D2.2 STEER requirements and experimental environment architecture

D2.2 STEER requirements and experimental environment architecture

Project Number:	**Contract Number: ICT -318343**
Project acronym:	**STEER**
Project Title:	**Social Telemedia Environment for Experimental Research**
Instrument:	STREP
Theme:	FP7-ICT-2011-8
Report Due:	M12 (August 2013)
Report Delivered:	
Lead Contractor for this deliverable:	[UoP]
Contributors to this deliverable:	Spyros Denazis (UoP), Nikolaos Efthymiopoulos (UoP), Luigi Mori (ADBB), Mu Mu (ULANC), Omar Niamut (TNO), Otto Baijer (TNO), Stephen Phillips (ITINNOV), Jacco Taal (BTN)
Estimated person Months:	15.50
Start date of project:	1st September 2012
Project duration	27 months
Revision:	Version 1.0
Dissemination Level:	PU – Public

Table of Contents

Table of Contents 2
1. Executive Summary 4
2. Architecture Requirements 6
2.1. Storytelling requirements 6
2.2. Augmented Live Broadcast Requirements 7
3. STEER's Approach to Modelling Information 8
3.1. Social Graph 10
3.2. Media Object Graph 10
3.3. Media Distribution Graph Data Model 11
4. STEER Architecture in a Nutshell 11
4.1. Dynamic Community Management 13
4.1.1. User Management 13
4.1.2. Event Management 13
4.1.3. Community Formation 14
4.2. Social Media Analysis 15
4.3. Media Services and Statistics 15
4.3.1. Media Object Management 16
4.3.2. Media Services 16
4.3.3. Service Statistics 16
4.3.4. Media Object Graph 17
4.4. Media Distribution 17
4.4.1. Synchronization 17
4.4.2. Bandwidth Monitoring 18
4.4.3. Media Distribution Graph 18
4.4.4. P2P Congestion Control 19
4.4.5. Scalable Bandwidth Real-Time Management 19
4.4.6. Social Aware Caching 20
4.4.7. Media Streaming & Home Gateway 21
5. Major Functionalities and Relevant Interactions 21
5.1. Dynamic Community Management 21
5.1.1. User Manager 22
5.1.2. Event Manager 23
5.1.3. Community Manager 23
5.2. Social Media Analysis 24
5.3. Media Service, Statistics and Media Object Graph 28
5.3.1. Media Object Management 28
5.3.2. Media Services 29
5.3.3. Service Statistics 29
5.3.4. Media Object Graph 30
5.4. Real-Time Social aware Media Distribution 30
5.4.1. Synchronization 30
5.4.2. Bandwidth Monitoring 34
5.4.3. Media Distribution Graph 35
5.4.4. P2P Congestion Control 36

5.4.5. Scalable Bandwidth Real-Time Management 37
5.4.6. Social-Aware Content Placement and Storage Allocation 38
5.4.7. Media Streaming & Home Gateway 39
6. Augmented Live Broadcast System Architecture... 42
7. Storytelling Architecture 44
8. Conclusions 46

1. Executive Summary

STEER develops around the concept of innovative social and network-aware media distribution architecture that:

- coordinates the effective sharing of media and networking resources among participating users, ensuring scalability, stability and network friendliness,
- exploits features and capabilities of modern home gateways, such as caching and transcoding, to limit the cost of media distribution through resource sharing, dynamically adapting media exchanges to the capacity of the underlying networks, reducing service discontinuities and improving the overall Quality of Experience of the customer, and
- makes the best usage of information about media object geographic location, social relationships and media object dynamics (e.g., popularity changes) to offer an efficient and personalized social-aware search and recommendation functionality.

This document provides a detailed view of the STEER architecture, through the definition of its functional components, their purpose, mutual interactions and the applications that the project plans to develop with them.

The document is organized as follows:

- Section 1 delineates the structure of the document and the underlying rationale
- Section 2 analyses the objectives that the STEER architectures tries to fulfil, expressed as system-level requirements affecting end users behaviour, user devices, home gateways, involved service providers, and STEER applications
- Section 3 shows how STEER provides a way of organizing the wide range of raw information collected across various social media networks into a number of complementary, structured databases (or 'graphs') whose manipulation allows to achieve a deeper insight of the mutual relationships among user, events and data aggregates. This in turn sets the foundations for optimizing features such as recommendation, caching and media distribution.
- Section 4 introduces a short but holistic view of the STEER architectural components and their main interactions
- Section 5 describes the STEER components in more detail, explaining their purpose, their functionalities and the research objectives that motivate their usage, with relationship to the requirements set in Section 2.
- Section 6 and Section 7 describe how specific STEER components are exploited to implement the STEER Augmented Live Broadcast and Storytelling applications, respectively, and the way they address the use cases described in Deliverable D2.1. The user interfaces for both applications has been defined in Deliverable D2.3.

- Finally, Section 8 summarizes the main achievements that the STEER architecture will make possible in the short term, and provide some indications about their future evolution.

2. Architecture Requirements

This section analyses the requirements that underpin the STEER architecture, depicting the rationale on which the overall system has been constructed and underlining the influences that the analysis of the current technology status, developing goals, and identified use cases have set on its design.

As shown in Figure 1, the STEER architecture is defined on the basis of requirements that the STEER partners have distilled from a harmonized set of specific research and business objectives. Such objectives are originated by the desire of enhancing or extending system components that the partners have developed, or started to develop, complementing their functionalities with the integration of additional elements available from the affiliated FIRE projects. These factors determined the structure of the STEER proposal and, subsequently, the identification of the STEER Reference Use Case and the design of the STEER applications.

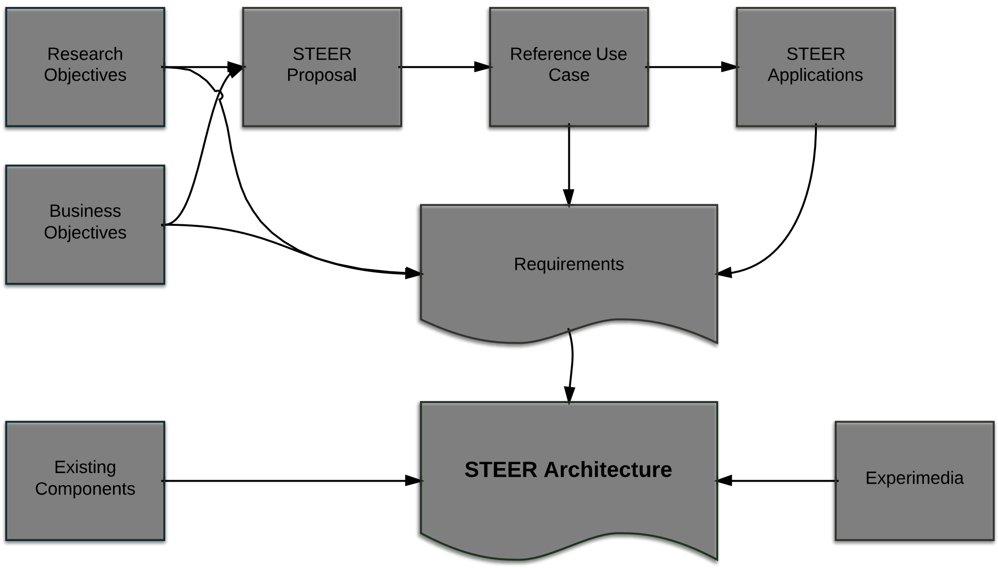

Figure 1. Dependency Graph for the STEER Architecture

The requirements have been organized in two main groups, corresponding to the two applications identified by STEER, i.e. Storytelling and Augmented Live Broadcast, as outlined in Deliverable 2.1.

An overview of the major requirements is presented in the following sections.

2.1. Storytelling requirements

- **User experience.** Unlike many other multimedia applications, STEER Storytelling is highly interactive as it involves content retrieval and inter-operation between users (i.e., storytellers and story-listeners). The user experience relies on the careful design of the system components that implement the user interface, content discovery, selection and retrieval, video story creation and manipulation, and smooth story playback, especially in an environment where network resources can fluctuate.

- **User interface**. Various types of user devices with different software and hardware specifications (including e.g. the availability of a video camera or a touch screen display, the display size, the type of internet connectivity) may be employed at different stages of the story telling process. Dedicated user interfaces must be designed for STEER-defined reference user devices.
- **Content retrieval**. It is essential for story creators to efficiently locate media objects related to an event or shared within a certain user community. This needs to be enabled by advanced content search and recommendation systems, which incorporate social informatics as a key metric.
- **Responsive story making**. Editing video stories often involves sophisticated multimedia processing such as cutting, joining and remixing audio and video tracks. Such data processing greatly affects the responsiveness of the Storytelling application; with the risk of deteriorating the overall user experience. Novel designs should be employed to limit the amount of media processing at end user device
- **Smooth story playback**. Stories may be displayed on user devices that make use of different types of Internet connections and display qualities. Therefore, it is required to employ multi-bitrate story archiving and quality independent story editing. Content caching infrastructure is also seen as a potential helper to improve media object distribution with respect to start up delay and video quality. Media transcoding capability in the home gateway can help in dynamically adapting the playback characteristics to both network conditions and display resolutions, optimizing the usage of available transmission bandwidth and/or storage/caching resources.
- **Synchronisation**. A complete story may contain multiple media tracks including video, audio, voice and text. Synchronized playback of these media tracks is therefore important to present the story to a listener in the same way that the storyteller originally created it.
- **Reliability and Scalability.** The number of user generated media objects and stories can increase drastically in a creative user community during a social event. Performance and reliability measures should be taken at the backend system to support community activities during large events, such as the Schladming ski championship taken as a representative example in Deliverable D2.1. Besides special designs in the user applications, technologies such as bandwidth monitoring, network management, proactive caching, and media transcoding would be contributing factors for the system reliability and scalability.

2.2. Augmented Live Broadcast Requirements

The major requirements that a social media distribution system, such the one employed in the Augmented Live Broadcast application, imposes on the underlying network are:

- **Efficiency and quality of video distribution**. This attribute aims at achieving the highest possible utilization of the available resources (media servers and peers), to obtain a media distribution system with an attractive trade-off between video quality and cost.
- **Stability**. The total available bandwidth of the distribution system changes quite frequently due to the time-varying performances of the underlying communication networks and the fluctuations in the rate of user arrivals and departures. An efficient live steaming system must be able to monitor and react to these dynamic conditions, in order to preserve as much as possible the stability of the offered Quality of Service (QoS) and Quality of Experience (QoE).
- **User experience.** Besides the technique and service itself, the way the user experiences the complete system, including the network performances, needs to be addressed. User experience is the result of the factors mentioned in the previous two requirements, combined with the ease of using the system, all influencing the way the user perceives the offered functionality.
- **Scalability.** This characteristic is determined by the amount of bandwidth, storage and processing capabilities that management and media servers must contribute as the number of participating users and media objects grow. For the design of a scalable system, distributed and real time management and control architectures are required.
- **Media distribution friendliness.** This feature is a requirement for the underlying network, realized through the control of the amount of the generated traffic and the provision of means for avoiding overload of the network segments interested by data transfer.
- **Synchronisation**. Synchronized playback of both broadcast and user-generated video flows is required when multiple streams are presented simultaneously to the viewer, whether on a single primary screen or in a multiscreen setup. Synchronization requires the insertion, at a location close to media originating point, of timing information referenced to a common, shared timeline.
- **Smooth media playback**. The aggregated live broadcast and user generated media may be displayed on user devices that employs different kinds of connectivity to the Internet or the broadcast distribution network, and have screens of varying quality. Introducing off-line and real-time media transcoding capability in network nodes and home gateway allows adapting media streams characteristics such as bitrate and video resolution to the specific playback conditions and the time-varying network capacity. This optimizes the usage of the available transmission bandwidth and/or local and geographically distributed storage/caching resources.

3. STEER's Approach to Modelling Information

STEER is, in essence, a community-centric social media ecosystem where communities of connected individuals, or 'users', are engaged in frequent interactions that involve the creation, exchange and manipulation of various media related to social events. This ecosystem is immersed in a sea of information generated by a wide range of social media sources such as Facebook, Twitter etc. Dealing with such huge amounts of dynamic and diverse data, recently referred to as 'Big Data', is a highly complex and challenging task.

Accordingly, one of the key challenges that STEER needs to address is how to distil from various social media, semantically rich, refined, community-, or event-related information that captures the subtleties of human behaviour and preferences. The problem is further complicated by the fragmentation and representation of a wide range of raw information in the social media cyberspace, which is usually locked up in platform- and application-specific silos. This gives rise to the ***Raw Information Stratum*** as depicted in Figure 2.

