



D7.4 Basic Specification of the Intelligent Media Framework Public Synopsis

Project Ref. No.	Integrated Project FP6 / IST 27312
Deliverable	D7.4 – Basic Specification of the Intelligent Media Framework (Public Synopsis)
Delivery Date	31/08/2007
Author(s)	Georg Güntner / SRFG / gguentner@salzburgresearch.at Tobias Bürger / SRFG / tbuerger@salzburgresearch.at Rupert Westenthaler / SRFG / rwesten@salzburgresearch.at Dietmar Glachs / SRFG / dglachs@salzburgresearch.at
QA	Wernher Behrendt / SRFG / wbehrendt@salzburgresearch.at
Filename	LIVE-7-D7.4-IMF-Specification-PublicSynopsis
Publication Level	PU (public)

Copyright Notice:	© LIVE Consortium. 2007. All rights reserved.	
Disclaimer:	<p>This document contains material, which is the copyright of certain LIVE consortium parties, and may not be reproduced or copied without permission. The commercial use of any information contained in this document may require a license from the proprietor of that information.</p>	
	<p>Neither the LIVE consortium as a whole, nor a certain party of the LIVE consortium warrant that the information contained in this document is capable of use, nor that use of the information is free from risk, and accepts no liability for loss or damage suffered by any person using this information.</p>	
	<p>Neither the European Commission, nor any person acting on behalf of the Commission, is responsible for any use which might be made of the information in this document.</p>	
	<p>The views expressed in this document are those of the authors and do not necessarily reflect the policies of the European Commission.</p>	
Project Information:	<p>Full project title: Project Co-ordinator: Project ID:</p>	<p>Live Staging of Media Events Joachim Koehler / FhG FP6-27312</p>

Table of Contents

Executive Summary	4
1 Introduction	5
1.1 Architectural Overview	5
1.1.1 Building Blocks of the LIVE Production Support System.....	5
1.1.2 Architecture of the Intelligent Media Framework.....	6
1.2 Structure and Scope of the Report.....	9
1.2.1 Structure	9
1.2.2 Scope	9
1.2.3 Relationship with other Deliverables	9
2 The Intelligent Media Framework.....	11
2.1 Overview on the LIVE Staging Domain	11
2.2 Requirements of different agents in the LIVE staging process.....	13
2.2.1 The “Levels of Meaning” classification schema	13
2.2.2 Automatic annotators.....	14
2.2.3 Human annotators.....	15
2.2.4 Editor	15
2.2.5 Video Conductor	16
3 Requirements for the IMF.....	17
3.1 Knowledge Requirements	17
3.2 Framework requirements.....	18
4 A Knowledge Model for the IMF.....	19
4.1 Knowledge Structure of the IMF.....	19
4.2 Term Model.....	20
4.2.1 Application of SKOS.....	20
4.2.2 Term Type Hierarchy	21
4.3 Event Domain Model	23
4.4 Basic IMA Model.....	26
5 Architecture of the Intelligent Media Framework	29
5.1 System Design of the Intelligent Media Framework.....	29
5.1.1 Components of the LIVE Production Support System.....	29
5.1.2 Components of the Intelligent Media Framework.....	31
5.2 The First Prototype.....	34
6 Conclusions and Future Work	36
6.1 Assessment of IMF Requirements	36
6.1.1 Assessment according to Knowledge Requirements.....	36
6.1.2 Assessment according to the Framework Requirements.....	37
6.1.3 Conclusions	38
6.2 Future Work	39
7 References.....	41
Appendix: Index of Figures	42

Executive Summary

The integrated project “LIVE Staging of Media Events” (LIVE; FP6-27312¹) attempts to radically improve on the linear approach to TV broadcasting of live sporting events by delivering digital technologies and content formats that enable viewers to shape their own personal highly interactive viewing experience as they watch the broadcast. The LIVE solution is a production support system that hooks into a broadcaster’s production infrastructure thus enabling contextual search and recommendation of multimedia content in a real-time production environment across multiple sources and formats. It consists of an Intelligent Media Framework to formalise and manage the semantic connections across the system, semi-automatic annotation tools to index multiple incoming streams, information databases and AV archives, and a recommender system to analyse and visualise consumer feedback that is delivered over a back channel system.

This report presents the basic specification of the *Intelligent Media Framework* (IMF), one of the core subsystems acting as the middleware for the LIVE production support system. The framework provides the following services to other LIVE subsystems, such as the *Metadata Generation System* and the *Recommender System*:

- Services to create, annotate and manage the intelligent media assets that make up the show under real-time conditions. These services operate on a metadata level and do not actually store any raw video streams (they rather reference these so called “*essences*”).
- A service to manage and deliver information about the staged events, e.g. the schedule of the contests and races, the participating athletes, the results, etc.
- A service to manage and deliver information on the way how a live broadcast of a sporting event is presented, e.g. which types of switching concepts are available and used in a certain concept of a show, which basic dramaturgic concepts are appropriate according to the disposition of the production team and/or a predefined mood of the show, etc.
- A service to access the vocabularies and the terms of the controlled vocabulary constituting the knowledge base of the live staging domain.
- A messaging system to support the real-time aspects of the staging process by offering subscription methods to other subsystems.

The Intelligent Media Framework is based on a combination of a classical three-tier architecture with the principles of service oriented architectures (SOA). Moreover, the framework relies on a knowledge model for the LIVE staging domain that on a semantic level integrates information about the event, the content, the users involved in the production process and the concepts of presenting the event (“staging”).

The objective of this report is to provide a synopsis of the basic specification of the Intelligent Media Framework as developed during the first iteration cycle of the LIVE project (from January 2006 to June 2007). The implementation of this specification formed the middleware of the first prototype of the LIVE production support system.

This report is made available for and addressed to, the interested public. It presupposes some basic knowledge of software and knowledge engineering as well as some understanding of broadcasting issues. The list of available reports of the LIVE project can be found on the official home page of LIVE.

¹ LIVE staging of Media Events: <http://www.ist-live.org/> - Last visited: 01.07.2007

1 Introduction

The central idea of the “LIVE Staging of Media Events” (LIVE) project is to create

- Novel intelligent content production methods, and
- Tools for creating interactive digital broadcasts

to stage live media sporting events such as the 2008 Olympic Games. “Staging live media events” means the creation of a non-linear multi-stream video show in real-time, which changes in response to the interests of the consumer (end user). The LIVE solution for the technical requirements of “live staging process” is a production support system that hooks into a broadcaster’s production infrastructure where it enables contextual search and recommendations of multimedia content in a real-time production environment across multiple sources and formats. The system relies upon a middleware framework called *Intelligent Media Framework* (IMF) whose specification forms the central topic of this report.

The objective of this section is to

- Introduce the overall system architecture and the main building blocks of LIVE’s production support system as described in [LIVE D9.3];
- Introduce the architectural approach for the system design of the Intelligent Media Framework, its service layer and the knowledge model of the LIVE staging domain on which the services are based; and
- Provide an overview of the structure and scope of this report as well as its relationship with other LIVE deliverables.

1.1 Architectural Overview

1.1.1 Building Blocks of the LIVE Production Support System

The technology infrastructure of the LIVE production support system is based on a *Metadata Generation System*, an *Intelligent Media Framework* (at the level of the building blocks represented by the *Intelligent Media Asset Information System*), a *Recommender System*, a *Consumer System* and a *Video Conducting System* (see Figure 1):

- The **Metadata Generation System** is responsible for the detection, extraction and annotation of live content. It contains automatic annotation tools as well as the human annotation tool (HAT), the graphical user interface to be used by a human annotator.
- The **Intelligent Media Asset Information System (IM AIS)** is a component of the Intelligent Media Framework which is responsible for the management of the controlled vocabulary, event information and intelligent media assets. IM AIS also implements services to manage and query this data.
- The **Video Conducting System** covers all components necessary for the LIVE staging process. This includes the user interfaces to be used by the editor and video conductor as well as interfaces and adapters for the broadcasting environment.
- The **Recommender System** is responsible for the calculation of content recommendations for professional users as well as for consumers. In addition the recommender system collects and analyses feedback provided by the consumer.

- The **Consumer System** supports the functionality needed at the user's end and provides personalisation and feedback options which are associated with the non-linear multi-stream video show.

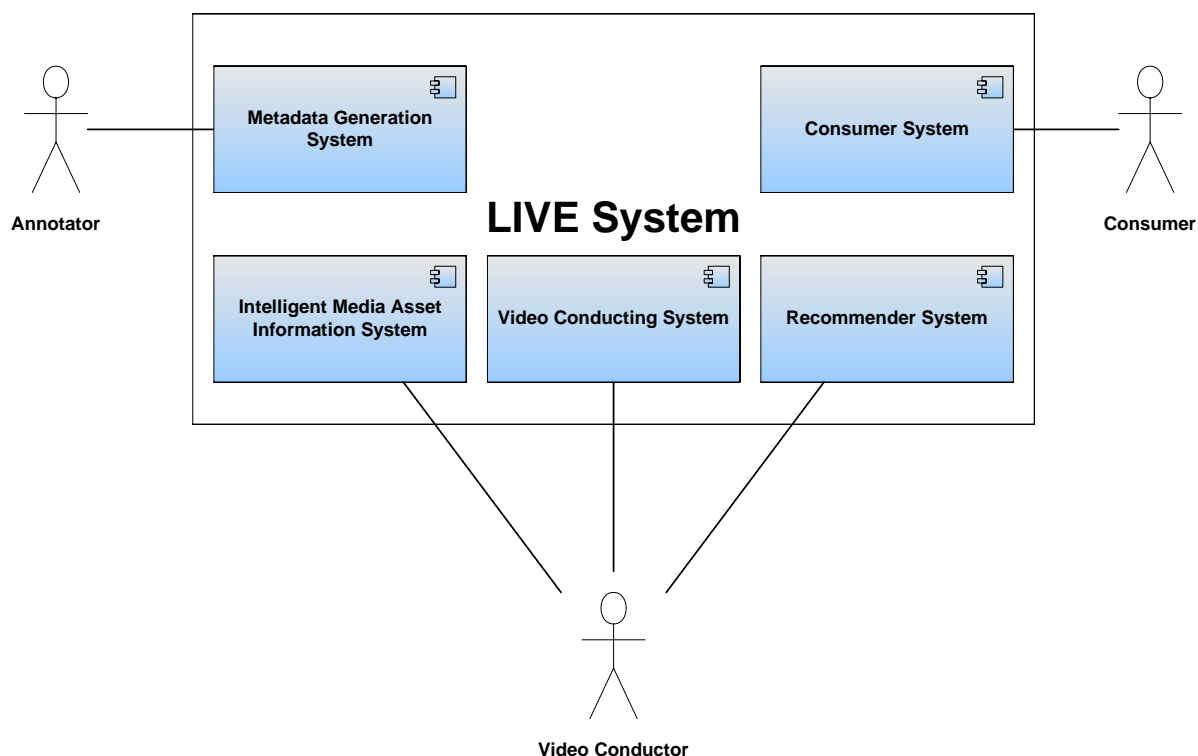


Figure 1: Building blocks of the LIVE production support system (acc. to [LIVE D9.3])

The *Intelligent Media Asset Information System* (IM AIS) and its interfaces and services form the main component of *Intelligent Media Framework* (IMF). Its role in the IMF is the management of the intelligent media assets and the support of other components (e.g. it provides the knowledge infrastructure for the *Metadata Generation System*, the *Recommender System* and the *Video Conducting System*).

