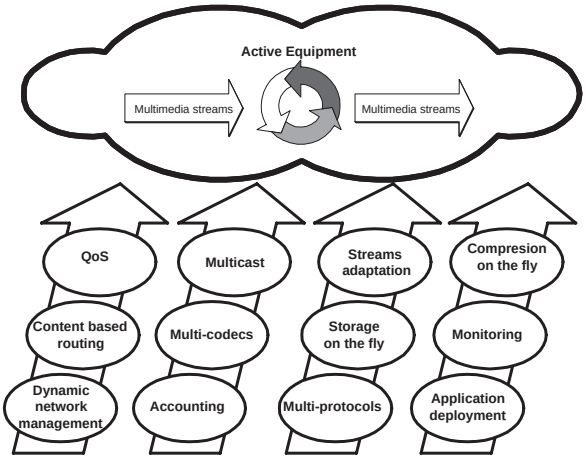

Figure 9. Active services for video

5.3. Video Adaptation

Todays' networks have to deal with three kind of heterogeneity. The first one is the *data* heterogeneity, which is depending by the applications requirements in terms of throughput, synchronism, jitter, error injected during transport, and so on. The second one can be identified in the *physical transport media* heterogeneity which, having different throughput characteristic, may limit the reliability and therefore introduce asymmetry. The last heterogeneity class comes from different *clients terminals* (PDA, cell phone, desktop, TV...), having different processing power and ability to restore informations. Active Networks gives us the opportunity to create and/or adapt transport protocols and configure it dynamically for SIRSALE Tools. They also give the capacity to process the data stream to adapt it for the client's processing needs and terminal heterogeneity.

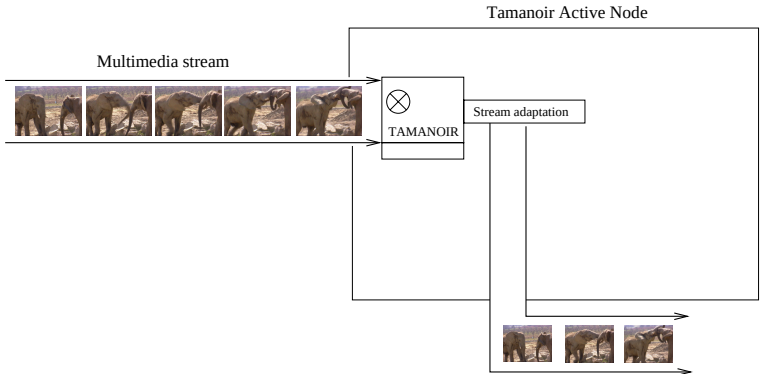

Figure 10. Active video adaptation

Figure 10 describes an active service for video adaptation adapted to SIRSALE requirements in terms of network and terminal equipments heterogeneity.

Services must be efficient and must guarantee an high performance to be able to sustain the bandwidth throughput without introducing too much latency. To achieve good results, we must use appropriate data structures for an efficient data processing. For instance, in video transfer we should use hierarchical codage like MPEG-4. A service embedded in the TAN is able to reduce the quantity of informations just by dropping the surplus of data unused by light terminal at a very light cost.

6. CONCLUSION AND PERSPECTIVES

This paper presents a novel content-based video indexing, searching and retrieving system, called SIRSALE. SIRSALE is based on a multilevel video indexing approach that combines both low level indexing which allows browsing of videos by structures (sequence, scene, shot) and hight level indexing that supports content-based video searching. Moreover, SIRSALE, through the management of modules, allows users to use domain specific data-models as well as user interfaces to search and retrieve video piece related to various domains.

Future evolutions of the SIRSALE system will first include a complete parallel video server that we have developed earlier[9] in order to support the huge volume of the video database as well as the high expected transactional workload. We are working to add a presentation tool which will allow users to construct their own multimedia presentations from requested video sequences.[10] We are following our integration of SIRSALE tools with active networks technology by providing new network services adapted to QoS requirements for video transport.

REFERENCES

1. B. Rubin and G. Davenport, "Structured content modeling for cinematic information," *SIGCHI Bulletin* **21**(2), pp. 78–79, 1989.
2. I. Konigsberg, "The complete film dictionary," *New York, Penguin* , 1997.
3. H. J. Zhang, "Content-based video browsing and retrieval," *Handbook of Internet and Multimedia Systems and Applications* , 1999.
4. J. Fan, W. Aref, A. Elmagarmid, M. Hacid, M. Marzouk, and X. Zhu, "Multiview: Multilevel content representation and retrieval," *Journal of electronic Imaging* **10**, pp. 895–908, 2001.
5. D. Tennenhouse and D. Wetherall, "Towards an active network architecture," *Computer Communications Review* **26**, pp. 5–18, April 1996.
6. J.-P. Gelas and L. Lefèvre, "Tamanoir: A high performance active network framework," in *Active Middleware Services, Ninth IEEE International Symposium on High Performance Distributed Computing*, C. S. R. S. Hariri, C. A. Lee, ed., pp. 105–114, Kluwer Academic Publishers, (Pittsburgh, Pennsylvania, USA), Aug. 2000. ISBN 0-7923-7973-X.
7. J.-P. Gelas and L. Lefèvre, "Mixing high performance and portability for the design of active network framework with java," in *3rd International Workshop on Java for Parallel and Distributed Computing, International Parallel and Distributed Processing Symposium (IPDPS 2001)*, (San Fransisco, USA), Apr. 2001.
8. S. D. Alexander, B. Braden, C. A. Gunter, A. W. Jackson, A. D. Keromytis, G. J. Minden, and D. Wetherall, "Active network encapsulation protocol (anep)." RFC Draft, Category : Experimental, http://www.cis.upenn.edu/switchware/ANEP/, July 1997.
9. C. P. A. Mostefaoui and L. Brunie, "Serveur de squences audiovisuelles parallle sur rseau haut dbit : concepts et exprimentations," in *RENPAR'11*, pp. 127–132, Rennes, France, June 1999.
10. A. Mostefaoui and L. Brunie, "Optimizing server i/o for multimedia presentations," in *IEEE International Conference on Multimedia and Expo*, (Lausanne, Switzerland), August 2002.