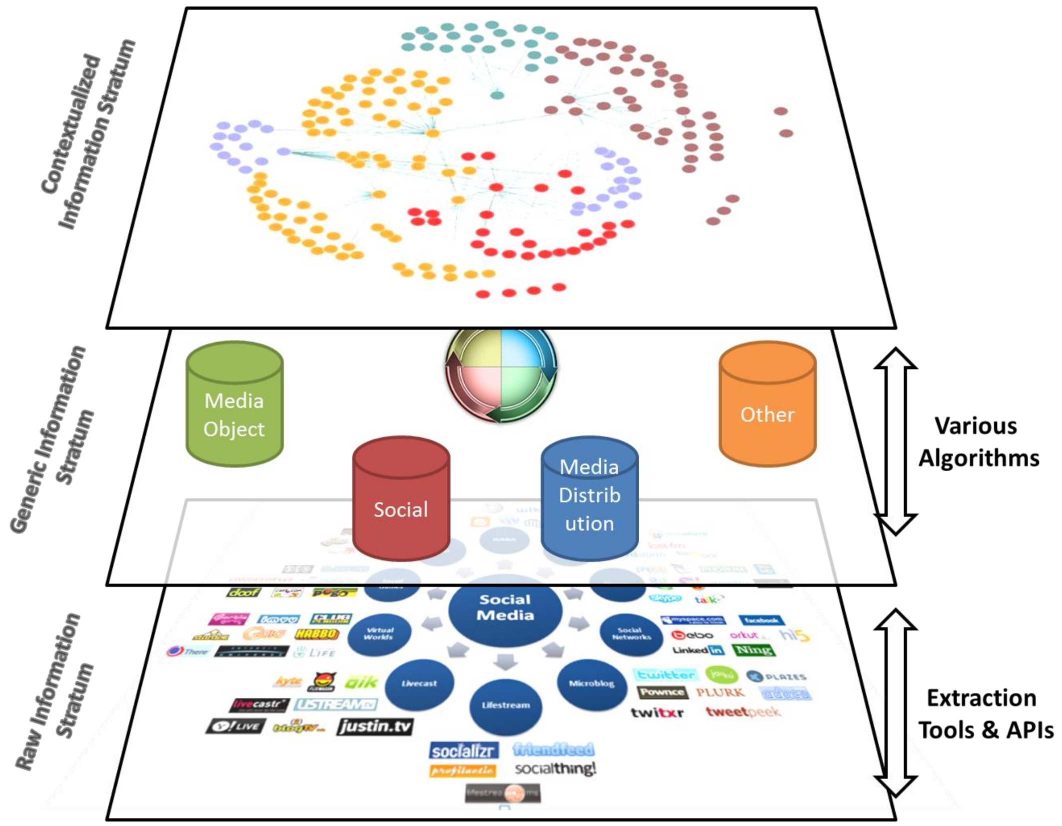

Figure 2. STEER Information Modelling

Using various tools and APIs this information can be extracted and further processed by sophisticated data algorithms to model, organise and store this data in various forms that facilitate its manipulation, refinement and more advanced analytical processing. To this end, the raw information may be classified into various categories, each one using a specific structure for its representation. A ***Generic Information Stratum*** may therefore be formed that offers a more organized information representation, structured in generic categories.

Having created a stratum with discrete, flat, groupings of information facilitates the discovery of contextual information represented in the form of associations dictated by the specific context. In other words, the context creates another stratum of derived and inter-related information, which we call ***Contextualized Information***

Stratum. It allows for highly sophisticated and intelligent analytics algorithms and tools that go beyond lexicographic processing, as they will discover correlations, relationships, dependencies and patterns that are relevant to specific contexts leveraged by media events and the communities built around them.

We see the formation of the two information stratums, raw and generic, as a highly dynamic process that change over time along with user habits and community size. Therefore, the information that lies within each stratum requires constant maintenance and updating, as obsolete information may lead to unreliable recommendations and search results. In this respect, the formation and active participation in various communities constitute sources of newly generated information that can be added to the existing pool of data and contexts, thus expanding and refreshing their contents. From this point of view, the two stratums are inter-linked and are constantly influencing each other, fed by the dynamics of the communities built around various events and the constant generation of information.

We note here that the proposed approach along with the methodologies to organize and model each stratum is a research challenge that falls outside the scope of the STEER project. STEER focuses instead on three major categories of information extracted from the Raw Information Stratum, which assume the structure of graphs, namely, the Social Graph (SG), the Media Object Graph (MOG) and the Media Distribution Graph (MDG). The aim here is to experiment with the manipulation of these graphs to create associations among the information according to the context (the event and the community) where the information is used. These associations are quite useful to the various functional components of the STEER architecture, e.g. recommendation, caching, media distribution. They heavily contribute toward their increased performance and accuracy. For instance, learning that a media object is becoming popular within a specific community from information provided by the SG may result in the effective deployment of media caches to the right home gateway or in the selection of a suitable user from the MDG to assist the distribution of the media content.

3.1. Social Graph

STEER aims to derive and model social context information, including user preferences, social network peers and friends, and sentiments. The social media analytics layer performs analysis on the information distilled from selected social networks and creates the Social Graph (SG). The social relationship between members of the STEER community is captured by the SG. It not only inherits 'explicit' social connections from existing social networks such as Facebook, but also discovers 'implicit' relations among individuals by matching their preferences and activities in social applications. Joining the explicit and implicit social relationships can greatly improve the density and diverseness of the SG.

3.2. Media Object Graph

The Media Object Graph (MOG) maintains the faceted relationships between media objects that are shared and requested by STEER users. In the MOG, a relationship between two nodes (media objects) depicts their linkage in one defined context. By exploiting semantic context in multiple dimensions, nodes in the MOG can have multiple relationships among each other with respect to e.g. their ownership, the way they are requested by community members, their genre and age. A highly populated multi-mode MOG enables node-to-node traversal through a composition of relationships (contexts). With the help of graph analysis tools, MOG is used to recommend 'related' media object to end users and to estimate future user requests based on service statistics.

3.3. Media Distribution Graph Data Model

The Media Distribution Graph (MDG) can be seen as tool that can produce low level information-statistics associated to the behaviour of the users in the consumption of each media object, and information-statistics related to the underlying network features and dynamic conditions. In more detail, the MDG is able to generate information connected to:

- The time interval when each user consumes each media object
- The time instant when each user consumes each media object
- The bandwidth resources of each user
- The network latencies between users
- The quality and stability of the connection between users

In STEER, a specific service will be able to deliver this information to any application that can effectively exploit such knowledge for various purposes, such as the monitoring, control and allocation of resources, the determination of social-popularity of media objects, the network-aware media search and recommendation, and social aware caching. Algorithms that are developed in STEER and exemplify the convenience of this service will be described in detail in the following sections.

4. STEER Architecture in a Nutshell

This section briefly presents the major components of STEER and the major interactions among them, also highlighting the main experimentation goals and providing some hints about relevant monitoring and experimentation tools.

A comprehensive view of the architecture of the whole STEER 'system' is given in Figure 3, which groups components in a number of principal subsystems that will be further detailed in the following subsections. In particular, individual functional elements will be referenced in capitalized *italics* in Section 4.

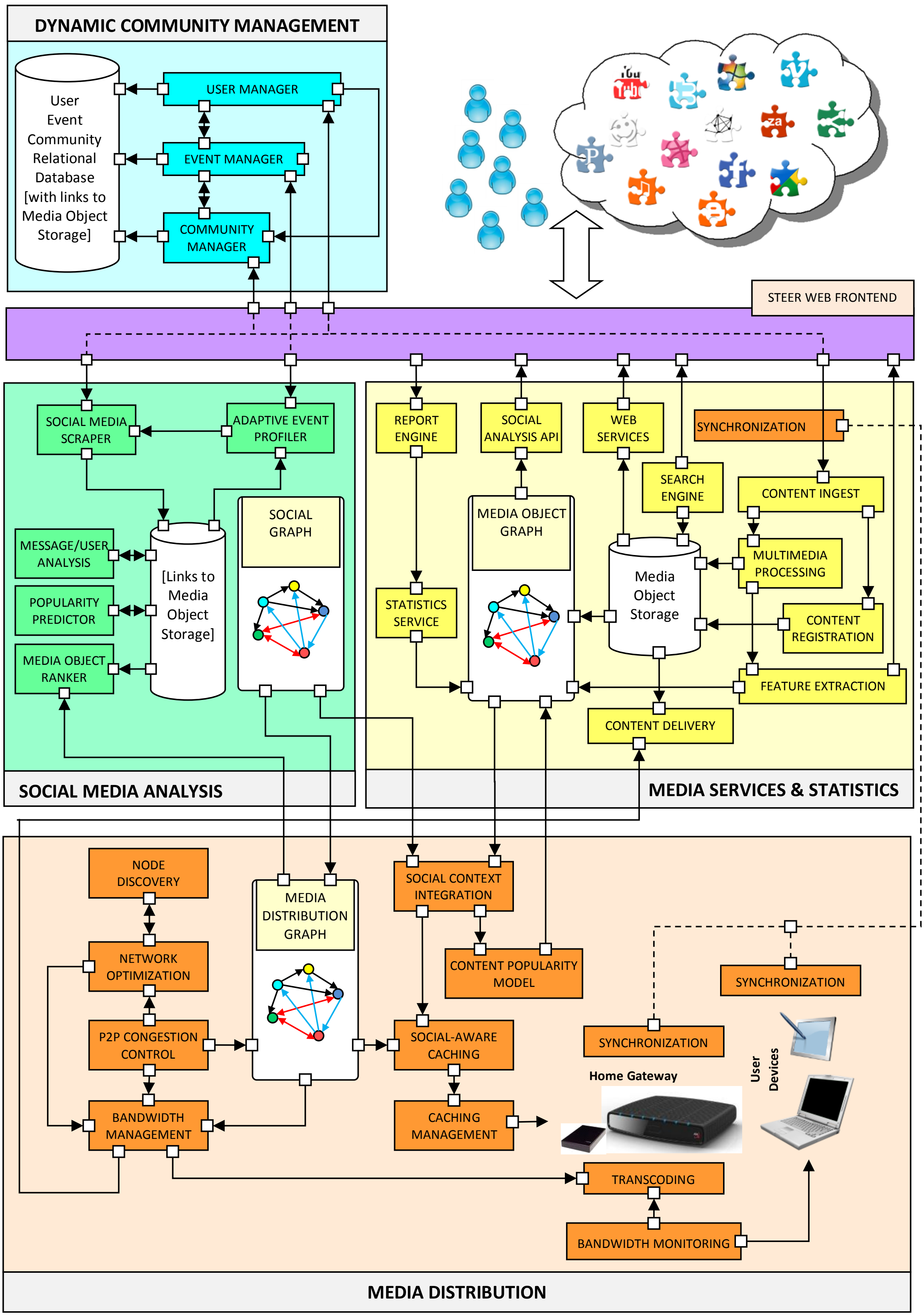

Figure 3. STEER Architecture

4.1. Dynamic Community Management

The social network management modules presented in this section support other modules in the STEER system with essential functionality required to obtain user data, connect to social networks and get information about events. Although these modules are innovative and not at all straightforward to implement, no specific research question is associated to them. In the context of STEER only the necessary operations will be implemented in order to facilitate the experiments.

4.1.1. User Management

The STEER approach to modelling presented in Section 3 greatly depends on linking user account data to other user accounts (Social Graph), media objects (Media Object Graph) and logging (Media Distribution Graph). It is essential that the STEER system has a clear and unambiguous way of representing a physical user, and of correlating the user with media objects, other user accounts and with entities in third party services.

Since most services and social networks in particular, have their own way of internally representing user accounts, the STEER system needs a means to correlate these different user representations to each other.

Furthermore, since registration and access to third party services normally requires a login procedure, using the STEER system can be cumbersome if the user has to repeat this procedure for every service separately. Single-Sign-On functionality eases this process, requiring the user to only login once at a popular social network: STEER components and services will then rely upon the so called OAuth2.0 method to authenticate the user, without requiring additional login process.

This brings two other advantages, namely:

1. The user does not actively have to create an account for STEER services: approving access by the STEER system to his/her social network account is sufficient
2. The STEER applications can automatically connect and tap into the users social network, in a secure and controlled way[1]

To support this feature, the STEER system integrates a *User Manager* component. The *User Manager* maintains a user database, containing basic user information (name, gender, etc) and the credentials for third party services and social networks.

STEER components can register with the *User Manager* to be able to obtain and query user information as needed for their operation.

4.1.2. Event Management

Events play an important role in the STEER Use Case and in the STEER system as a whole. They are defined by a moment in time and a bounded location and can have properties such as a name, organizer, etc. Events are the most central data objects

[1] Social networks allow third party applications to tap into a user's social network, if and only if the user has granted access to such an application. The user can revoke this access at any time.

in the STEER system. They connect to other data objects such as users, media objects and communities. STEER events are created explicitly, by a user who 'starts' a new event, for example when he records and uploads a first video for the event. The user provides all relevant data for the event, such as a name, time and location. Other users can pick up this new event when it is recommended by the recommendation system.