1.1.2 Architecture of the Intelligent Media Framework

The system design of the Intelligent Media Framework is based on a combination of the classical three-tier architecture with principles of service oriented architectures (SOA). The IMF is responsible for storing data, for manipulating data and for distributing data to other components. These main responsibilities are addressed in different layers as shown in Figure 2:

- The **Service Layer**: Due to the IMF's nature as a middleware framework the top level layer consists of the services provided by the IMF to the other building blocks of the LIVE staging support system and external systems. These building blocks include components responsible for the manipulation, semantic enrichment and recommendation of data.
- The **Business Layer**: The business layer is designed to interact with the services and therefore is in charge of handling the data in specified incoming data formats. This in-

cludes the data formats specified by the IMF as well as standard data formats such as MPEG-7 ([Mas2002]), NewsML and SportsML2 (see [LIVE D7.1]).

- The **Storage Layer**: Firstly, the storage layer is responsible for the transformation of data into the data formats specified by the IMF data model and secondly for the provision of a persistence layer for the LIVE staging support system.

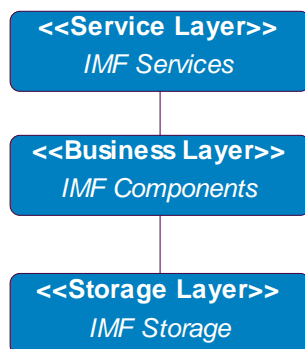


Figure 2: IMF layered architecture

The Service Layer of the IMF

The service layer as depicted in Figure 3 comprises the interfaces used by other systems:

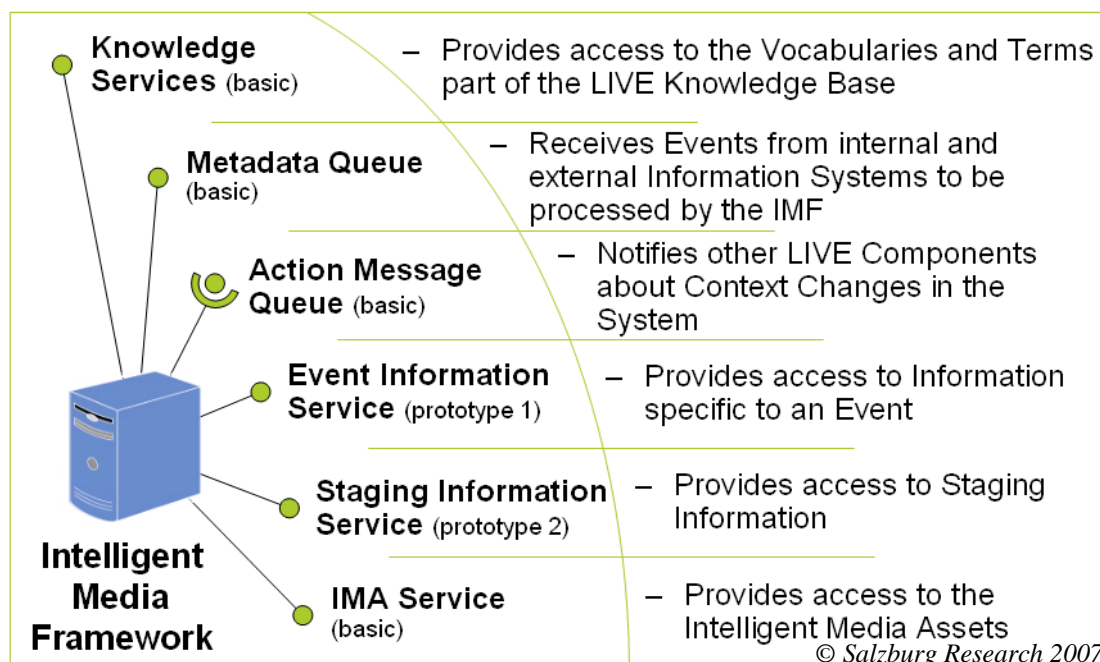


Figure 3: Overview of the IMF service layer

- The **Knowledge Service**: Provides access to the vocabularies and terms stored in the LIVE Knowledge Base. The knowledge base holds the terms used in the LIVE system. As an example, the Human Annotation Tool uses the Knowledge Service to retrieve relevant information of all participants of an event.

² <http://www.sportsml.org/> - last accessed: 31.07.2007

- The **Event Information Service**: Provides access to information specific to a currently staged event. During preparation, this includes the date and location of the event as well as its participants. During the staging of an event, related information is collected and stored.
- The **IMA Service**: Provides access to Intelligent Media Assets.
- The **Staging Service**: Provides Access to the staging information. This includes properties of the show as well preferences of the editors and video conductor(s).

Additionally, the IMF includes a messaging system to support the real-time aspect of the LIVE staging environment. This messaging system defines the following two queues:

- The **Metadata Queue** receives triggers form internal and external metadata generators. This mainly includes the Metadata Generation System and external event information systems (e.g. the sports information system of the Olympics).
- The **Action Message Queue** propagates messages about the current state of the event, staging and available content.

Knowledge Model for the LIVE Staging Domain

The foundation of the IMF is a knowledge model for the LIVE staging domain through which the envisaged *intelligence* of both, the media assets and the framework is achieved. The knowledge model for the LIVE staging domain was developed taking into account information about the event, the content (raw audio-video material and related metadata), the involved users and the knowledge about how an event is presented (“staging”). Various descriptive schemes for content objects (i.e. archival video material and live broadcast streams), the staging concepts (i.e. a description of how an event is broadcasted), the consumer profiles (individual or filtered group based profiles) and the professional user profiles (characteristics of the video conducting team) are integrated on a semantic level and form the core of the knowledge model of the LIVE domain.

The knowledge model of the Intelligent Media Framework is divided in the following sub-models:

- The **Term Model** defines the foundation of the knowledge structure. Its aim is to provide the necessary functionality for the definition of controlled vocabularies used within the Intelligent Media Framework.
- Based on the Term Model the knowledge models for the two core domains are defined: the **Event Model** and the **Staging Model**. The intention of both domain models is to define an ontology supporting the definition of formal statements about the current state (annotating/describing) or future state (planning) of the LIVE staging environment.
- On the top level of the knowledge structure we defined the **Intelligent Content Model**. This model defines the *Intelligent Media Assets* (IMAs) managing the relation between the knowledge created based on the *Event Model* and the *Staging Model* and the content objects.

1.2 Structure and Scope of the Report

1.2.1 Structure

This report, the public synopsis of the basic specification of the *Intelligent Media Framework*, is structured as follows:

- Section 2 analyses the LIVE staging domain and identifies the requirements of different agents in the LIVE staging process
- Based on the requirements identified in section 2, the knowledge and framework requirements defined in the “State of the Art Report for the Intelligent Media Framework” are summarised in section 3.
- The actual specification of the knowledge model for the Intelligent Media Framework is described in section 4. Currently three sub-models of the knowledge structure are specified: the Term Model, the Event Domain Model and the Basic IMA Model which constitutes the intelligent media objects for the LIVE staging domain.
- Section 5 describes the system architecture of the Intelligent Media Framework, introduces the system components and specifies the database schema and the services of the first prototype.
- Finally, section 6 summarises the major findings, assesses the IMF specification according to the requirements identified in section 3 and draws conclusions for future extensions of the Intelligent Media Framework in the course of the LIVE project.

1.2.2 Scope

This document is the public report on the basic specification of the Intelligent Media Framework and is based on the technical non-public deliverable D7.2 (“Basic Specification of the Intelligent Media Framework”).

1.2.3 Relationship with other Deliverables

The following *public* LIVE deliverables are related to this document and provide further information on the overall LIVE system:

- [LIVE D4.3] “Methods, design guidelines, workflows for online staging”
Addresses concepts and methods for conceivable TV formats entailing a multitude of live audio-visual material, which shall be transformed into an interactive live TV event for the consumer at home.
- [LIVE D4.5] “Specifications of Concepts and Professional User Interfaces for Live Staging with Consumer Feedback”
Upcoming deliverable, including a formalisation of the staging concepts developed in LIVE.
- [LIVE D7.1] “State of the Art Report (Intelligent Media Framework)”
Covers an analysis of technologies and standards related to the Intelligent Media Framework.

- [LIVE D9.3] “Public synopsis document on basic system architecture”
Contains a summary of the basic system architecture and the components of the LIVE system defined in the (not public accessible) [LIVE D9.2] “Basic system architecture”.

Throughout the document references to other LIVE deliverables are marked with [LIVE Dn.m] (“Dn.m” denoting the number of the deliverable, e.g. “D5.2”).

Further information about the LIVE project, its publications and deliverables can be obtained from the project web-site (www.ist-live-org), i.e. the “Knowledge and Dissemination Management System” (KDMS). The KDMS offers a facility to ask for access to documents which are not publicly available.

2 The Intelligent Media Framework

The Intelligent Media Framework acts as a middleware for the overall LIVE production support system for the LIVE staging process and the handling of media assets in an intelligent way. The framework integrates the various descriptive schemes for the content objects (i.e. archival video material and live broadcast streams), the staging concepts (i.e. a description of how an event is broadcasted), the consumer profiles (individual or filtered group based profiles) and the professional user profiles (characteristics of the video conducting team) on a semantic level. Based on an in-depth understanding of the domain the specific requirements of the framework can be defined. The remainder of this section therefore provides an overview of the LIVE staging domain and an analysis of the working environment and the requirements of the identified professional users of the intended LIVE System.