Using information from the event database, the STEER system can correlate events with STEER media objects, related social media objects (e.g., Facebook posts, tweets, groups etc) and users, through the *Adaptive Event Profiler* (see Sections 4.2 and 5.2). STEER users can 'subscribe' to a scheduled live event or experience an ongoing live event. Users can be at the actual event physically, or at home, experiencing the event on their TV or mobile devices.

The STEER *Event Manager* supports querying the event database for event information, related media objects, related communities and related users.

4.1.3. Community Formation

All STEER users that show interest in an event form together an event community. Each community member might have his/her personal explicit social network (Twitter, Facebook, YouTube, etc.). Social media activity of community members and at least their first-order explicit network connections will be real-time monitored for relevant content by a *Community Manager*. A relevance weight is assigned to users based on their distance to other users, where community members are considered zeroth-order connections of each other. STEER aims at combining communities from different social networks. The STEER *User Event and Community Relational Database* stores references to these social networks.

Figure 4 below depicts the relation between STEER communities and entities on existing social networks.

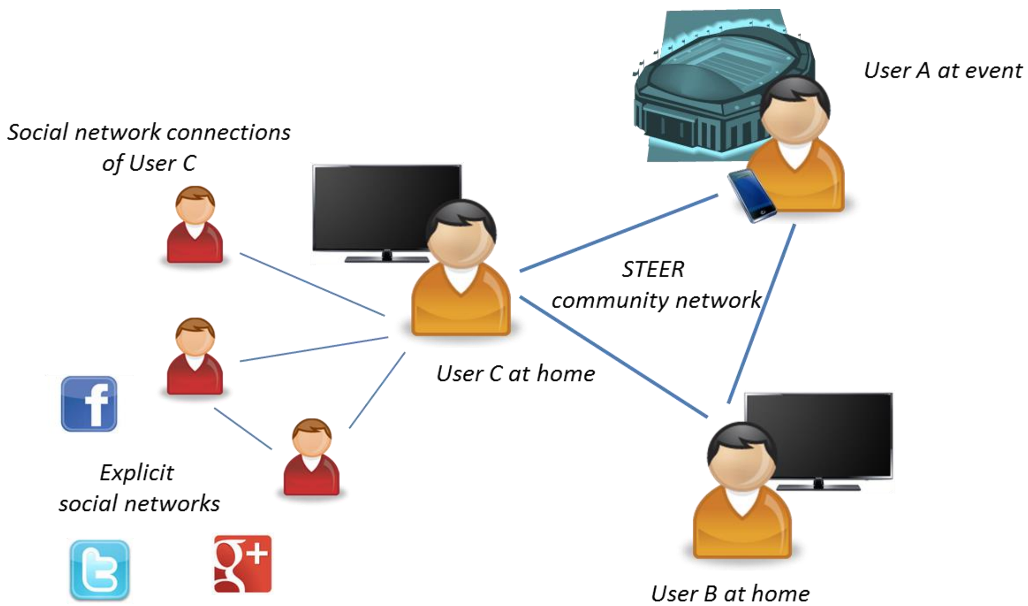

Figure 4. Relations between STEER communities and entities on existing social networks.

Since STEER encourages cooperation and social interaction, users can be made aware of the activities in the STEER system by people they know, or by people they do not directly know but are correlated in other ways. For instance, people may be related because they share the same interests, attend the same events, etc.

4.2. Social Media Analysis

The STEER system captures social relations in a Social Graph (SG). Therefore, the STEER architecture incorporates functional elements that

- Analyse social networks and extract relevant social information
- Determine and detect a user's social context, emerging social events and social communities
- Enable social context-aware personalization and ranking, that allow for personalized, media object search and recommendation based on social context models and sentiments
- Construct the STEER social network graph, which captures the essential information about, and relations between, social entities.

STEER has two approaches to determine and derive SGs.

- In the 'explicit' approach, the STEER system harvests social connections directly from social networks (*Social Media Scraper*). By converging social relationships from multiple sources, users are clustered and categorized to build an SG.
- In the 'implicit' approach, the STEER system harvests messages and posts from social networks together with information about user activities on content services, to determine users and their social connections that can constitute an SG (*Adaptive Event Profiler, Message/User Analysis)*.

Further elements for social media analysis within the STEER system aggregate over the data extracted from social networks (i.e. entities and media objects). The *Media Object Ranker* ranks media objects based on sentiments expressed in social network messages, the number of times they are linked, their relevance weights, and so on. The *Popularity Predictor* predicts the popularity of entities and media objects by looking at early sentiments, number of times they are linked, authority of users involved, and so on.

Section 5.2 further describes the social media analytics components necessary for these approaches.

4.3. Media Services and Statistics

The STEER applications provide media services for users to share, publish and exchange media productions such as video recordings in the context of social events. Furthermore, user and system statistics are recorded by end-systems and forwarded on to a statistics server. This enables the analysis of usage and social activities in media services. Statistics are also aggregated and maintained in the form of graph representation to enable advanced social analysis.

4.3.1. Media Object Management

The media object management is required for multiple purposes. It creates a shared space for STEER users, whether they are professional content producers or amateur individuals, to maintain their audio-visual creations. A set of interfaces are required to allow uploading multimedia content (e.g., video recordings and texts) in arbitrary formats and sizes to a persistent storage (*Content Ingest*). Users should be able to keep track of their uploads and commit relevant changes such as titles and annotations at any time. The ownership and access control of user uploads is also maintained by the management service. Unique media object ID should also be generated for every user uploads to identify and manage media assets throughout their life-cycle (*Content Registration*).

The service should also process user uploads so that all media objects in the STEER system conform to a set of pre-defined specifications, e.g. in terms of audio and video codec and metadata schema (*Multimedia Processing*). This assures the consistency in media playback and data representation across multiple types of user devices. Media objects are hosted in mass persistent (cloud) storage (*Media Object Storage*).

To enhance the user experience in content service, certain features of media object should be extracted upon content ingest (*Feature Extraction*). For instance, key frames of a video object can be extracted to improve the search functionalities.

4.3.2. Media Services

Content sharing in social networking environment requires efficient content search, retrieval and delivery. The media services must offer a web-based service portal through which the service functions are delivered (*Web Services*). Users can browse and search existing media objects using multiple criteria (such as search keywords and results filters). To enable such user interactions, a search engine is required to index content and conduct multifaceted search (*Search Engine*). As is defined by STEER, the user community can form in the context of social events as characterized by a combination of time and location. While a conventional search function allows specifying such 'time and location' by means of keywords, STEER envisions a more advanced media service that adopts location-awareness using geographical metadata. When a media object or a composite of media objects is requested, dedicated media servers or P2P distribution infrastructures are responsible to stream corresponding content to user devices (*Content Delivery*). Therefore, dedicated chunking and encapsulation mechanisms should be in place and streaming servers such as a P2P seeding server or HTTP streaming server should be implemented.

4.3.3. Service Statistics

One of the main objectives of the STEER project is to investigate the exploitation of social informatics to enhance media services and media delivery. Besides social relations in social networks such as Facebook and Twitter, user activities such as requests and ratings on media content in conjunction with content metadata in media services are also believed to be invaluable and more deterministic social information for the purpose of STEER researches.

The collection and management of user and service statistics is not a trivial task. Reporting mechanisms must be implemented at the user client to monitor and report user statistics. A highly efficient and robust statistics service needs to be designed and maintained to receive, process and host statistics data from a large amount of concurrent user sessions (*Statistics Service, Report Engine*). The statistics should also be stored using technologies that are suitable for social network analysis. The statistics service is also expected to provide social analysis API to other service elements such as the search engine.

4.3.4. Media Object Graph

While the social graph gives insights into the relationship between STEER users in social networks, the Media Object Graph (MOG) explores the relationships between media objects shared by STEER users. The relationships incorporated by the MOG should be multifaceted and cover various aspects of a media service. A piece of content can be related to others with respect to their ownership, the way they are requested by community members, their genre and age, etc. The MOG must be able to combine these relationships and form a multi-mode graph. The information distilled from the MOG using graph analysis tools can be used to recommend 'related' media object to end users and to estimate future user requests based on service statistics (*Social Analysis API*).

4.4. Media Distribution

4.4.1. Synchronization

As part of its innovative social aware media distribution architecture, STEER foresees the need for inter-destination media synchronization (IDMS) mechanisms for synchronized media playout and social media communication across multiple devices and locations (*Synchronization*). STEER aims at developing and experimenting with an IDMS system in combination with current over-the-top video delivery mechanisms such as those being supported in current smartphones and tablets.

Therefore, the STEER architecture will incorporate functional elements that:

- Enable IDMS-type functionality to be used in combination with current over-the-top video delivery mechanism such as those being supported in current smartphones and tablets
- Allow for different tablets and smartphones to use timing information to synchronize media play-out across devices, locations and protocols
- Interface with other STEER architecture components, such as the network monitoring and the external social and media-aware environment
- Ensure consistent media synchronization when integrated into the STEER experimentation environment
- Enable the positioning of start/end points of audio and video objects from multiple media tracks
- Provide synchronization at story viewers as directed by a storyteller

- Provide synchronization of both professionally created and user-generated content in a multiscreen setup.

To achieve these goals, STEER employs a heterogeneous media object synchronization approach that includes a synchronization engine supporting media object synchronization on one or more devices over hybrid delivery protocols, e.g. local files, DVB(-S), HLS, MPEG-DASH. This requires the embedding of timing information at media ingest points and timestamp modification at play out or intermediate rendering positions. Additionally, STEER considers a timed interactivity approach for synchronized media play out with second screen interactivity control.

4.4.2. Bandwidth Monitoring

'Bandwidth' as defined within STEER is the maximum available or necessary data capacity in an end-to-end chain between the service located somewhere in the cloud or network and the application that the user is running from a device at his home.

Generally, the operators and service providers monitor very closely their networks and services on available bandwidth and take the respective corrective measures to prevent degradation of the bandwidth in their networks. The most likely part of degradation in the complete chain will occur in the user home network, between the home gateway and the user device. The home network is, from a monitoring point of view, a very complex network due to a multitude of connected devices and connection techniques. To deal with this complex part of the chain, STEER introduces a bandwidth monitoring tool, called ALLBEST, to make sure that the QoE or QoS of the complete chain is assured.

The *Bandwidth Monitoring* component is able to inform the home gateway about the available bandwidth and capacity between the home gateway and the end device, giving the home gateway, or any other client application that can access such information, the possibility to make decisions on e.g. streaming priorities, optimal transcoding settings or even decide to redirect traffic to other devices with more bandwidth.

The Bandwidth Monitoring component will be integrated into the home gateway.

4.4.3. Media Distribution Graph

As already stated, STEER envisages experimentation towards an innovative social aware media distribution architecture where resources from participating users and clouds are jointly optimized towards a scalable and stable real time media distribution system. The Media Distribution Graph (MDG) is a graph in which each node represents a user or a media server (cloud) and each arc that connects two nodes represents the unidirectional or bidirectional transmission of video blocks between these two nodes.

In contrast with the existing media distribution graphs, STEER designs an innovative social & network aware MDG (overlay) that will:

- Exploit the social informatics to make the media distribution more stable than the socially agnostic media distribution graphs used today

- Be able to connect each node (user) with a user close to the underlying network and to exploit the media distribution servers (clouds) only in case that the home gateways have not enough upload bandwidth resources: in this way, we aim at an effective management of the resources of the home gateways
- Discover and select network nodes employing totally distributed algorithms, to ensure scalability in terms of users and media events that STEER is able to support (*Node Discovery*)
- Optimally exploit the available dynamic and heterogeneous bandwidth resources of the home gateways while minimizing the traffic that media distribution introduces to the underlying network (*Network Optimization*).

To achieve this, the MDG will be dynamically informed by a *P2P Congestion Control* element about the available upload bandwidth of each participating peer. Additionally, it will monitor, aggregate and report to a real-time *Bandwidth Management* function the total available bandwidth resources of each overlay. Finally, it will dynamically use home gateways as caching components and exploit the resources of users that are not interested directly in a media object but are socially connected with the users that interested in it.