2.1 Overview on the LIVE Staging Domain

From a knowledge engineering perspective, the LIVE staging domain is characterised by four concepts: Content, Event, User, Staging. Figure 4 gives an overview of the domain:

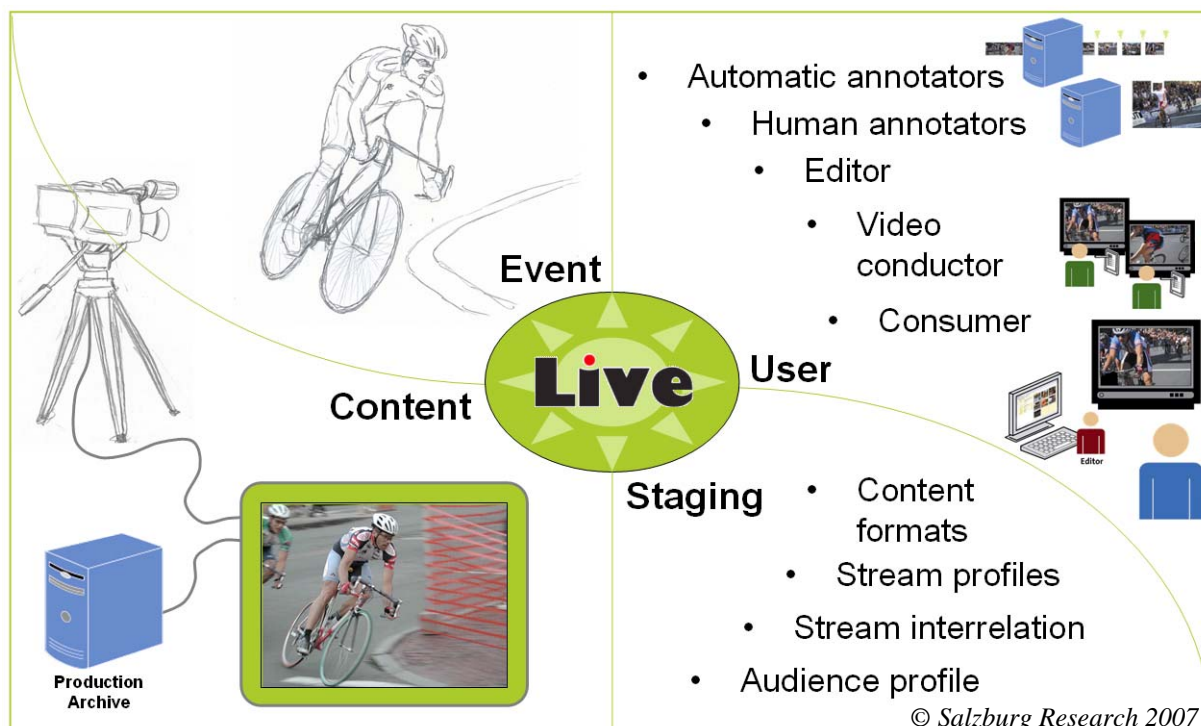


Figure 4: The LIVE domain

- **Event:** Information about the event to be staged is crucial for planning the show. During the event a production support system has to collect and analyze information in real time to support the LIVE staging process.
- **Staging:** Staging comprises all knowledge about how an event is presented to the consumer. This includes:
 - (i) content formats, which consist of general stream types, different types of switching between streams as well as screen layouts (e.g. “split screen” or “picture in picture”).

- (ii) Stream profiles define hot topics for a specific stream and the target audience of each specific stream.
- (iii) The stream interrelation defines situations, in which linkages in-between two streams are possibly.
- (iv) The audience profile models the target audience of the show.
- **Content:** The system has to handle *essences* (i.e. the raw video streams) as well as their metadata. Live *essences* are created by filming the event and activities around the event (e.g. interviews, analysis of experts, etc.). Content may also come from production archives containing prepared and previously annotated *essences*. To support recommendations, *essences* have to be annotated and these annotations have to be combined with real time and archived AV objects.
- **User:** The LIVE staging process defines different user roles:
 - (i) Automatic annotators which are software agents responsible for automatically detecting features within live streams.
 - (ii) Human annotators who manually annotate live streams.
 - (iii) Editors searching and pre-selecting content for the video conductor.
 - (iv) Video conductors who decide on how an event is staged.

A framework designed to support these four aspects of the LIVE staging domain has to consider the relationships between them, which are captured by the following relations:

- **Event ↔ Content:** Live content is produced in the course of an event. This creates an immediate relation between the content and the event. Additionally, the editor and the video conductor can use archive content to present special aspects of the event to consumers. This is another type of relation between event and content because it relates the archive content to the current state of the event and the show.
- **Event ↔ User:** Each identified professional user type will have a different role and thus, an implicitly different relation to the event. Automatic annotators will only have limited knowledge about the event and are just able to indirectly observe the event through the live content they process. Human annotators have extensive real-world knowledge about the event and are thus able to annotate live content based on this knowledge. An editor or a video conductor has a narrow focus since s/he is responsible for a specific part of the show. This is why s/he is just able to relate the course of the event to the part of the show he is responsible for.
- **User ↔ Staging:** Of the identified user roles, only the editor and the video conductor have a relation to the staging domain. The *staging concept* defines the profile of the show and the single streams which are part of the show. Editors and video conductors are responsible for staging these streams. The relation between the user- and the staging-domain is therefore about combining the personal preferences of the editor and the video conductor with the specifications in the staging concept. This combination of the personal profile with the profile of the show can be exploited to utilise content recommendations for editor and the video conductor.
- **Staging ↔ Content:** The staging concept defines a set of interesting topics for a show. A staging support system has to evaluate available content based on a given set of topics and recommend content to the professional users. Content includes live streams as well as prepared media assets. During the preparation of a show this relation is important to select available content from the long term archive and to decide about content which needs to be produced (e.g. interviews with favourites, reports about the location of an event, etc.).

- **Event ↔ Staging:** The staging concept represents a conceptualisation of the event. It describes the plan of how to present the event to the different consumers (groups) of the show. Therefore this relation is of utmost importance during the preparation of a show from two different perspectives:
 - (i) From the user-domain-view, there is the need to plan which persons are responsible for which part of the show and how these persons need to communicate.
 - (ii) From the content-domain-view there is the need to plan which type of content will be used in anticipated situations of the event (e.g. prepare reports of the national athletes, the locations, hot topics, moderations, etc.).
- **Content ↔ User:** The fundamental goal of the LIVE production support system is to deliver the right content to the professional user whose task it is to create a compelling set of streams for the consumer. Therefore such a system needs to have knowledge about the available content and the tasks, responsibilities and different perspectives of its users. This relation can be addressed from two directions:
 - (i) From the event side, it is necessary to annotate (interpret) content based on the current state of the event, e.g. automatic and human annotators are working in exactly this way
 - (ii) From the staging side, most decisions are based on specifications defined in the staging concept of the show, e.g. the editor(s) and the video conductor(s) are working in exactly this way, however they also consider information about the event, when they make their decisions.

Based on the identified basic concepts and their relations in the LIVE staging domain the following sections will analyse the requirements of different agents and investigate the professional users, their environment and the knowledge they create during the staging process.

2.2 Requirements of different agents in the LIVE staging process

This section considers the identified agents of the LIVE staging domain with respect to the knowledge they need and the knowledge they create. An agent is defined as a participant of the LIVE staging process that carries out some actions such as annotating or searching for content, taking decisions, etc. Note that this definition also covers software components such as the automatic annotation component. This analysis yields the requirements for an Intelligent Media Framework that is capable of supporting the tasks of these agents.

2.2.1 The “Levels of Meaning” classification schema

Annotating content includes two tasks: Firstly, to interpret the subject matter of the content and secondly, to formalise this interpretation based on a data schema. In communication theory there are several approaches to classify the interpretation context of information (e.g. in the area of information communication theory and in the area of semiotics - [Hab1981], [Har1968], [Cas2005]). For the purpose of LIVE the project team provided a classification schema to describe the interpretation context for the LIVE staging domain (see [LIVE D4.3]): Interpretation – in that context – can be described as a person (the annotator) conceiving a description about the meaning of a content item based on his current mental model (his context). That implies that different contexts will cause different interpretations of one and the same content item. LIVE defines the following four “levels of meaning”:

- *Particle* refers to an interpretation without any further context. Only the common sense meaning of the objects presented in the *essence* is used to create an interpretation. Examples for interpretations on the particle level are: a face, a ball, a cut, etc.
- *Local* refers to an interpretation which tries to relate the illustrated objects present in the footage with “particles” identified in the current context. Examples for interpretations on the local level are: the “person” related to a detected face, the annotation “football” for a detected ball.
- *Global* refers to interpretations conceived based on relating the subject matter of AV objects with the mindset behind the global schema of the story. This can result in divergent interpretations if different stories represent a special viewpoint on the same scene. An example for an interpretation on the global level is: the different views on a foul in a soccer game by supporters of each of the two teams.
- *Universal* refers to an interpretation based on a context which has a foundation outside the borders of the story. Universal interpretations relate the meaning of the actual AV object to abstract conceptualisations. Examples for interpretations on the universal level are: “David beats Goliath”, the “rise like a phoenix from the ashes”, the “eternal struggle between good and evil”.

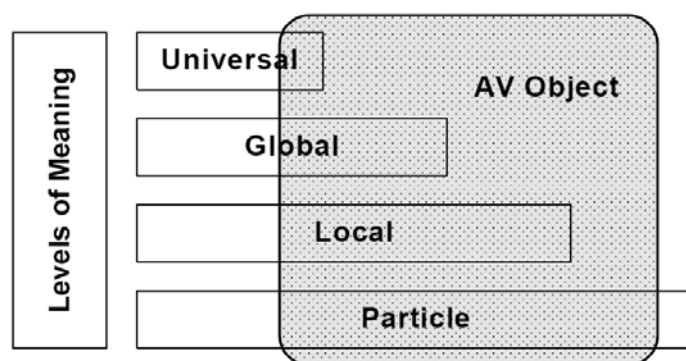


Figure 5: The four levels of meaning (acc. to [LIVE D4.3])

The following analysis of the different user roles of the LIVE staging environment uses these levels of meaning to identify knowledge requirements of the different users. This is done by first assigning annotations created by a user to a level of meaning and then evaluating the knowledge requirements of that user role based on the needed contextual information to create such kind of interpretations.

2.2.2 Automatic annotators

Automatic annotators are software agents used to annotate live streams. They are developed as part of the metadata generation system of the LIVE project.

The environment consists of an event, one or more live streams, one or more automated annotators and the IMF. The automatic annotators analyse the input streams and send recognised facts to the IMF.

In most cases the interpretation of low level features by automatic annotators will be on the *particle* level (i.e. without further context of the event and the staging concept). Therefore automatic annotators do not need knowledge from the IMF. The only requirement is the definition of a controlled vocabulary, that helps to identify and consistently use features that are detected by the automatic analysis tools provided by LIVE’s Metadata Generation System, e.g. the term “close-up” denotes the fact that a person or object was detected to be tightly

framed in a video stream by the close-up detection tool, the term “face” denotes the fact that a face was detected in a video stream by the face detection tool.