4.4.4. P2P Congestion Control

Network traffic in P2P real time streaming is heading towards multiple network destinations (nodes) and it is composed of small video blocks requested by peers at arbitrary time, according to a distributed P2P block scheduler. Under these circumstances, the congestion control requires a specific implementation that goes beyond the traditional TCP congestion control algorithms. STEER develops and experiments with an innovative congestion control algorithm that:

- Will be able to optimally exploit the available bandwidth of home gateways and media servers
- Will be stable and robust under time-varying delays and dynamic underlying network traffic
- Will be fair, in terms of allocated network resources, to other 'traditional' TCP flows that exist in the underlying network, in order to allow other applications to co-exists with live streaming and avoid starvation
- Will avoid overloading the buffers of the home gateways and the routers in the underlying network (BufferBloat problem)
- Will avoid unnecessary high latencies and retransmissions

The *P2P Congestion Control* element will be able to provide in real time the application layer with updated information about the network conditions and the amount of idle network resources. This will allow exploiting the available bandwidth resources to a higher degree and increased stability.

4.4.5. Scalable Bandwidth Real-Time Management

STEERS develops an innovative scalable and dynamic monitoring of the total available bandwidth of the home gateways tracked by the Media Distribution Graph. A media distribution graph segment is created for the distribution of each media object to one or more interested users: the MDG segment represents the 'offered bandwidth', i.e. the total bandwidth that involved home gateways are able to contribute, and the 'requested bandwidth', i.e. the playback rate of the delivered media object multiplied by the number of interested users. The user arrivals and departures, the changes in underlying network conditions and possible variations of the playback rate create dynamically changing differences between the offered and requested bandwidth. The *Bandwidth Management* function will monitor this difference accurately and in real time, in order to allocate resources only when needed and make the media distribution stable over time.

This component will be also able to lead the allocation of bandwidth from media servers to ensure low cost and stability at the same time. Additionally, real time transcoding functionality of the home gateways will be also exploited towards a stable media distribution, in cases where the system experiences a lack of bandwidth resources.

4.4.6. Social Aware Caching

Distributing media objects in the context of social networking activities is challenging, especially during the time of flash crowd and resource constraint in delivery networks. STEER addresses the aforementioned challenges by introducing a social-aware caching function, which integrates contextual social information, media object caching and proactive content placement features (*Social Context Integration*, *Social Aware Caching*, *Caching Management*). The goal of the social-aware caching function is to improve the efficiency of network resource utilization and to improve user experience in media streaming. We look into new caching strategies to reduce the amount of duplicated traffics in network by exploiting distributed caching facilities that are close to the end user. The novelty of such caching strategies lies in the integration of social context to predict content consumption within a social community in the near future (*Content Popularity Model*). By storing social media at the edge of network, we foresee improvement in some key quality metrics of media distribution such as start-up delay and picture quality, which directly contribute to the overall user experience.

The functionalities are facilitated by the presence of 'smart' home gateway. Home gateway resides in domestic household or commercial facilities such as a hotel, and provides a range of content management and processing features besides the conventional packet routing functions. As a storage-capable network node, a home gateway equipped with a Network Attached Storage (NAS) device (typically an external USB memory stick or USB / Ethernet hard-disk), allowing to increment the number of geographically distributed storage nodes. Through a cluster of APIs, functional modules in STEER architecture are capable of cooperating and commanding internal features at the home gateway, such as the above mentioned persistent storage (for caching), bandwidth estimation (for adaptive streaming, see Section 4.4.2), and transcoding (for curb-side re-coding of caching content with respect to certain playback specifications, see Section 4.4.7).

Again, these feature can be well exploited, particularly by the STEER P2P-based content distribution system, to minimize the traffic introduced in the network, helping to achieve the stated goal of increasing the performance and stability of applications involved in real time media distribution and user audio-visual communication.

4.4.7. Media Streaming & Home Gateway

The STEER reference architecture includes a home gateway that incorporates the capability of performing real time, on the fly media transcoding (*Transcoding* element), i.e. changing the encoding format and/or bitrate and resolution of digital audio and video component flows while the stream is transferred from a content source (e.g., a storage device, or a live feed facility) to a content sink (a display, or target storage device).

This functionality is gaining appeal in application environments that involve the exchange of real-time contents across both wide area and local networks, such as in-home content sharing, stream broadcast or multicast, and peer-to-peer communications. In these contexts, transcoding can improve the user experience and facilitate the introduction of new types of usage paradigms, thanks to its ability of dynamically optimizing the bandwidth utilization and allowing end user devices with limited capabilities (computation and rendering speed and/or display resolution) to access and reproduce high-quality contents efficiently and with a controlled quality loss.

For a series of technical and marketing reasons that have been presented in Deliverable D2.1, the latest generations of high-end home gateway seem well suited to host and efficiently and effectively use a transcoding functionality.

Home gateway-based transcoding can easily become a basic facility in most of the STEER use cases that concern the exchange of multimedia flows, which will be promoted by the social network and peer-to-peer infrastructures enhanced usage patterns that the project will explore.

Transcoding will as well nicely integrate with most of the other STEER media distribution components, namely Social Aware Caching, Scalable Bandwidth Real-Time Management, P2P Congestion Control, and Bandwidth Monitoring. The latter function, in particular, can be conveniently exploited by STEER applications for determining the best settings for transcoding operations, considering that the two functions can be co-located in the home gateway.

5. Major Functionalities and Relevant Interactions

This section presents the architectural components outlined in the previous section from the viewpoint of the detailed internal functionalities to be designed and implemented along with the mutual interactions with other functional components. Further details of each of the functional components will be provided in the corresponding deliverables of workpackages WP3 & WP4.

5.1. Dynamic Community Management

The STEER system connects to existing social networks and taps in to communities and media objects on them. The User, Event and Community Managers support this functionality, forming a bridge between the STEER system and other social networks. Figure 5 depicts these components.

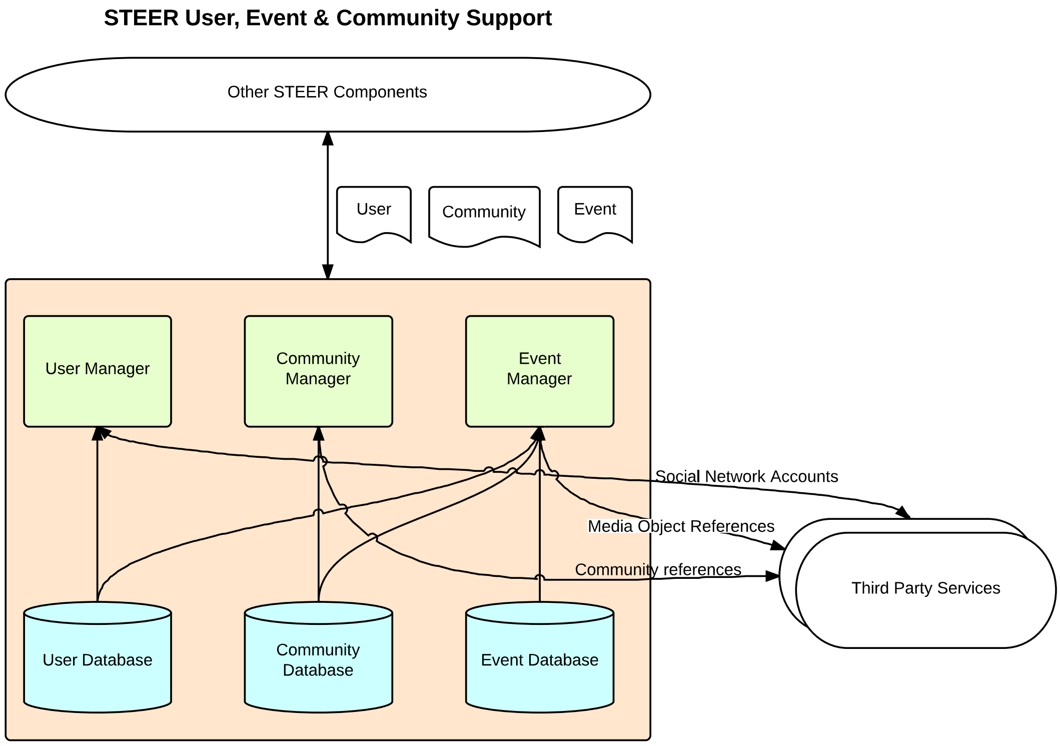

Figure 5. STEER User, Event and community support components and their internal cooperation

A relational database is used to store records for users, event, communities and the references to their counterparts in the external services. Furthermore, the event database stores references to related media objects in these communities.

5.1.1. User Manager

Objective

The User Manager maintains a database of user accounts. It stores user account information as well as account credentials for other social networks such as Facebook and Twitter. Finally, it stores social relationships within the STEER system. Other STEER components can query the user manager to obtain user credentials and the user accounts for third party social networks.

Functionalities

- Establish a connection between a STEER user account and the accounts of that same user in various other social networks, typically Twitter and Facebook, interfacing to them through their external API's.
- Support a single-sign-on procedure with these social networks to verify the credentials of a user.

Inputs from other STEER components

This component provides services to other STEER components, but does not strictly depend on any of them

Outputs to other STEER components:

The frontend user applications exploit the User Manager to verify user login credentials. The User Manager can provide user details and connections to social networks needed to other STEER components, required for instance to perform data mining or analysis.

5.1.2. Event Manager

Objective

The STEER Event Manager maintains a table of known 'events'. Events are created by an end-user when there is no existing event to which he can contribute.

Functionalities

- Event creation
- Provide event information to other components
- Maintain connections between events, users, communities and media objects

Inputs from other STEER components

This component interfaces with the STEER user applications, and receives raw metadata about media objects from the Content Ingest component.

Outputs to other STEER components

The STEER user applications use the Event Manager to keep track of media objects belonging to events, and to enable cooperation between different users attending or interested in the same event. The STEER Social Media Analysis components use information from the Event Manager as input to the Adaptive Event Profiler.

5.1.3. Community Manager

Objective

The Community Manager maintains the relationships between end-users and events, exporting this information to other STEER components and user applications. Furthermore, it recommends suitable existing communities and related event to the STEER user applications. In general the Community Manager acts as a gateway between STEER components, such as users, media objects, events and communities, and their counterparts in other social networks, supporting e.g. cross-posting of notifications and messages.

Functionalities

- Maintain and provide community information and relations to end-users. Maintain relationships with objects on other social networks (e.g., linking a STEER community to a Facebook group)
- Recommend suitable community-specific parameters such as time, locations, and current user interests

Inputs from other STEER components

The Community Manager gets input from the User Manager as well as the Event Manager. The User Manager provides user data and social relations. The Event Manager provides event data.

Outputs to other STEER components

The STEER user frontend applications query the Community Manager for information about a specific community and how it is connected to an event, media objects and user.

5.2. Social Media Analysis

The objectives of the Social Media Analysis functional components within the STEER architecture are as follows:

- To determine the popularity of the social media textual content and use it to rank media objects
- To mine social media entities, such as users, messages, and media objects, all with attributes from the social media textual content and relational information
- To enable implicit derivation of the Social Graph through message and user analysis
- To determine the relevance of entities and media objects based on the number of times they are linked or re-tweeted, as well as authority of, and distance to, the concerning users
- To enable early detection of potentially popular entities or objects.

The above objectives are mainly focused on the Live Augmented Broadcast application, through which STEER aims at investigating whether the user experience can be improved by delaying the video, so that it can better match the shown social media content.

Functionalities

In order to achieve the aforementioned objectives, the project will focus in the development and the experimentation of new components in several parts of its architecture.

The *Adaptive Event Profiler* generates and incrementally adapts the main event query. The *Social Media Scraper* retrieves/filters event-related information from social networks based on relevant user-id's and keywords. The *Social Graph* database contains interlinked messages, users and media objects and all their relevant attributes, within a certain relevant time window. The *Message and User Analysis* function performs entity extraction, sentiment detection, and authority measurement. The *Popularity Prediction* performs early estimation of popularity potential. Finally, the *Media Object Ranker* ranks media objects based on relevance and sentiment.

The overall architecture of the social media analytics stratum is depicted in Figure 6 below:

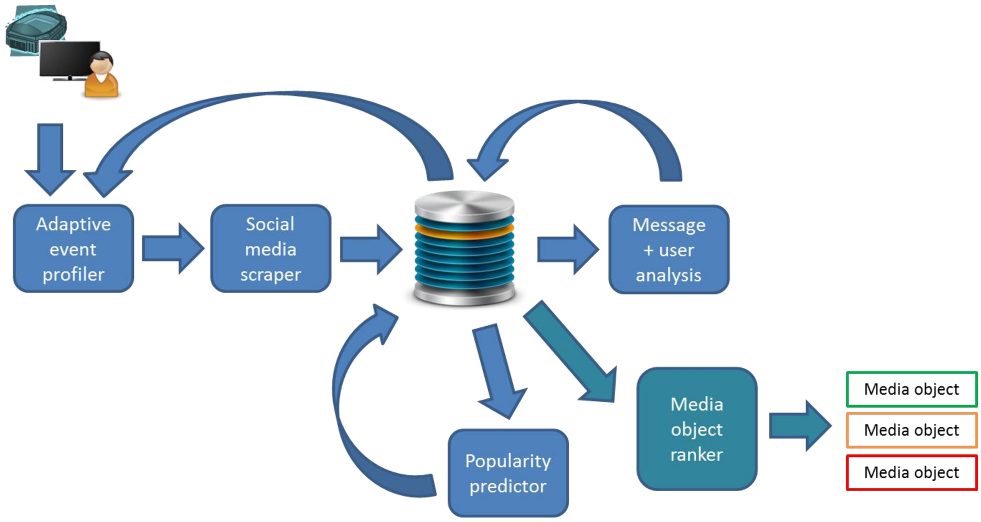

Figure 6. Functional modules in the STEER Social Analytics Stratum.

A more elaborate description of each functional module follows below.

Adaptive Event Profiler

The *Adaptive Event Profiler* is responsible for generating and continuously updating a search profile/query for an event. Initially this profile is based on the available event broadcast meta-information. The profile will be dynamically adapted based on information the *Event Profiler* retrieves from external sources (news, Wikipedia) and on information from the social media which is scraped for the event. Figure 7 below provides a more elaborate view on the *Adaptive Event Profiler*.

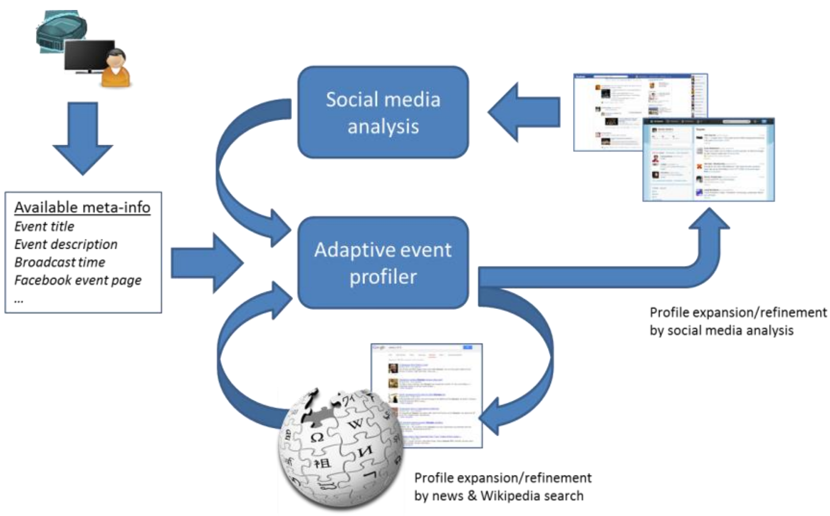

Figure 7. A more elaborate view on the Adaptive Event Profiler.

Social Media Scraper

Every time an updated profile is sent to the *Social Media Scraper*, it will refresh/adapt its API components. The scraper produces: users, messages, and media objects, all with attributes (network distance for users, timestamp, links, match relevance for messages, etc.). Relevant tracking keywords will be extracted from the event profile to filter tweets in real-time by using the Twitter Streaming API. By default, all activity of relevant users will be monitored as well. The selected tweets will be used for incremental profile adaptation, e.g. by extracting new event-related hashtags. The search API can be used to get more relevant users and messages from the recent history. The Facebook page API can be used to monitor activity on dedicated event pages. The YouTube Search API can be used to find relevant media objects (i.e. videos) based on keywords from the event profile. YouTube videos will also be retrieved indirectly through the links found in tweets.

The scraper outputs the messages, users, and media objects to the central database. This database is continuously updated with output of the *Message/User Analysis* module. For scalability and accuracy issues, we will use a decaying buffer of data. The database also contains the links between messages, users, and objects. Figure 8 below provides a more elaborate view of the *Social Media Scraper*.

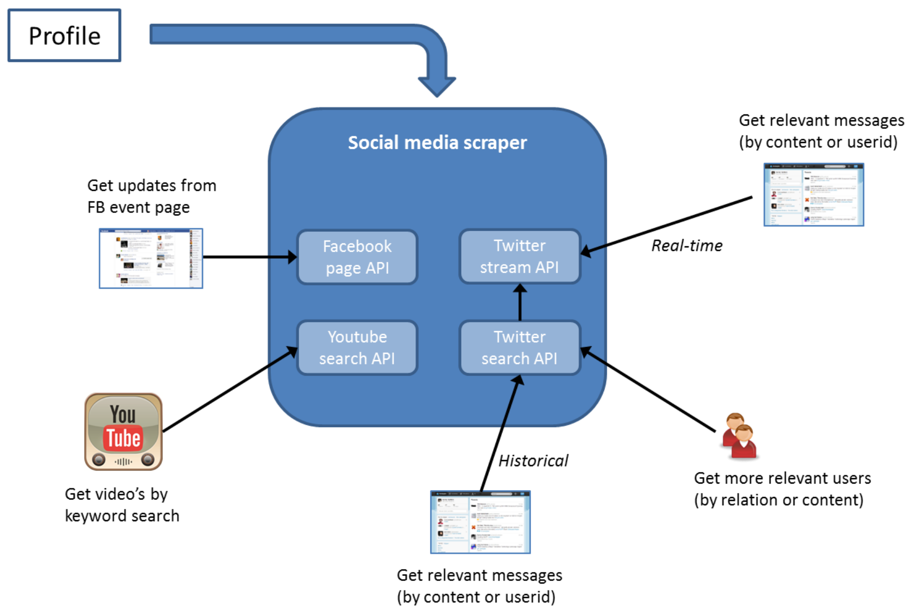

Figure 8. A more elaborate view on the Social Media Scraper.

Message/User Analysis

The *Message/User Analysis* block processes individual text messages, extracts the entities mentioned in the text and their expressed sentiments. The users will be analysed for authority by looking at their number of followers, history etc. Authority is added to the database as an attribute for users. Entities and their sentiments will be stored in the database. Links to media objects are resolved and added or updated with sentiments in the database. Figure 9 below provides a more elaborate view on the *Message/User Analysis*.

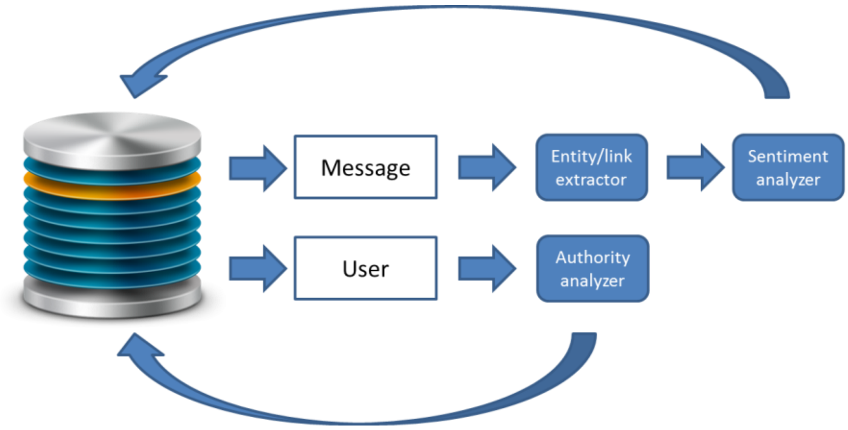

Figure 9. A more elaborate view on the Message/User Analysis.

Media Object and Entity Aggregation

There are two components that aggregate over the data (i.e. entities and media objects):

The *Media Object Ranker* ranks media objects based on sentiments, the number of times they are linked, their relevance weights, etc. The *Media Object Ranker* is personalized, i.e. content from people in your network is more relevant. Entities can be used to group media objects, e.g. all media objects related to a specific professional snowboarder, or snowboarding in general. An entity can also be a sub event (e.g. a certain match, or gig).

The *Popularity Predictor* predicts the popularity of entities and media objects by looking at early sentiments, number of times they are linked, authority of users involved, etc.

Figure 10 below provides a more elaborate view on the media object and entity aggregation.

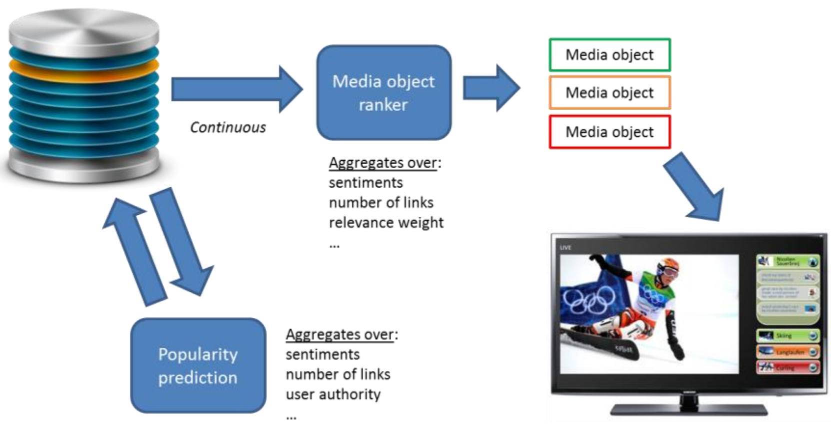

Figure 10. A more elaborate view on the media object and entity aggregation.

Inputs from other STEER components

The *Social Media Analysis* group of component interfaces with the STEER *Event Manager*, via the *Adaptive Event Profiler*. Initial information about events provides a basis for the event search profile/query.

Outputs to other STEER components

Social Media Analysis provides a *Social Graph* to be used by other components in the STEER system. Results from the media object ranker and popularity predictor are used in the Augmented Live Broadcast user application.

5.3. Media Service, Statistics and Media Object Graph

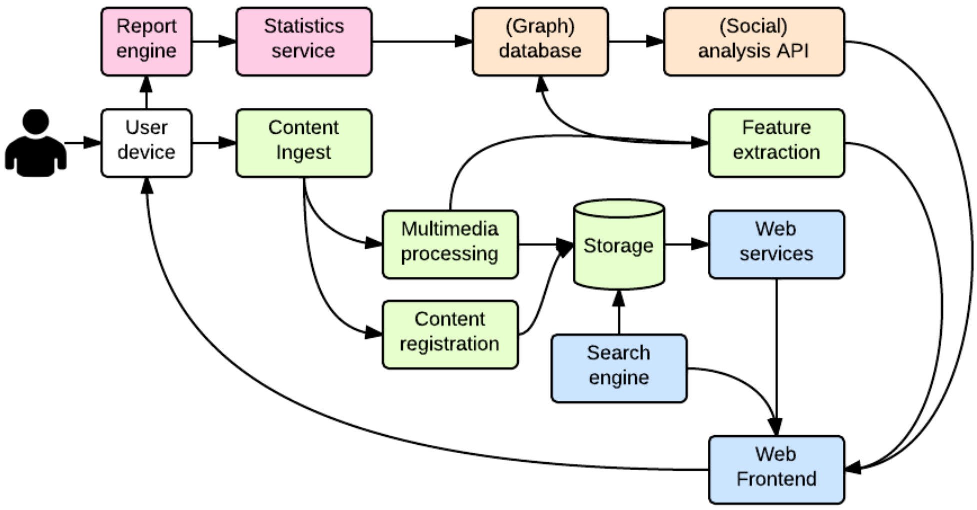

Figure 11. Media Service, Statistics and Media Object architecture.

5.3.1. Media Object Management

The main objective of the media objective management is to offer a reserved space where media objects can be uploaded, processed to a standard format, identified using a common identification schema, and retrieved upon user request in the context of social activities. Media object management is comprised of the following key service functions:

- **Content and Metadata Ingest service**

  The service component offers an API for user devices to upload audio-visual objects as well as any companion metadata. As part of the process, interoperation with user management function should also be realised to authenticate uploading activities and to maintain authorship and copyrights information whenever applicable.

- **Multimedia Processing**

  User uploads can be in various formats in terms of audio-visual formats and codecs. To guarantee the media playback of media objects from different sources and improve the efficiency in content delivery, this function conducts transcoding, chunking or re-encapsulation using a set of pre-defined service configurations.

- **Content Registration and Storage**

  A global identification schema needs to be designed to register and retrieve media objects in the STEER system. Related content, e.g., media objects of the same content in different quality levels, should also be recognized. Regarding the storage, (cloud) data storage should be available to host media content, and database storage should be in place to maintain metadata associated with any media object.

- **Feature Extraction**

  Metadata submitted by end users during content ingest might not be valid or sufficient for social analysis functions. Therefore, feature extraction function will be employed to derive additional metadata to represent aspects or characteristics of given media content.

5.3.2. Media Services

Media services aim at providing functionalities for content feed, retrieval and delivery.