2.2.3 Human annotators

For each event footage from one or several live streams is available. Each live stream is annotated by a human annotator. The human annotator uses LIVE’s Human Annotation Tool to create statements about recognised facts in the annotated stream. The IMF provides the contextual information about the event and stores the annotations created by the human annotator. In contrast to automatic annotators, a human annotator has always a mental model about the state of the event. That means, even if the LIVE production support system does not provide him/her with contextual information about the event he/she will use his/her own mental model to create annotations about the watched content. Therefore annotations of a human annotator constitute an *interpretation at the local level of meaning*.

To support the human annotator in his/her work and to get computer processible annotations the IMF must provide

- (i) a controlled vocabulary for all the things the human annotator has to recognise, and
- (ii) a formal model which defines possible sentences (formal statements) to be used for the annotations.

Annotations created by a human annotator are all links to a specific part of the *essence* (i.e. the raw video stream). Since human annotators use some contextual information to create these annotations they may also include facts which are not directly shown within the *essence*. Therefore these annotations have to be treated as interpretations of the live stream by the actual annotator and different annotators may interpret the same *essence* differently. This is also relevant for the recommender system calculating recommendations based on such annotations/interpretations.

2.2.4 Editor

An editor has the responsibility to evaluate the current course of the event and the available content with respect to the profile of the stream as defined by staging concept of the show. He suggests relevant content to the video conductor, who based on the recommendations of the editor(s) decides on the content actually streamed. Figure 6 shows the environment of an editor within the LIVE production support system.

The editor sits in the control room in front of the monitors. S/he uses a staging user interface provided by the LIVE production support system. This application uses knowledge provided by the IMF and is based on functionality implemented by the recommender system (another sub-system of the LIVE production support system).

The editor – as a knowledge worker – has the possibility to search for prepared content fitting the current staging situation. Based on this work s/he then has to select some of the available content and suggest it to the video conductor. By selecting and suggesting content the editor annotates (interprets) the *essence* to be “appropriate” for the current state of the stream/show and the course of the event.

Such annotations create a link between the staging situation, the event situation and an *essence*. That means that content is annotated by an editor based on its meaning based on the special viewpoint the staging concept has on the current course of the event. This corresponds to the global level of meaning.



© Salzburg Research 2007

Figure 6: LIVE staging environment of an editor

Available content includes the live streams, near live content (e.g. from a loop server) and content available in a production archive. Relevant knowledge includes event and staging information. Event-driven knowledge includes information about the event (participants, time table, etc.), and the actual state of the event (incoming annotations from automatic and human annotators as well as information from external information systems). Staging based knowledge includes the staging concept (profile of the streams, switch point definitions, audience profile, etc.) and information about the state of the show (used content, consumer feedback, etc.).

The knowledge requirements of an editor are already outlined by the description of the environment. In comparison to a human annotator the editor needs additional knowledge about the staging concept. Annotations should be automatically created by tracking the interactions of the editor with the staging user interface. Such annotations

can be linked to the *essence* selected by the editor and represent an interpretation of the meaning of the *essence* in context to the show as conceived by the actual editor.

2.2.5 Video Conductor

A video conductor is the person responsible for staging the show. He/she selects the content to be part of a stream and decides on switching points between streams.

The video conductor is working in the control room. He instructs the editor(s) to search for content based the current state of the event and the show, receives pre-selected recommendations and decides on the actually streamed content.

Decisions of the video conductor advance the show, by adding a new piece of content to it, or by linking to streams at a switching point. Because of that it is valid to assume that there is – at least on some occasions – some “universal” schema involved when a video conductor makes decisions on how to stage an event (e.g. David beats Goliath, to rise like a phoenix, something far away from the event, show the person behind the athlete...). This implies, that decisions of a video conductor annotate the used *essences* with interpretations on a universal level of meaning.

The knowledge requirements of a video conductor are similar to the one of the editor described in the section above. In addition the video conductor needs a controlled vocabulary for these “universal” schemas to refer to them explicitly while staging the event. Note that such a controlled vocabulary is not sufficient to formally describe the semantics of such “universal” schemas.

Annotations of a video conductor represent decisions about the usage of an *essence* as part of one of the streams that make up the show. Therefore these annotations can be assigned to the *essence* as well as to the stream. By making decisions about the usage of a specific *essence* the video conductor also updates the state of the show.

3 Requirements for the IMF

Based on the analysis of the LIVE staging domain (section 0) this section extends the requirements already described as part of [LIVE D7.1]. These requirements are split into two categories. Firstly, requirements referring to features of the knowledge structure of the IMF and secondly, requirements for the software architecture of the IMF.

3.1 Knowledge Requirements

Based on the combination of technical requirements identified within the Basic System Architecture ([LIVE D9.3]) with those identified in the analysis of the LIVE staging domain the following five core aspects of an intelligent content model used within the IMF where identified.

Make information machine processible

Knowledge must be encoded using a formal language: Knowledge content must be interpretable by machines in order to enable the implementation of systems supporting knowledge workers in handling content objects. Therefore, it is a requirement to encode the knowledge in a formal language. The use of statistical methods only – as seen in most web based tools – will not be sufficient in closed, corporate environments.

Support cross-domain interoperability

Some aspects of knowledge about content are independent of the application domain. Media coding information and rights management are the most prominent examples. Established standards of these domains must be respected and aligned to the concept of knowledge content.

Allow multiple interpretations

This aspect is typically neglected by traditional content management systems. However, as the new wave of social software shows, different interpretations are an important factor in the daily work of knowledge workers. This is not only true for web based communities as shown for LastFM and Flickr, but also for corporate settings, where different employees with different job roles have different perspectives on the same content objects. This introduces the requirement that knowledge content must support multiple descriptions of the meaning of content objects.

Unleash hidden information

Link content with knowledge that cannot be directly derived from the content: Media content only *contributes* to a larger narrative. Even though only a small part of the story may be supported by some content it must still be possible to state that the whole story is related to the content. Looking at it the other way round, this requirement explains also the significant weakness of systems which only rely on feature recognition to explore the meaning that is hidden in media objects.

Make knowledge independent of content

In traditional media asset management system knowledge is typically dependent on the content object. As shown, this is not a valid assumption for most of the interrelations between knowledge and content. Particularly planning activities of knowledge workers first create a description of the topic of interest and later search for suitable content objects which support the conceived story. In general, descriptions of the subject of the content seem to be more dependent on the topic and the user than on the content itself. This introduces the requirement that knowledge descriptions must be autonomous from the content objects.

3.2 Framework requirements

The Intelligent Media Framework needs to provide services to meet all the identified requirements of the professional users. In addition the Intelligent Media Framework must also support the management of the knowledge structure and the instance data used and created within the LIVE staging process.

Knowledge and information integration

The LIVE production support system depends on the integration of different internal and external information sources. Internally, information of the different tools used by the different professional users must be integrated. Also information about the show and the event must be connected with annotations created during the staging of a show. External information system like event information systems, time keeping systems, etc. must also be integrated into the Intelligent Media Framework. Where possible, standards should be used for the definition of such interfaces.

Management of intelligent media assets

Newly created knowledge and the links between the knowledge and some content must be managed by the Intelligent Media Framework. This does not only include the storage and access to such objects, but also the import and export based on broadcasting domain standards. Also an important aspect of this topic is the combination of different modelling technologies like relational models, XML schemas and RDF/RDFS/OWL ontologies.

Real-time delivery of knowledge

Staging an Event is a real-time task. The professional users need to respond to changes of the staged event within some seconds. Therefore, the LIVE production support system needs to provide information about the event and recommendations of content in a near to real-time manner. This has some implication on the design of a knowledge framework:

- It is not advisable in such a setting to wait for the completion of an information: If an athlete is recognised in a live stream this information needs to be communicated instantly by the system. Without this real-time requirement one could also wait until the end of the segment showing the athlete.
- Messaging based services (e.g. publish and subscribe) are typical for real-time information systems. The IMF needs to provide such services in addition to classical pull type services used to access static information in the knowledge base.

4 A Knowledge Model for the IMF

Based on the definition and analysis of the LIVE staging domain, this chapter describes the structure of the different knowledge models defined for the Intelligent Media Framework. The knowledge models are designed to reflect the knowledge requirements defined in section 3.1 (“Knowledge Requirements”, p. 17).

Starting with an overview of the internal structure of the knowledge models, the different models are subsequently explained in more detail.

4.1 Knowledge Structure of the IMF

The knowledge model used by the Intelligent Media Framework has four parts which are outlined in Figure 7:

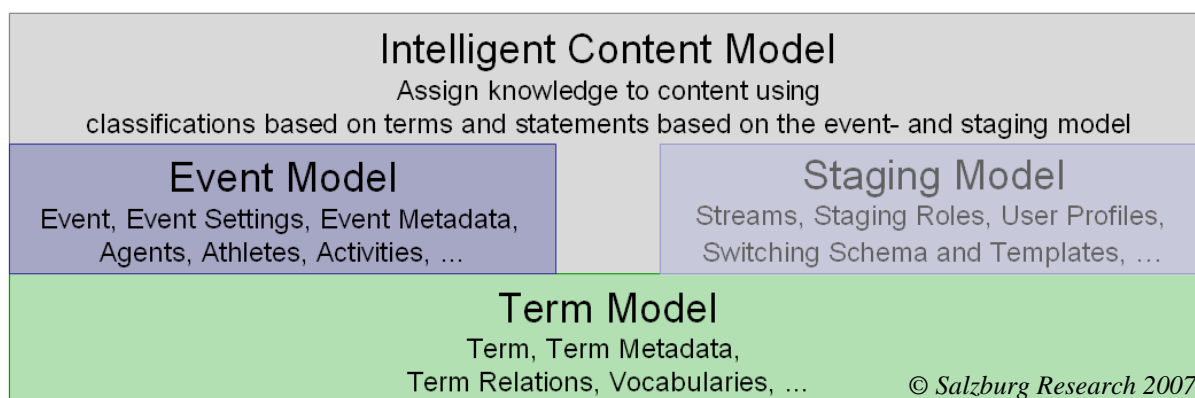


Figure 7: Knowledge structure of the IMF

- The knowledge structure for the IMF is grounded in the **term model**. The term model provides means to define *controlled vocabularies* which are to be used within the IMF: It supports the definition of vocabularies, terms, term metadata and semantic relations holding between terms. In addition the term model defines a *base typology for terms – the term type hierarchy* – which provide means to easily define controlled vocabularies for new application domains. Based on the term model knowledge models for the core domains of the LIVE staging environment are defined. Their intention is to provide means for defining formal statements about the current state (annotating/describing) or future state (planning) of the LIVE staging environment, which refer to terms in the controlled vocabulary.
- One of the two core domains of the LIVE Staging Environment is the event domain. The formal model for the event domain – the **event model** – includes the context of an event, its participating actors as well as activities performed within the context of an event.
- The second core domain describes how an event is staged. This includes descriptions of the streams, the streams interrelations, and of the different persons in the video

conducting team including their roles, tasks and responsibilities. The domain knowledge of the staging domain will be captured in the **staging model**³.