- **Web Services and Frontend**

  The web frontend subscribes content and metadata feed from backend web services to allow users to explore social and media content. Modules must be designed to facilitate user interactions with items manifested on the frontend to conduct collaborative social activities such as story making.

- **Indexing and Search Engine**

  A search engine should index selected user and service data and customizable service API needs to be implemented to allow social information assisted searching and filtering, as well as location-based service to enable STEER test scenarios.

- **Media Distribution**

  STEER adopts different media distribution mechanisms for individual use cases. When a media object or a composite of media objects is requested, dedicated media servers or P2P distribution infrastructures are responsible to stream corresponding content to user devices. Therefore, dedicated chunking and encapsulation mechanisms should be in place and streaming servers such as a P2P seeding server or HTTP streaming server should be implemented.

5.3.3. Service Statistics

The statistics service records and manages system and user statistics in order to provide insights of social interactions. Researches then can be carried out to study how these social interactions may be exploited to improve content services and media distribution.

- **Report Engine**

  Statistics related to user social activities are ideally collected at a point that is close to end users. Reporting mechanisms are to be implemented at user client to monitor and report user statistics including user requests and ratings on media content.

- **Statistics Service**

  A robust and efficient statistics service (possibly comprised by a cluster of statistics servers) should be set up to become an entry point for all statistics reported by user devices and other relevant service elements. Statistics should then be processed, aggregated and maintained in a database system that is suitable for social network analysis.

5.3.4. Media Object Graph

Information regarding media assets and social activities in media services storied in statistics database can be formulated as a Media Object Graph. The relationships incorporated by MOG should be multifaceted and cover various aspects of a media service.

- **Graph Storage**

  Relational database and management system may not be suitable for the flexible schema of social objects and complex relationships between database entities. Therefore, advanced database models such as Google's Big Table, Facebook's Cassandra (Apache) or graph storage must be evaluated to best support social analysis in dynamic community.

- **Social Analysis API**

  Through the social analysis API, a number of generic graph analysis functions such as graph traverse should be made available to related media service functions, other social graphs in STEER system or experimentation modules.

5.4. Real-Time Social aware Media Distribution

5.4.1. Synchronization

<u>Objectives</u>

The objectives of the *Synchronization* functional component within the STEER architecture are as follows:

- To provide a synchronized view of a live event with a dynamic ranking of relevant media objects
- To synchronize social media content with the main event video for an improved user experience
- To provide synchronization of media at story viewers as directed by a storyteller.

Functionalities

In order to achieve the aforementioned objectives, the project will focus in the development and the experimentation of new components in several parts of its architecture.

Timeline insertion and clock synchronization modules in the STEER content ingest platform. The timeline insertion module inserts DVB timelines into broadcast content whereas the clock synchronization module facilitates clock synchronization between input sources. For mobile ingest devices that are part of the STEER content ingest platform, the clock synchronization module can be envisioned as a local application library. Additionally, such devices should accurately timestamp the captured video.

Synchronization and NTP servers in the Home gateway. The synchronization server performs orchestration amongst client devices whereas the NTP server facilitates clock synchronization between clients.

NTP clients in the playout device. The NTP client functions as a clock synchronization entity.

Additionally, an existing framework for timed interactivity will be expanded to address new synchronisation requirements in storytelling scenarios:

- Multiple destinations
- Multiple AV objects from different data sources
- Specifications and configurations of user devices (e.g., buffer)
- Story flow directed by a live storyteller (seek, pause, resume, fast forward, etc.)

Inputs from and output to other STEER components

In order to perform the synchronization functionalities, the synchronization modules will communicate with, and acquire information from, several other components within the STEER architecture. For the Live Augmented Broadcast application, the overall interfacing is depicted in **Error! Reference source not found.** below.

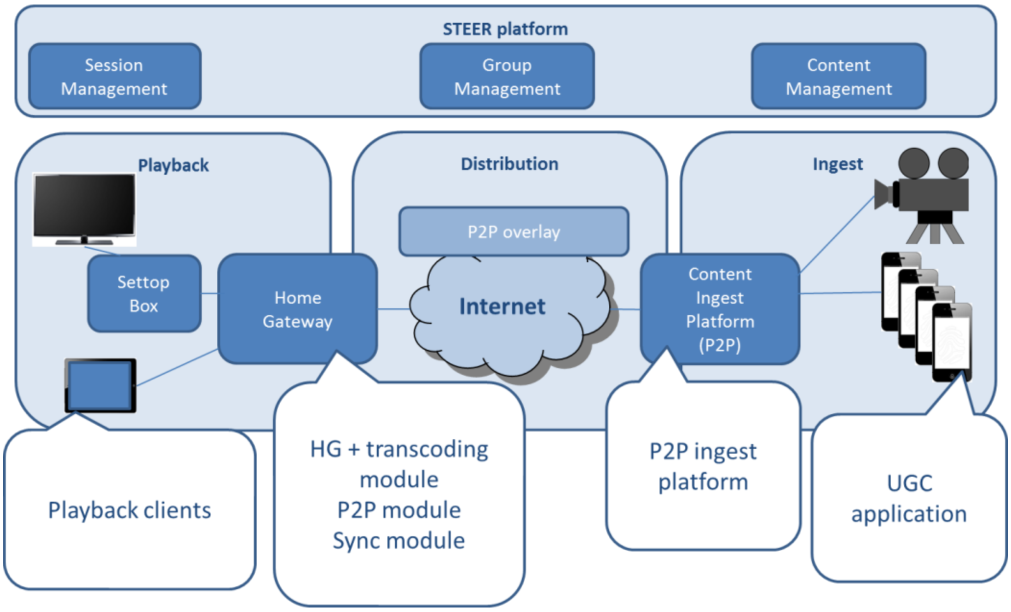

Figure 12. Live Augmented Broadcast application - Interfacing between synchronization components and other components within the STEER architecture.

Below we elaborate on the involved three parts of the STEER system.

Ingest

At the ingest point, the synchronization modules are co-located and interact with the content segmentation modules, in order to insert timeline information into the relevant media streams (see Figure 13).

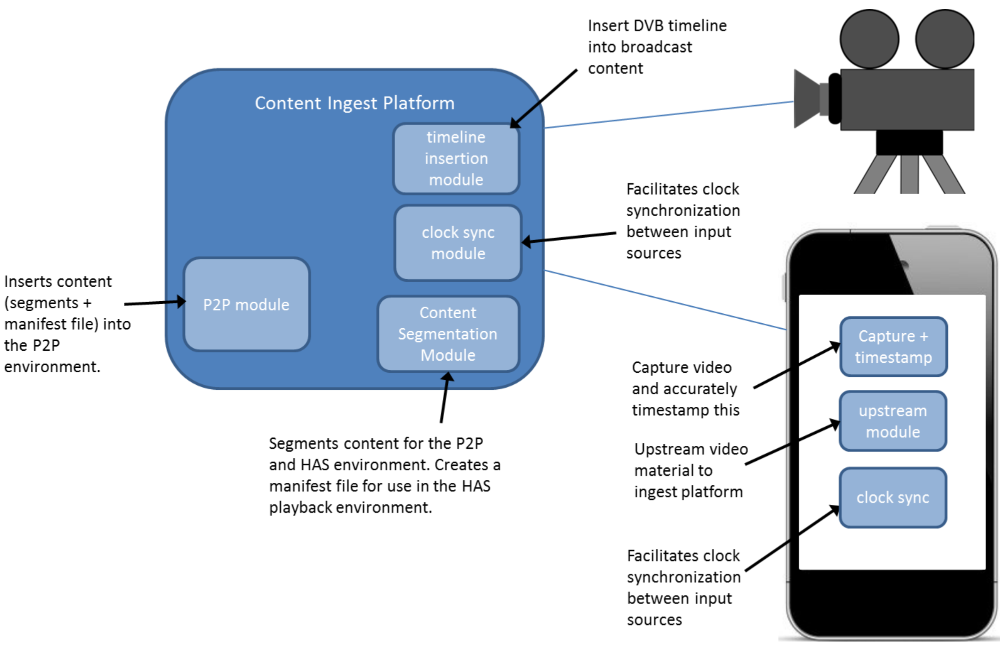

Figure 13. Interfacing between synchronization components and other components within the STEER architecture at ingest point

Home Gateway

In the home gateway, the synchronization modules are co-located with other home gateway functional modules, but mainly interact with client devices (see Figure 14).

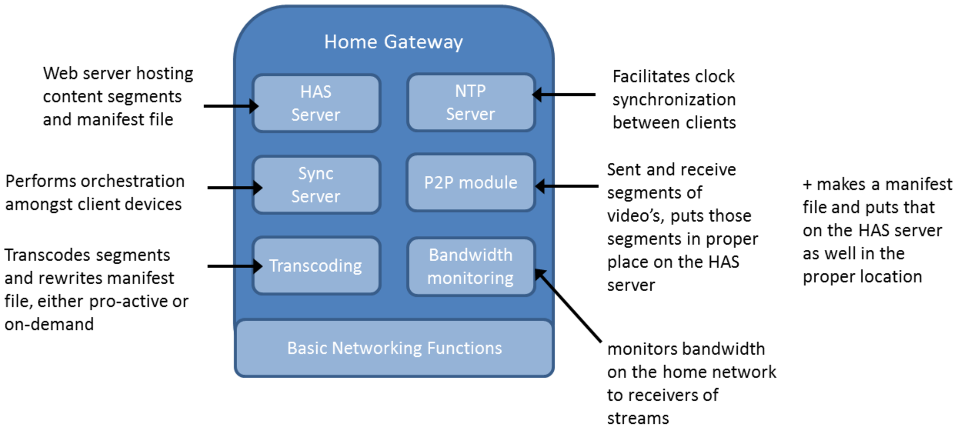

Figure 14. Interfacing between synchronization components and other components within the STEER architecture on the Home gateway.

Playout

On client devices, such as TV sets, tablets and smartphones, the synchronization modules are co-located with and interact with video rendering and social media messaging modules (see Figure 15).

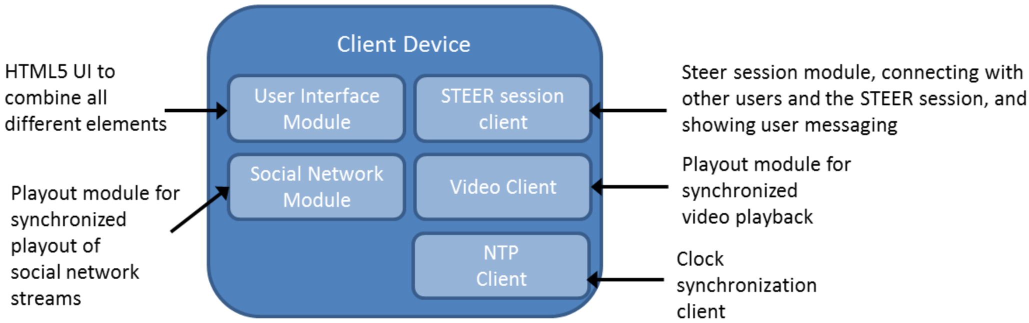

Figure 15. Interfacing between synchronization components and other components within the STEER architecture on the client device.

In case of the Storytelling application, the interfacing is depicted in Figure 16 below. The storyteller composes a story using audio-visual content stored in a form of 'cloud' storage. A story manifest is derived so that a story, which is complex composition of audio, video and image chunks, can be defined using a text-based description and stored in story server. The manifest is then used by application at story viewers to determine the timing information of the corresponding audio-visual fragments in multiple media tracks within a story, so that the user experience is consistent despite the sources of media chunks at the time of playback. In the case of 'live storytelling' where storyteller actively orchestrates the playback position for all story viewers, synchronization and control signals are also exchanged.

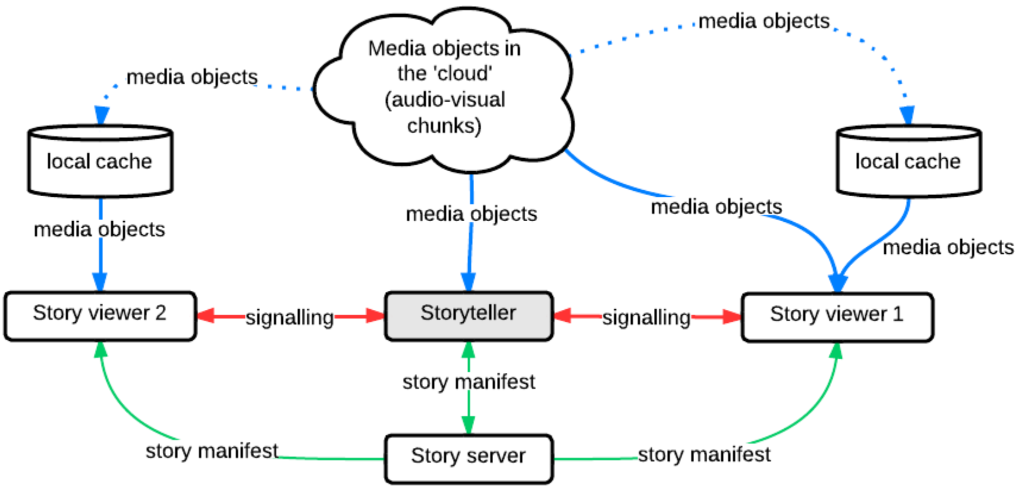

Figure 16. Storytelling application - Interfacing between synchronization components and other components within the STEER architecture on the client device.

The storyteller in Figure 16 does not have to be a PC operated by a user. When a story is published as an on-demand composite content, a media server can essentially host the functionalities of the storyteller and manage playback sessions of all story viewers.

5.4.2. Bandwidth Monitoring

Objective

The major objective of the *Bandwidth Monitoring* component is to probe the available bandwidth and capacity of point-to-point links established between the home gateway and the end devices in a home network.

Bandwidth Monitoring can (at least) support the following popular transmission techniques:

- Ethernet via UTP cables
- Wi-Fi
- Powerline communication (PLC)

The *Bandwidth Monitoring* component will be able to estimate the available bandwidth and capacity with a certain accuracy on all three network types.

Functionalities

The basic functionalities of the *Bandwidth Monitoring* component are already developed and tested in a simulated environment. Within the STEER project the goal is to integrate the component into a home gateway for experimentations purposes and offering the necessary data to the other STEER components.

Inputs from other STEER components

As *Bandwidth Monitoring* is in a low layer, there are no inputs from other components of STEER.

Outputs to other STEER components

Bandwidth Monitoring generates information about the connection between the home gateway and the end devices. The home gateway, or any other client application that can communicate with the home gateway in a suitable way, can use this information for decisions on streaming priorities and or optimal transcoding. This information also provides applications with the possibilities ofsuggesting the user to redirect traffic to other devices with more bandwidth, e.g. watching the video content on the TV instead of the tablet.

5.4.3. Media Distribution Graph

Objectives

As we clarified earlier, the Media Distribution Graph (MDG) dynamically determines, according to the requirements and constraints that applications introduce, the set of users (or home gateways) and media servers that facilitate each user towards the reception of the video stream. This way it also determines the set of users which each media server or home gateway feed with video blocks. The requirements for a media distribution graph are:

- To exploit in a high degree the bandwidth resources of socially correlated users
- To be able to deliver each video block to every participating user with a low latency
- To maximize the utilization of a set of heterogeneous home gateways in terms of upload bandwidth
- To be stable to user arrivals and departures
- To be friendly to the underlying network (locality aware)
- To be scalable in terms of the management overhead that it introduces
- To adapt to the dynamic traffic of the underlying network

Functionalities

In order to achieve the aforementioned objectives, the project will focus in the development and the experimentation of three components.

The graph architecture. As graph architecture we define the creation of a set of rules that will enforce our objectives. Indicatively, these rules will be related to: the number of nodes to which each node sends video blocks, the number of nodes from which each node receives video blocks, the set of nodes with which each node communicates. All these will be determined according to the network location, the upload bandwidth and the social relationships of each participating node.

A distributed node discovery algorithm ('gossip protocol'). Thanks to this element each node will be able, dynamically and in a scalable fashion, to discover other nodes with which it will potentially exchange video blocks.

A distributed optimization algorithm. The requirements that we have from our MDG will be quantified and put in terms of a distributed optimization problem. In order to solve it, each node will periodically execute a distributed optimization algorithm in order to achieve a global optimum.

Inputs from other STEER components

In order to perform the functionalities that MDG requires, this element will communicate and acquire information from two other components, namely:

- the *P2P Congestion Control* block, that will dynamically inform the MDG about the upload capabilities of each node and the network conditions (i.e. latency, jitter) between two nodes in different network locations;
- the *Social Graph*, which will propose to the MDG nodes that have high social correlation with those requesting the media object that MDG distributes. The MDG will exploit the resources of such nodes towards a more effective media distribution.

Outputs to other STEER components

The MDG will be able to facilitate other components to implement a more stable and social aware media distribution system. In more detail, it will be able to inform the scalable bandwidth real-time management function (*Bandwidth Management*) about the surplus or the deficit of the upload bandwidth in each MDG segment. Furthermore, the MDG will be able to give detailed information to the *Media Object Ranker*. This information concerns the viewing behaviour of each user in terms of the time instant and the duration that consumed each media object.

5.4.4. P2P Congestion Control

Objectives

As briefly stated earlier, the major objectives of STEER's P2P congestion control algorithm are:

- To utilize in a high degree the upload bandwidth of home gateways and media servers
- To be stable and robust under time-varying delays and dynamic underlying network traffic
- To be able to measure accurately and dynamically network latencies, path conditions (i.e. error rate), upload bandwidth capacity and idle resources
- To be able to enforce, in terms of network resources that it allocates, the policy that applications require when they coexist with other 'traditional' TCP flows that pass through the same home gateway.
- To avoid overloading the buffers of the home gateways and the routers in the underlying network (BufferBloat problem).
- To avoid unnecessary high latencies and retransmissions

Functionalities

In order to achieve the aforementioned objectives, we will focus in the development and the experimentation of four major functionalities:

- Dynamic calculation of the number of packets that will remain unacknowledged (due to sliding window size), according to the dynamic features of the underlying network paths (throughput, latency, jitter). An accurate estimate will be able to guarantee high network utilization and congestion avoidance.
- Control strategy for dynamically determining the rate with which each node infuses packets into the underlying network. The role of this function is to enforce stability of congestion control, which translates as QoS in the application layer.
- Dynamic monitoring of the rate with which packets are acknowledged, and of the network latency of paths to which each node sends video blocks. This is a very important functionality as it gives feedback to the control function and affects directly the effectiveness of P2P congestion control.
- Development and experimentation of a retransmission mechanism that factorizes the multiple network locations in which each node sends, and fulfil the real time constraints that live streaming sets.

Inputs from other STEER components

As P2P congestion control is a low layer functionality that needs no input from other STEER components.

Outputs to other STEER components

P2P congestion control will calculate accurately and dynamically the upload bandwidth of each node and the network latency between two nodes. This information will be sent to the Media Distribution Graph in order to adapt media distribution to the underlying network features. Additionally, P2P congestion control will inform the scalable bandwidth real-time management function about the amount of the idle upload bandwidth. In this way, it will give the opportunity for controlling the quality of streaming by allocating bandwidth from media servers or transcoding of the fly the video streams.

5.4.5. Scalable Bandwidth Real-Time Management

Objectives

Scalable Bandwidth Real-Time Management (*Bandwidth Management*) is the component that ensures the sufficiency of bandwidth resources to each MDG segment. The objectives that we have for such a component are:

- To be able to monitor with an accurate, stable and scalable manner the total available upload bandwidth in each MDG segment
- To be able to calculate the amount of bandwidth that is necessary for complete and stable media distribution

- To be able to allocate in a dynamic and stable fashion, in case this is needed, an auxiliary amount of bandwidth from media servers when participating nodes are not able to support the live distribution of the requested streams.

Functionalities

In more detail, *Bandwidth Management* will be implemented as a distributed and dynamic architecture for the monitoring and control of the bandwidth resources in each MDG segments, facilitating a stable live distribution of each media object**.** This will take place again among the home gateways that belong to each specific MDG segment. As noted in Section 4.4.5, a 'offered bandwidth' is defined as the total available upload bandwidth that nodes of an MDG segment can contribute considering the dynamic underlying network conditions. A 'requested bandwidth' is defined as the dynamic playback rate of each media object multiplied by the dynamic number of users within an MDG segment (community). The user arrivals and departures, the changes in underlying network conditions and possible variations of the playback rate make the difference between the offered object bandwidth and the requested object bandwidth to change significantly in time.

The *Bandwidth Management* function will monitor this difference accurately and in real time. In case it is positive (indicating a surplus of offered bandwidth), the function will estimate the exact amount of idle resources. In case the difference is negative (indicating a deficit of offered bandwidth), the function will calculate dynamically the exact amount of missing bandwidth that media servers have to contribute. Our research objective here is the creation of decoupled large scale state space models that would allow to exploit modern control theory in order to develop a distributed (and therefore scalable) and stable bandwidth monitoring system, which is robust against several uncertainties that users behaviour and network behaviour introduces. These features will enhance the QoS of media object distribution within for each MDG segment.

Inputs from other STEER components

As we described earlier, the *Bandwidth Manager* takes input from MDG and *P2P Congestion Control* elements to monitor in real time the availability of the total upload bandwidth in each MDG segment.

Outputs to other STEER components

The *Bandwidth Manager* will be able to allocate dynamically bandwidth from media servers and give input to the real time transcoding mechanism that will adapt video streaming quality in the home gateways.

5.4.6. Social-Aware Content Placement and Storage Allocation

This group of functional elements, represented in Figure 17, constitutes a set of functions used to establish a content delivery scheme that reduces the impact of a flash crowd in service and resource constraint in distribution networks by using intelligence gathered from social networks and service statistics. More specifically, the work in this area will investigate how social information in a dynamic community can be integrated in media services and distribution networks. The integrated social information is then adopted as a metric in managing the life-span of media content and improving the efficiency of media distribution through advanced caching strategies in distribution networks.

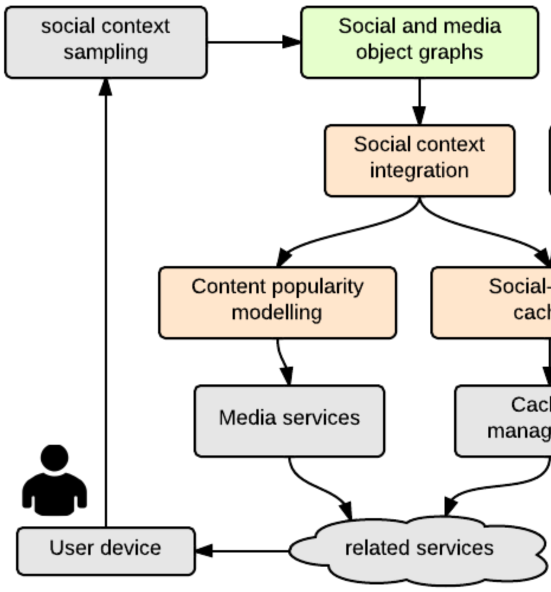

Figure 17. Social aware content placement and storage allocation.

The main functionalities include:

- **Social context integration**

  This function integrates social information using given social analysis APIs provided by relevant Social and Media Object graphs and models social activities of a community.
- **Content popularity modelling**

  Based on system statistics, the function estimates future popularity of media objects using input from social content integration tasks. The function then provides guidance for media services to dynamically adjust the composition of a service catalogue.
- **Social-aware content caching**

  This is a cross-layer function that directs caching management in distribution network to proactively apply social information-assisted caching strategy. The function also appreciates network-level metrics from the media distribution graph.

5.4.7. Media Streaming & Home Gateway

Objective

The main usage envisaged for the Home gateway in STEER is:

- As a real-time transcoding unit for streaming media exchanged on top of the STEER Media Distribution architecture
- As a platform hosting, or working in conjunction with, some STEER components, namely the *Synchronization* modules (Section 5.4.1, Figure 14),

the *Bandwidth Monitoring* module (Section 5.4.2), the P2P functionality (Section 5.4.4), and storage allocation (5.4.6).

To derive indications that can be directly applied in view of short term exploitation, a commercial, albeit advanced, device will be considered for integration and experiments.

Functionalities

The basic architecture for transcoding in the home gateway is shown below in Figure 18.

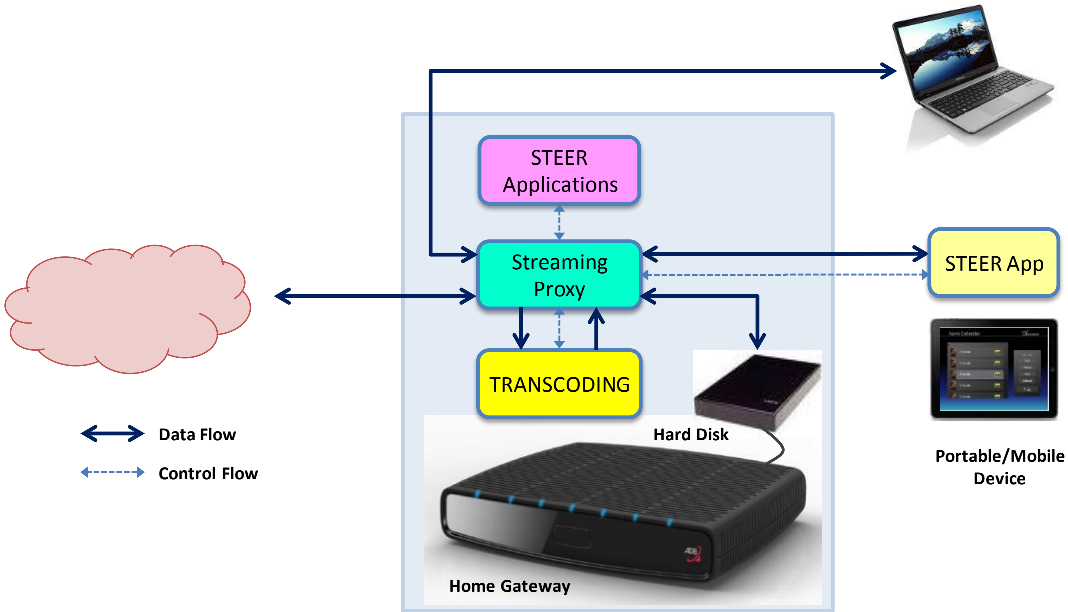

Figure 18. STEER Home gateway reference model.