- The **intelligent content model** forms the upper most layer of the knowledge model⁴. The aim of the intelligent content model is to provide a formal way to link content with available knowledge about the content (in a simplified way this link can be considered as a kind of “annotation”). Knowledge is formalised based on the term-, event- and staging model. In addition the creator of the knowledge (the “annotation” in the above metaphor) is preserved. Therefore each knowledge description represents a formal interpretation of the content by a user.

The knowledge models for each layer are described in more detail in the subsequent sections.

4.2 Term Model

The term model forms the grounding layer of the knowledge structure: all controlled vocabularies used by the LIVE production support system are instantiated based on it.

The motivation for defining a new term model was the fact that data structures for defining terms as established by existing standards like MPEG-7 ([RMKP2002]) or NewsML⁵ do not fulfil all the requirements of the IMF: These models are typically tailored to the specific requirements posed within the context of that standard. This usually precludes the formalisation of higher level formal statements. However, the IMF demands a clear definition of the semantics of terms and semantic relations in between terms. This is why the term model is based on the simple knowledge organisation system (SKOS) and on parts of the vocabulary as defined by the NewsML standard.

Besides the model for defining terms and vocabularies, the term model defines the so called term type hierarchy which is based on a categorisation defined in DOLCE foundational ontology [Mas2002] and the Descriptions and Situations (DnS) Ontology [Gan2003], an extension of DOLCE. This foundation makes it possible to apply definitions of the DOLCE foundational ontology⁶ to all terms of each category.

The following sub-sections provide additional information about the different aspects of the term model.

4.2.1 Application of SKOS

The Simple Knowledge Organisation System (SKOS) is a model created by W3C⁷ to define thesaurus-like knowledge structures which are compatible with semantic web technologies. SKOS is based on RDF and RDFS, and a version of it exists for use with OWL.

³ The staging model will be developed during the second iteration cycle. Currently the staging model is not defined and therefore this model is depicted semitransparent in the above figure.

⁴ For an introduction to intelligent content models see [LIVE D7.1] (“Approaches to Intelligent Content Models”)

⁵ <http://www.newsmml.org/> - last visited: 31.07.2007

⁶ The DOLCE foundational ontology is a formal knowledge model which can support automated reasoning. Such formal models are required for autonomous intelligent agents and some aspects of the IMF can be regarded as such an agent.

⁷ The World Wide Web Consortium (W3C) is responsible to develop interoperable technologies for the Web. <http://www.w3.org/2004/02/skos/> – last visited 29.05.2007

The main elements of the SKOS model as depicted in Figure 8 are firstly a metadata model for terms and secondly a model for the description of semantic relations between terms. By using *semantic relations* as defined by SKOS we are able to build up hierarchies (*broader* and *narrower*) and non hierarchic relations between terms (*related*). *Broader* links a SKOS concept with concepts more general in meaning. *Narrower* is the inverse relation. The semantic relation *related* can be used to express the fact, that two concepts are related with each other. Within the IMF terms are defined based on the semantics defined by the SKOS standard. We defined an XML Schema to encode terms which can be used to define new terms and which is able to hold all information defined by the SKOS standard. In addition the schema defines additional metadata based on the assigned *term type* of a term (e.g. terms which define persons use v-card properties, terms which define geographical points have latitude/longitude and address information assigned).

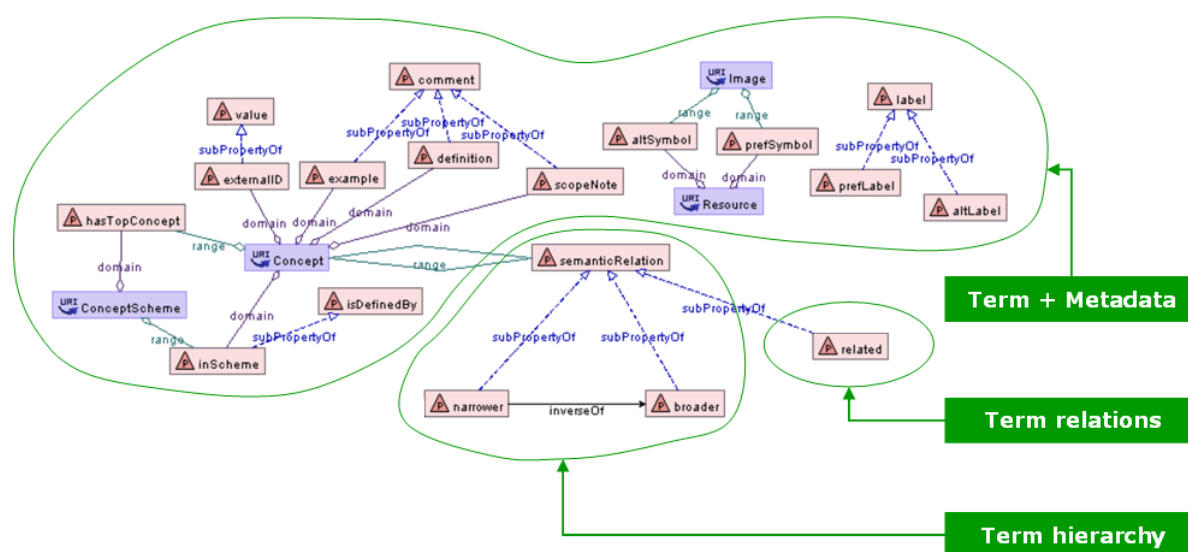


Figure 8: SKOS model

4.2.2 Term Type Hierarchy

The *term type hierarchy* establishes a category system for terms. Categories have assigned semantics which enforces proper use of terms of a domain with higher level models. Its aim is therefore to provide a very simple but powerful model to structure terms of a domain. The semantics for the categories are based on the DOLCE foundational ontology [Mas2002] and one of its extensions, the Descriptions and Situations (DnS) ontology ([Gan2003]). Figure 9 depicts the term type hierarchy as currently used within the Intelligent Media Framework⁸.

Term is used as abstract super class for all defined term types.

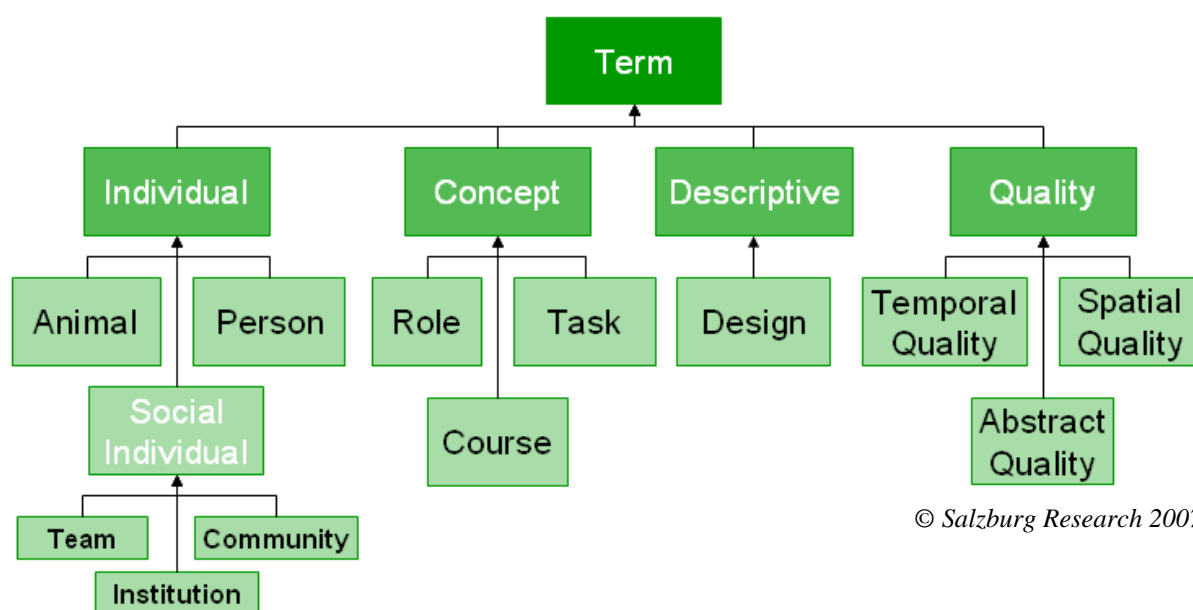
The term type *Individual* covers *Persons*, *Social Individuals* and also “named” *Animals* like the race horse of a jockey. *Social Individual* has the three sub categories *Team*, *Institution* and *Community*. *Social Individuals* are naturally and socially constructed individuals which have a

⁸ White letters in Figure 9 indicate that a term type is defined as an abstract category [Mas2002].

name and concrete identity (e.g. “Apple Inc.”, “Manchester United”, the “Ubuntu Community”⁹, etc.).

The term type *Concept* covers all kind of conceptualisations to classify objects, individuals and activities based on the current situation. *Roles*, *Tasks* and *Courses* are sub categories of the term type *Concept*.

- *Roles* classify objects and individuals and therefore are conceptualisations about the function of an individual in a specific situation (e.g. a person may be classified – in the context of the actual event - by the *Role* “winner”).
- *Tasks* classify activities (e.g. an activity may be classified by the task “sprinting”).
- The sub category *Course* is used to classify the chain of activities on the granularity level of events (e.g. an event may be classified by the *Course* “winning ceremony”).



© Salzburg Research 2007

Figure 9: Term type hierarchy

The term type *Descriptive* catches *Designs*, *Plans* and *Methods*. *Descriptions* are created to plan future situations or to describe current or past situations. We only use the sub concept *Design* because this type may be used for layouts of well known events like “Tour de France”, “Olympic summer games”, “football world cup”. *Descriptive* terms are therefore used to classify real world settings or in other word to state that a situation satisfies a design/plan. As an example the “Olympic summer games in Beijing 2008” satisfy the principal *Design* of “Olympic summer games”. *Plans* and *Methods* are more relevant for the description of business processes which is not in the focus of the IMF.

The term type *Quality* covers *Temporal*, *Spatial* and *Abstract Qualities*. *Qualities* are used to parameterise objects based on time space and abstract classification systems (e.g. genres). Cities, countries, points of interest... correspond to *Spatial Qualities*. *Temporal Qualities* include categorisations like morning, evening, summer, winter, ... *Abstract Qualities* correspond to

⁹ The Ubuntu Community as an example for a community around a (open source) product. <http://www.ubuntu.com/community> - Last visited 27.08.2007

human made classification systems like the different genres defined by the MPEG-7 standard or the NewsCodes as defined by the IPTC.

4.3 Event Domain Model

The event domain model is based on the semantics defined by the term model and the term type hierarchy. It defines a knowledge structure which allows specifying statements about events, its participants and activities which are part of an event. The aim of the event domain model is to provide means for the creation of statements describing the current state of an event. The definition of event types, different roles of participants, tasks and courses classifying activities of an event is already covered by the term model.

The Event Domain Model which as depicted in Figure 10 is based on the *Participation* and the *Role Task Pattern* as described in [Gan2005]. This figure provides an overview of the event domain model, its concepts and relations¹⁰:

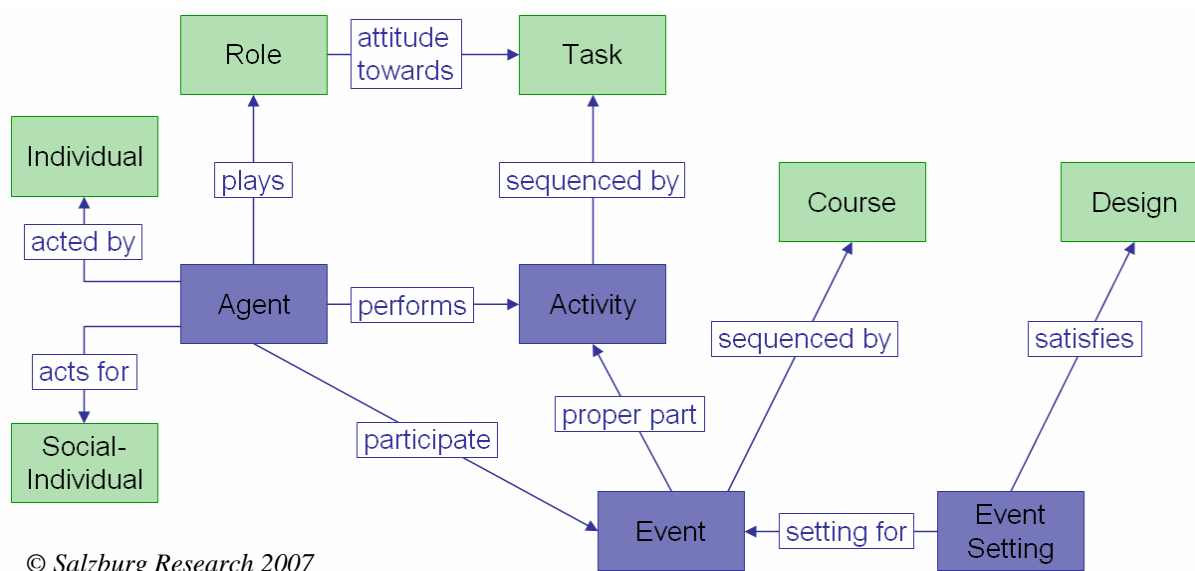


Figure 10: Event domain model

The Agent Model

An agent represents an individual (person, team, organisation ...) playing a role in a given situation. The event model defines an *Agent* as playing a *Role* and being impersonated by an *Individual* (this relation is called "acted by" in our model). In addition an *Agent* may act for a *Social Individual*. An *Agent* is therefore a context dependent statement in which a *Role* is temporary assigned to an *Individual* and the *Individual* can act for its own or a *Social Individual*. The context is provided by the *Event Setting* (see below for more information on the definition of *Event* and *Event Setting*). The *Agent* therefore provides the utility to repurpose the role of *Individuals* based on the current situation. *Agent* can have a *Spatial* or *Abstract Quality* assigned as a parameter (e.g. his current location).

¹⁰ Green/light boxes refer to term types defined by the term model. Blue/dark boxes are concepts defined within the event domain model.

Defined relations of the agent model are:

- *Agent* plays a *Role*. The *Role* classifies the *Agent*.
- *Agent* is acted by an *Individual*
- *Agent* acts for a social *Individual* (optional)

Other possible relations include:

- *Agent* performs (participates) in *Activities*
- *Agent* participates in an *Event*
- An *Event Setting* provides the setting (context) for an *Agent*

Example 1: The person “Lance Armstrong” as an *Individual* is part of the controlled vocabulary for bicycling. The *Role* “Winner” is defined in the controlled vocabulary for sports. Based on these static terms, it is possible to define an *Agent* playing the *Role* “winner” and acted by the *Individual* “Lance Armstrong”. Such an *Agent* needs a setting (context) to be valid, e.g. the *Event Settings* “Tour de France 1999, 2000 ... 2005”.

Example 2: To demonstrate the difference between the static terms in the controlled vocabulary and the context dependent statements based on the event model a different setting is chosen: the chemotherapy of “Lance Armstrong” in 1996. There, an *Agent* acted by the *Individual* “Lance Armstrong” is playing the *Role* of a “Patient”. This *Agent* would not act for any *Social Individual*.

The Activity Model

Within the event domain model activities represent actions performed by agents. Activities can be classified by tasks (e.g. the activity of the agent “Lance Armstrong” crossing the finish line can be classified by the task “Finishing”). In more detail, each *Activity* is defined as sequenced by a *Task* and performed by one or more *Agents*. In principle an *Activity* may be part of another *Activity*, but this relation is restricted within the event model to the fact, that an *Activity* may be part of an *Event*.

Defined relations of the activity model are:

- An *Activity* is sequenced by a *Task*. The *Task* classifies the *Activity*
- An *Activity* is performed by *agents*. *Agents* participate in *activities*. Via participation *Activities* are linked to *Roles* and *Individuals*.
- An *Activity* is part of an *Event*. (optional)

Other possible relations include:

- An *Event Setting* provides the setting (context) for an *Activity*.
- *Activity* has a *Temporal* or *Abstract Quality* assigned (e.g. the start time and duration of the *Activity*)

The Event Model

The word “event”, as used in natural language, can be mapped to three different meanings:

1. A *Design* can be referred to by saying sentences like “Olympic summer games are the biggest sport event in the world.” This meaning of event is covered by the concept *Design* which is part of the term type hierarchy as defined within the term model.

2. The word “event” can be used to refer to a situation. That means it does not refer to the specific type of an event, but to an actual setting for an event. A typical sentence using event in this context would be “The 2008 Olympic summer games in China will be the biggest sport event in the history of mankind”. In this case event refers explicitly to the setting of a specific instantiation of the *Design* “Olympic summer games”.
3. The third possible usage of the word “event” is to refer to the course of activities. In this case event refers to a happening covering a set of *Activities* and describing the sequence in which these *Activities* happen. A typical usage of the word "event" in this meaning is “The event develops in an unexpected direction” or “The individual time trial event starts at 9:00 and is expected to be finished at around 12:30”.

The first possible meaning of the word “event” is of limited relevance to the event domain model, because *Designs* are assumed to be static (they do not need a context) and are therefore part of the term model. Actual designs are part of the controlled vocabulary.

The differentiation of the second and the third meaning is of importance for the event domain model.

The second possible meaning of the word “event” is represented in the event model by the concept *Event Setting*. An *Event Setting* represents a situation, which satisfies the *Design* of an event (e.g. the *Event Setting* “Tour de France 2007” satisfies the *Design* of “Tour de France”). Each *Event Setting* provides the setting for one or more *Events* (in the third meaning).

Defined relations are:

- An *Event Setting* satisfies an (event)-*Design*. The *Design* provides a description of the *Event Setting* on a conceptual level.
- An *Event Setting* is the setting for zero or more *Events*

Other possible relations include:

- An *Event Setting* provides the setting for *Activities*. *Activities* can be directly linked to an *Event Setting* if no event covering these *Activities* is defined.
- An *Event Setting* provides the setting for *Agents*. This allows linking an *Agent* with an *Event Setting* without the requirement that he participates in an *Activity* or an *Event* part of the *Event Setting*.
- An *Event Setting* can have *Abstract Qualities* assigned.

The third possible meaning of the word “event” is modelled by the concept *Event*. The concept *Event* represents a happening sequenced by a *Course* and a container for zero to many *Activities*. *Agents* can participate within the course of the *Event*.

Defined relations are:

- An *Event* is sequenced by its *Course*. The *Course* classifies the *Event*.
- *Agents* participate in an *Event*. That includes also that *Agents* participate in *activities* part of an *Event*
- An *Event* has *Activities* as its parts. An *Event* describes the sequence of *Activities*.

Other possible relations include:

- The setting (context) of an *Event* is defined by zero to many *Event Settings*. This models, that an *Event* can be viewed in different settings. E.g. the *Event* “individual

time trial” a stage of the “Tour de France 2007” (*Event Setting#1*) is also part of the “Pro Cycling Tour 2007” (*Event Setting#2*).

- *Event* can have a *Temporal* or *Abstract Quality* assigned (e.g. the start time and duration of the event)

4.4 Basic IMA Model

The Intelligent Media Asset (IMA) model is based on the term model, the event domain model and the staging domain model. The IMA model defines the intelligent content model used within the LIVE production support system. Its main goal is to provide model to formally describe the possible interrelations between content, knowledge and users working with the content and knowledge.