This architecture can support a number of use cases that mainly differs from the nature and location of streaming sources and sinks, including

- Direct content (chunk) download to, or caching on, the home gateway NAS
- Direct content (chunk) upload or share from the home gateway NAS
- Content (chunk) download to, or cache on, home network or user device storage space
- Content (chunk) upload or share from home network or user device storage space

all of which involving on the fly media transcoding.

The transcoding unit hosted by the home gateway can be seen as a configurable black-box that takes an encoded media stream as input and produces as output the same stream re-encoded according to a specific set of parameters. The access to this function is mediated by an element, the 'Streaming Proxy', which receives requests of retrieving or uploading a stream from / to a destination end-point indifferently located in the home network or in the cloud, by a STEER application installed on a user device (e.g., a smartphone or a tablet) connected to the home gateway. In principle, the user device can reside in the home or be connected to the home gateway from a remote location through an authenticated link. The STEER application on the user device determines the end-point location according to the activity it is carrying on, for example downloading media contents from a remote caching node to be used for composing a STEER story. The application also selects the stream encoding characteristics it desires, normally in terms of (average) bit-rate, frame rate and/or display resolution. This decision typically depends on the specific usage that the application wants to make of the media, and must be based on the knowledge of the original media format and the transcoding capabilities of the home gateway (either known in advance, or acquired through a preliminary interaction with the device).

Furthermore, other STEER applications executed by the home gateway can provide useful information, as in the case of the *Bandwidth Estimation* element, which may help determining that the streaming transfer should not preferably exceed a given bandwidth to avoid incurring in excessive frame errors, delay or jitter.

The end-point location and transcoding parameters are then communicated by the user application to the home gateway Streaming Proxy, which will take care of mediating the transfer from source to destination and perform transcoding in between them. Variations to this paradigm can be devised, such as determining transfer and transcoding 'coordinates' by an application running on the home gateway instead of on a user device, as may be the case of a cache controller managing (partitions of) the home gateway NAS.

The home gateway can as well integrate other applications that do not make use of transcoding, but rather export services that are exploited by other elements of the STEER architecture or the transcoding function itself (such as the already mentioned *Bandwidth Estimation* module). The integration of such applications is supported either through the services offered by an embedded Virtual Execution Environment based on a standardized Framework (OSGi), or a lower-level skeleton for software modularity which is part of the home gateway operating system. Applications will coexist with home gateway 'core functions' (such as traffic routing/bridging, firewall, user interaction, device management).

Inputs from other STEER components

Inputs from the transcoding function are generated by other STEER application modules that can provide information for setting transcoding parameters and/or initiate a transcoding operation from a specific source to a specific sink.

Outputs to other STEER components

The transcoding function is implemented mainly as a 'tool' for exploitation by other application modules in the STEER system, in particular those already mention above

6. Augmented Live Broadcast System Architecture

Deliverable D2.3 describes in detail the Augmented Live Broadcast (ALB) application with which users are able to ingest content that is synchronized and played in parallel with professional coverage of events. In this section, by using Augmented Live Broadcast as an example, we describe how those functionalities are orchestrated towards a social aware media distribution, shown in Figure 19.

End user device – Content Capture & Ingest Application (CIA): This is the first of the two applications in ALB and its objective is to give the capability to both professional content providers and users to generate content and to stream it live (in real time). Content can be generated with professional cameras and/or end-user devices, enhanced with information useful for synchronization (i.e. timing and session aware), and forwarded directly to the Content Ingest Platform. The Content Ingest Platform manages media object properties such as encoding protocol, resolution and frame rate. Moreover, in this stage, user information such as user id, location and timing elements that are later used for synchronization are also managed.

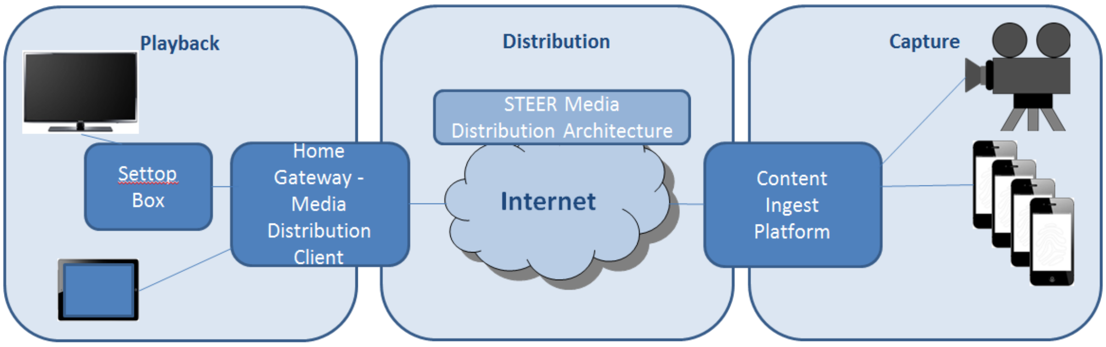

Figure 19. STEER Augmented Live Broadcast architecture

STEER Media Distribution Architecture: There are two types of media distribution architectures that the ALB can deliver, namely a traditional client -server media distribution architecture and a P2P-cloud media distribution architecture, aimed at scalability, low cost and stability. As mentioned before, in both cases the content is propagated from the content ingest platform.

In the P2P-cloud media distribution architecture, data chunks generated by the content ingest platform are propagated to a P2P streamer for further fragmenting into P2P blocks of data, and then infused in the Media Distribution Graph.

Additionally, the scalable bandwidth real-time management function monitors the available total upload bandwidth of the users that participate in the distribution of each media object and interacts with media encoding protocols in order to dynamically adjust playback rate to the available total upload bandwidth according to the underlying network conditions.

Home Gateway: In the home network there can be bandwidth limitations as well, caused for example by the use of shared media technologies, such as Wi-Fi or Powerline communications. The media stream as delivered by the Media

Distribution Architecture may be too bandwidth-intensive for a delivery over such links. To cope with this situation, the home gateway will be able to transcode media streams on-the-fly. As described in Section 5.4.7, the home gateway is equipped with a bandwidth monitoring function, which enables the measurement of the available bandwidth between the gateway and any end user device. When the home gateway detects insufficient- or strong fluctuating bandwidth, it transcodes the media stream to fit within the available bandwidth.

End user device - Content Consume Application: In case of P2P media distribution, a P2P client runs in a layer between the home gateway and the end user device and performs three major functionalities, two of which have been described earlier in detail and constitute major experimentation objectives of STEER. These are the Media Distribution Graph (P2P overlay) and P2P congestion control. Additionally, a distributed scheduler coordinates P2P block exchanges among participating peers. P2P client also provides input for scalable bandwidth real-time management.

P2P client interacts with the home gateway that performs bandwidth monitoring and transcoding in order to ensure stable media distribution.

7. Storytelling Architecture

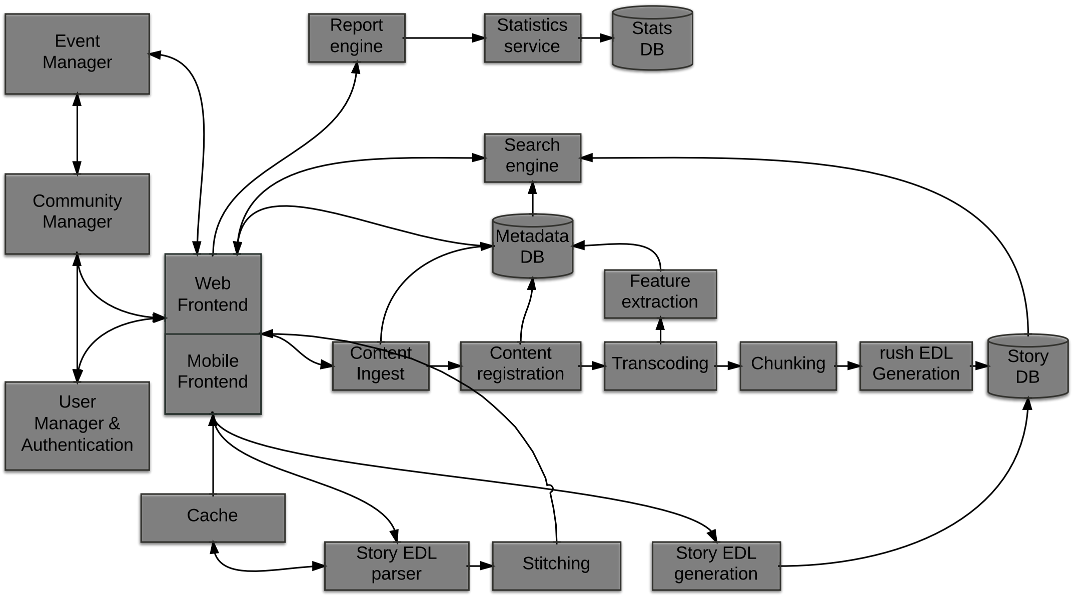

Figure 20. STEER Storytelling application architecture

Figure 20 shows the architecture of the STEER Storytelling application, the components of which have been already introduced in the previous sections. Here the interactions between these components are discussed more specifically in the light of the Storytelling application.

The Storytelling application consists of two frontends as part of the Media Services (4.3.2), namely a Web frontend and a mobile frontend. The Web frontend is used for creating, editing and managing stories. The mobile application, constrained by it display form factor, limits the functionality of media object ingest, browsing and consuming, and has limited editing functionality. Most functional components, typically computation intensive, are implemented in backend components. Some components are highlighted here to explain how the Storytelling application works.

The Web and Mobile Frontends connect with the *User Manager* to verify user-login credentials. The *Community Manager* is the gateway to third-party social networks, for posting messages and notifications. The *Event Manager* is used to provide relevant events, based on current location, time, the end-users social network, etc. Media objects recorded with the mobile application are ingested to the media object management service (4.3.1). Within the media object service, user uploaded videos are received by the *Content Ingest* module, after which they are processed by the *Content Registration* and *Multimedia Processing* blocks, perming transcoding, chunking and EDL generation (see below), and are finally stored in the Metadata and Story databases (which together compose the *Media Object Storage*). This chain of content processing is designed to facilitate content retrieval, management by the users of their own uploads, quality-independent video editing, and story sharing.

The STEER Web application uses a special XML format, coined 'EDL', to express and create the structure of a storyline. The EDL parser is used to 'Stitch' individual video fragments into an assembled video, which is suitable for playback. The 'Stitching' function interoperates with STEER caching infrastructures (see Section 4.4.6, social-aware content placement and storage allocation) so that fragments can be discovered and retrieved from a caching node near the end user. A synchronization engine (specified in Section 4.4.1) coordinates between story playback modules at frontend and corresponding media sessions. Service statistics are reported by user applications and subsequently filtered, processed and stored by the *Statistics Service* (Section 4.3.3) for the purpose of both service experimentation and social analysis.

8. Conclusions

STEER architecture may be seen as an example of a system of systems that aims at efficiently combining three complex instances of architectures, namely, social networks, media objects and media distribution. It approaches each one of them as rich sources of information and attempts to explore various ways of discovering and exploiting the correlations among them. We consider this a key research challenge that STEER aspires to shed some light through experimentation.

Another key research challenge that overarches these systems is how their functionalities may be combined architecturally and operationally in order to give rise to a novel architecture. STEER, for practical purposes, has already taken some architecture decisions, the result of two very innovative use cases. It is, however, the experimentation with these use cases and the STEER system behind them that will give us valuable feedback about the validity of some the designs and provide insights for new ones.

The two aforementioned challenges are complemented by hot research questions that emanate from within each one of these (sub) systems. These research questions will be tackled through extensive experiments that will be specified in WP5 when deploying and operating the STEER architecture.