The IMA model depicted in Figure 11 is based on the Knowledge Content Object (KCO) - model (see [Beh2005], [LIVE D7.1]):

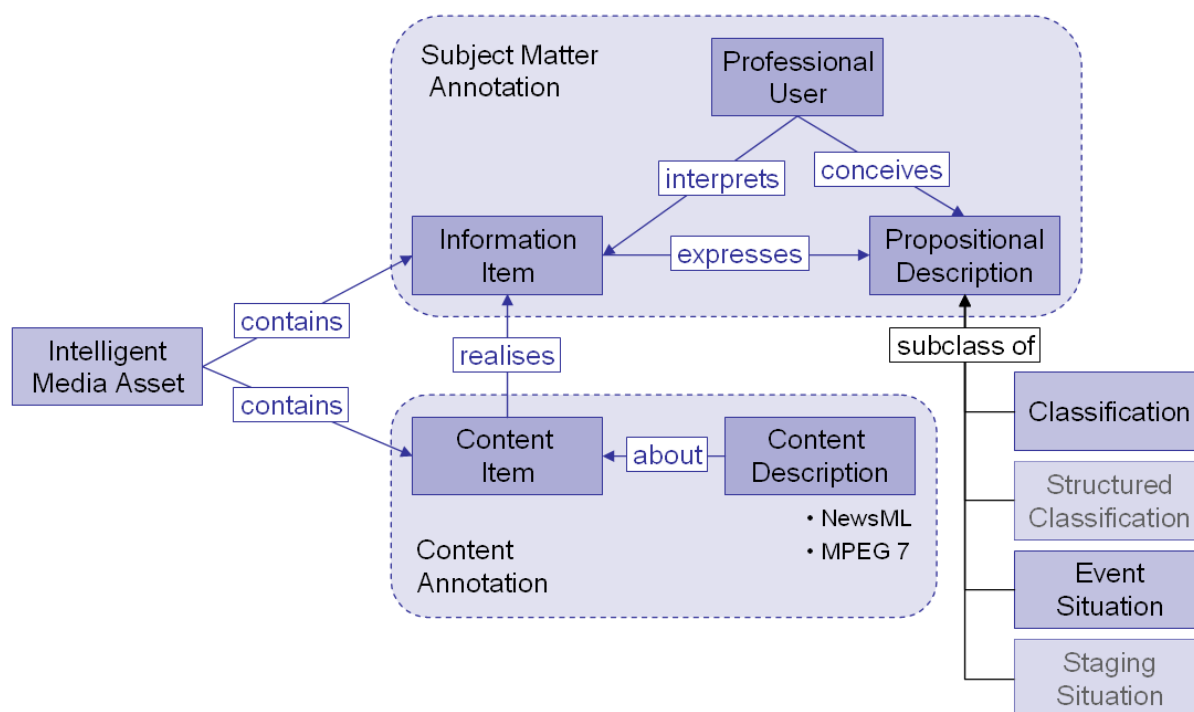


Figure 11: IMA model

© Salzburg Research 2007

The core parts of the IMA model are content annotations which provide information about the *essence* (i.e. the raw video stream) and subject matter annotations which provide information about the subject matter of the *essence*.

The **content annotation** part is made up of *Content Item(s)* and *Content Descriptions* about *Content Items*: *Content Items* are needed to provide a unique ID for *essences*, and *Content Descriptions* provide metadata describing features of the *essences* (e.g. storage location(s), encoding, access rights, or publication rights). For the *Content Descriptions* existing metadata standards like NewsML, MPEG-7 or MPEG-21 may be used.

Subject matter annotations are created by professional users interpreting an incident (a situation spanning a number of objects, individuals and activities) which they observe. *Professional users* formalise their interpretations by conceiving *Propositional Descriptions*. *In-*

formation Items may combine different interpretations of different *Professional Users* on one and the same incident. Therefore an *Information Item* can contain different interpretation by different users about the same incident.

A *Content Item* realises an *Information Item* if the described incident is part of the subject matter of the *Content Item*. For an IMA it is assumed that the *Content Item* part of the IMA realises all *Information Items* of the IMA.

The *realises*-relation between *Information Item* and *Content Item* can be applied in two different use cases: for planning activities and for the annotation of content objects.

During planning activities, professional users create *Information Items* based on *Propositional Descriptions* defining possible topics for the show. Based on that, content *realising* these *Information Items* can be searched/produced and prepared for the LIVE production process. The central elements of that usage case are the *information items* describing planned elements of the show. The prepared content is then *realising* these *Information Items*.

Annotation is the opposite case: A given *Content Item* is annotated by a *Professional User* which is capable of interpreting the subject matter of an *essence*. During the LIVE production process, the creation of IMAs is driven by the available content, therefore *content items* are the dominating elements. The *Information Items* representing annotations are attached to – and therefore dependent on - these *Content Items*¹¹.

Information Items may be personalised based on the IMA model. This is done by keeping information about the *professional user*, who codified his interpretation of the *Information Item* by creating a *Propositional Description*. This information could be used by a recommender engine to provide personalised recommendations based on collaborative filtering.

The IMA model defines several different types of content annotations:

1. Simple content *Classification* based on a list of terms taken from the controlled vocabulary.
2. *Structured Classifications*, which will extend the *Classification* type by providing, constructs to split up the classification along different dimensions. Possible dimensions will include time, location, event and staging.
3. *Event Situation* description based on statements defined in the event model
4. *Staging Situation* description based on statements of the staging model¹²

Example: Let us assume that an editor creates a description about an bicyclist which was born in a village which is on the way of the 6th stage of the Tour de France. This propositional description in combination with the author (i.e. the editor) forms an *Information Item*. Next the editor searches the long term archive for a report about the athlete. Additionally, he has the possibility to take an interview with the athlete or the parents of the athlete in the preparation phase. All the content collected realises this *Information Item*.

During the staging of the 6th stage of the Tour de France the athlete is part of the top flight. When the group reaches his birth place a human annotator creates a propositional description containing the athlete, the village, “top flight” and the time. This description in combination

¹¹ The two usage cases are related to the planning- and production-cycle as defined in section 2.1 (“Overview on the LIVE Staging Domain”, p. 11).

¹² These will be formalised as soon as the staging model will be implemented

with the human annotator builds up another *Information Item* which is realised by the according segment of the live stream.

The IMF now contains two Intelligent Media Assets (IMAs) with similar propositional descriptions: The first one was created during the preparation phase of the event and the second one was created based on the current situation of the staged event. The recommender engine can now recommend the prepared IMA to the video conductor(s) responsible for staging the different streams of the show. Note that such a recommendation is calculated on the subject matter level: The interpretation of the editor preparing the content is matched with the interpretation of the human annotator about the current situation of the event. Personal preferences (e.g. a video conductor does not trust annotations of the editor preparing the content) could strongly influence recommendations on the subject matter level.

5 Architecture of the Intelligent Media Framework

An introduction of the overall system architecture and the main building blocks of the LIVE production support system and an overview of the general architectural approach for the system design of the *Intelligent Media Framework* (IMF) was provided in section 1 (see p. 5): There the IMF was introduced to be based on a combination of a classical three-tier architecture with the principles of service oriented architectures (SOA). We further showed that the framework provides the following services to other LIVE subsystems:

- **IMA Service:** a service to create, annotate and manage the intelligent media assets making up the show under real-time conditions. This service operates on the metadata level and does not actually store any raw video streams (they rather reference these *essences*).
- **Event Information Service:** a service to manage and deliver information about the staged events, e.g. the schedule of the contests and races, the participating athletes, the results, etc.
- **Staging Service:** a service to manage and deliver information on the way how a live broadcast of a sporting event is presented, e.g. which types of switching concepts are available and used in a certain concept of a show, which basic dramaturgic concepts are appropriate according to the disposition of the production team and/or a predefined mood of the show, etc.
- **Knowledge Service:** A service to access the vocabularies and the terms of the controlled vocabulary constituting the knowledge base of the LIVE staging domain.
- A messaging system consisting of a **Metadata Queue** and an **Action Message Queue** to support the real-time aspects of the staging process by offering subscription methods to other subsystems.

5.1 System Design of the Intelligent Media Framework

This section shows the system design of the overall LIVE production support system and introduces the components and interfaces of the Intelligent Media Framework at some level of detail.

5.1.1 Components of the LIVE Production Support System

The service layer of the IMF is designed to provide its functionality to the other components of the LIVE production support system. To outline the usage of the services the subsequent section takes a more detailed look at the components of the LIVE production support system in general and the IMF in particular: Figure 12 gives the “big picture” of the components of the LIVE production support system and sets the components constituting the IMF (respectively the IMAIS) in relation to the overall system: The components related to the IMAIS and the IMF are marked with two rectangular boxes. While the vertical ordering of the components in the component diagram follows the layered architecture of the IMF and the LIVE staging support system in general, the horizontal position of a component indicates whether a component is heavily connected to the content (left) or to the users (right). The IMF as providing the knowledge infrastructure for the system is located in the centre of the figure. That means that the IMF aims to provide necessary data suitable to combine both the raw *essence* data with user profile data.

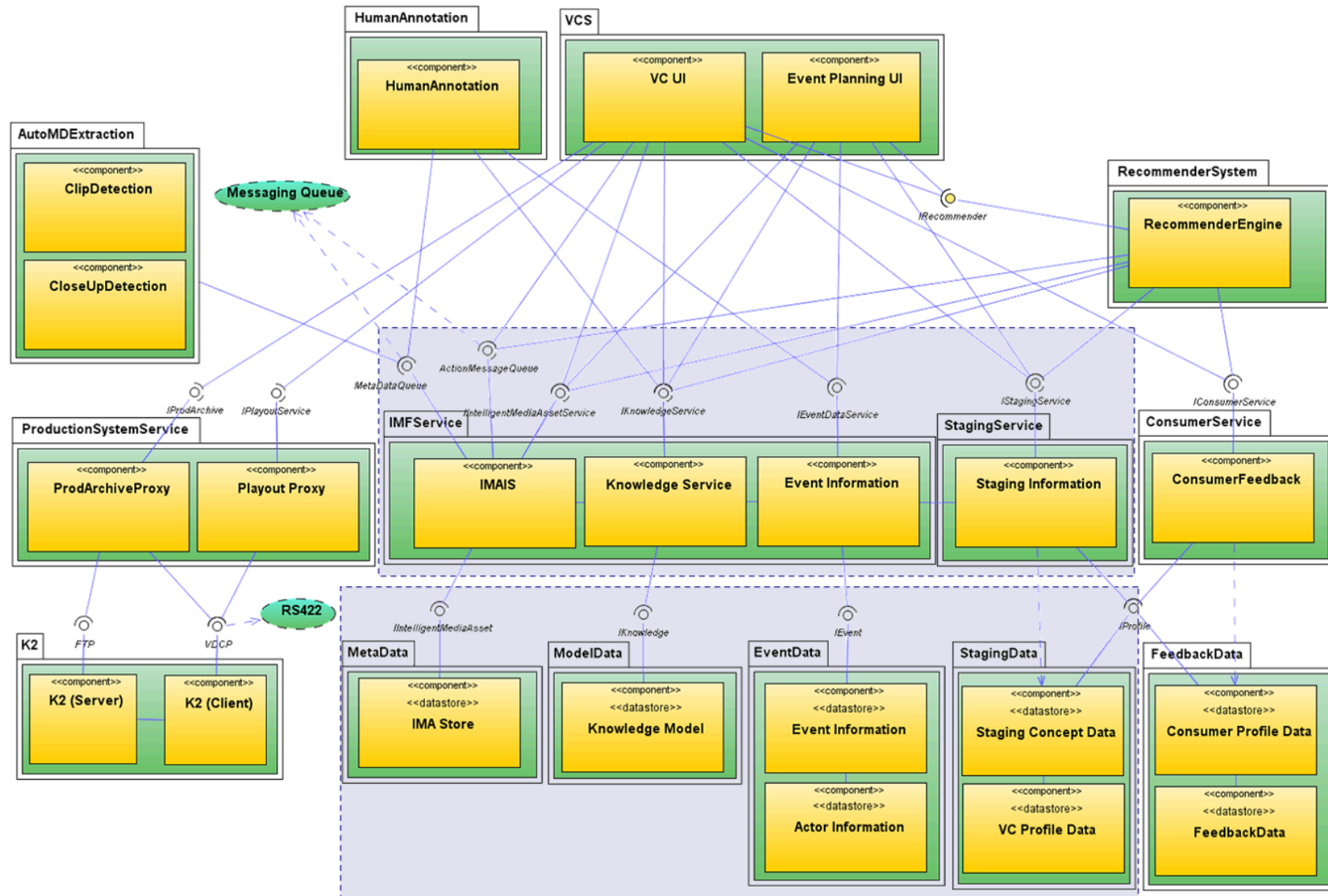


Figure 12: LIVE system components

On the top of the diagram the three visual components of the LIVE System are shown. Firstly, the Human Annotation Tool part of the Metadata Generation System, secondly the Video Conductor User Interface and thirdly, the Event planning User Interface both part of the part of the video conducting system building block.

The *Intelligent Media Asset Information System (IMAIS)* stores the Intelligent Media Assets, the knowledge model holding the controlled vocabulary and all information available about the staged event(s). Feedback data, consumer profiles as well as profiles of the professional users are managed by the recommender system. The interfaces connecting the data layer with the business and service layer represent the domain models used by the different data stores.

In the middle part of the diagram, the components holding the business logic are located. The components act as the middleware between the data and the visual components which are located at the top line of the diagram. The production system services include interfaces and adapters to functionality of the broadcasting environment. The IMF services include the IMAIS, the knowledge services and the event information service. The staging information service is listed separately in the above diagram as it is a requirement for the LIVE staging support system: According to the general work plan, this service was not implemented during the first iteration cycle of the LIVE project but will be integrated with the other IMF services in the second iteration cycle. The consumer services currently include only the consumer feedback service, but services for accessing consumer and professional user profiles would also fall in this category.

The automatic metadata extraction tool and the recommender system have a special position within that layer because they actually use services of other components. The annotation tools need to access the controlled vocabulary provided by the knowledge service. In addition, the annotation tools use the metadata queue to notify the IMF about extracted features. Similarly the recommender system needs to have access to the knowledge service for configuration purposes and for event and staging information. For calculating recommendations the recommender engine queries IMAs by using the IMA service.

5.1.2 Components of the Intelligent Media Framework

This section provides overview descriptions of the components and interfaces of the IMF. A detailed description of the service specification is available in the technical, yet non-public report [LIVE D7.2], an initial prototypical implementation is described in section 5.2 (“The First Prototype”, p. 34).

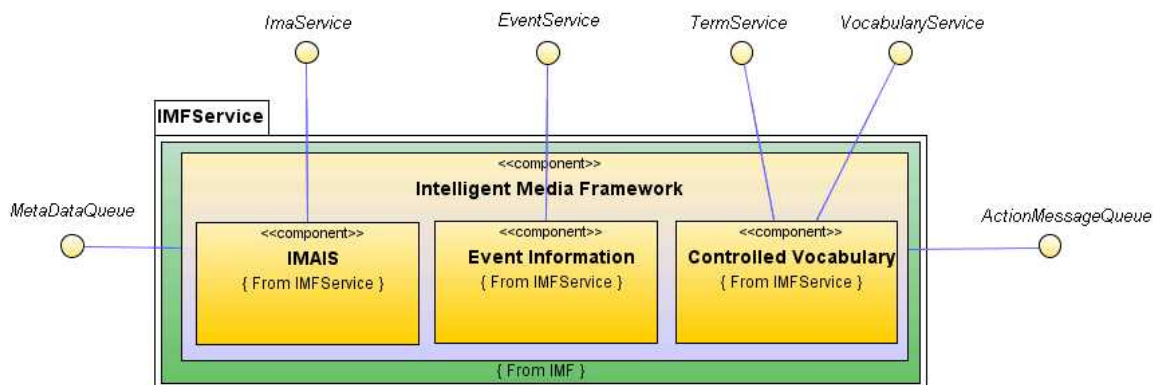


Figure 13: The components of the IMF

Figure 13 depicts the components and interfaces of the IMF in detail and provides a reference to the subsequent component descriptions.

Intelligent Media Asset Information System (IMAIS)

The Intelligent Media Asset Information System (IMAIS) deals with clip based metadata. This includes several functionalities concerning management, query and storage of IMAs according to the IMA model. To cope with real-time requirements the IMAIS implements a message brokering functionality. The mentioned functionality is exposed by service interfaces:

The IMA Service

The IMA Service which is based on the basic IMA Model (see section 4.4) provides query based access supporting query refinement to IMAs stored in the IMA store.

The MetaDataQueue and the ActionMessageQueue

These interfaces enable messaging within the IMAIS environment: Triggering information from internal and external metadata generators is sent as messages to the metadata queue. External metadata sources can be automatic annotators or the human annotation tool and internal sources are event information systems or time keeping systems. The messages are processed by the IMAIS and republished as so called action messages to all the other components of the LIVE staging support system. To provide a single access point for different generators, the MetaDataQueue accepts message based on several schemes:

- IMA Model: Item based knowledge can be sent to the metadata queue by using the IMA model
- LIVE MPEG-7 profile: This profile of MPEG-7 may be used to describe features recognised in *essences* by an automatic detection tool or a human annotator. Creator, Time/Duration and Terms used within the <semantic> element of the MPEG-7 messages are parsed by the IMAIS
- Event Model: Event Information describing the actual state of an event can be encoded using the event model
- SportsML / NewsML: The metadata queue will have support for this standard for the integration of external information systems. This feature is not part of the first prototype.

The ActionMessage Queue is realised as a messaging queue where multiple subscribers may receive messages. The ActionMessages will be available in the same Formats as the Metadata Queue. Primarily the LIVE specific schemas will be used, but it is also planned to encode action messages by using Sport ML, News ML and MPEG-7 to provide support for LIVE external systems. This can be implemented by using different queues for different formats, or by using message filters.

The information generated by external metadata extraction systems or other external sources is processed by the IMF in order to refine the incoming information with additional knowledge. The IMF then publishes the refined information again whereas the tools supporting the Editors and Video Conductors can subscribe to this information. Multiple Metadata Generators may send information to the IMF which receives and processes this information. After processing, the IMF re-publishes the processed metadata to allow multiple subscribers to process the information.

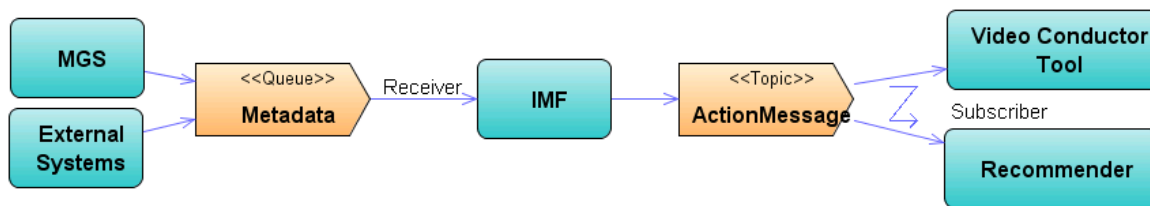


Figure 14: Real-time information chain

This approach guarantees separation of concerns while supporting the entire information chain. The Metadata generators do not need to know anything about the preferences of the editors and video conductors and can simply be tailored to their intended core functionality. Tools supporting the Editors and Video Conductors are also loosely coupled to their information sources. They do not have to care about the origin of the incoming information but are in a position to automatically react on incoming notifications by e.g. the generation of new recommendations.

Knowledge Service

The Knowledge Service maintains the knowledge base and provides access to controlled vocabularies.

To allow proper management of the controlled vocabularies, the Knowledge Service is split into three distinct services which all provide access to information based on the term model as described in 4.2 (“Term Model”, p. 20):

- The *VocabularyService* is responsible for the management of vocabularies. It is possible to store an unlimited number of vocabularies. A vocabulary can therefore be treated as a logical container for associated/related terms made available for external systems.
- The *TermService* provides a query interface for terms coming from controlled vocabularies, IDs and labels. This service is also responsible for the management of terms which can be added, updated or removed.
- The *TermContextService* allows the querying about relationships between terms: all related terms based on a given source term can be retrieved.

Event Information Service

The Event Information Service provides information about the event(s) to be staged. As an entry point, the event setting is maintained where each entry refers to a “design” concept out of the knowledge base. Each event setting also manages a list of associated events. Each event setting may have any number of events, such as the Tour de France (event setting) usually consists of a prologue (event) and at least 20 stages (events) whereas each stage belongs to the Tour de France (event setting) and the Pro Tour Series of the UCI (event setting).

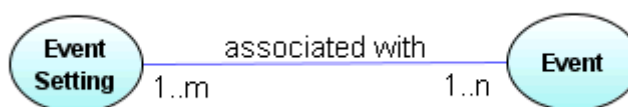


Figure 15: Relationship event - event setting

For each event the list of participants (Agent) and a list of actions (Activity) are maintained. All words in brackets refer to concepts defined in the Event Domain Model. For more information see section 4.3 (“Event Domain Model”, p. 23) of this deliverable.

5.2 The First Prototype

The first iteration cycle of the LIVE project (from January 2006 and June 2007) aimed at the delivery of demonstrators for the tools and sub-systems of the LIVE production support system and integrating these into a functional prototype. In the case of the Intelligent Media Framework a demonstrator was implemented and delivered in June 2007. We provide a short overview of the prototypical implementation of the IMF which fulfils two major functions: Firstly, it acts as a persistence layer for all the connected components. Secondly, it provides message brokering for the entire prototypical LIVE production support system.

In general, each piece of information sent to the IMF can later be retrieved by the appropriate methods of the web services available. By propagating messaging to the LIVE production support system, the IMF ensures the information flow from the metadata generators to the metadata consumers such as the recommender system. In the first prototype, message brokering is basically a forwarding of the incoming messages to the metadata consumers. The intention however is to enrich incoming messages with contextual information from the knowledge model before propagating to the metadata consumers. The IMF acts as a stateless component, therefore each incoming information is published onto an outgoing queue. Subscribers can then decide on which type of information they want to react upon.

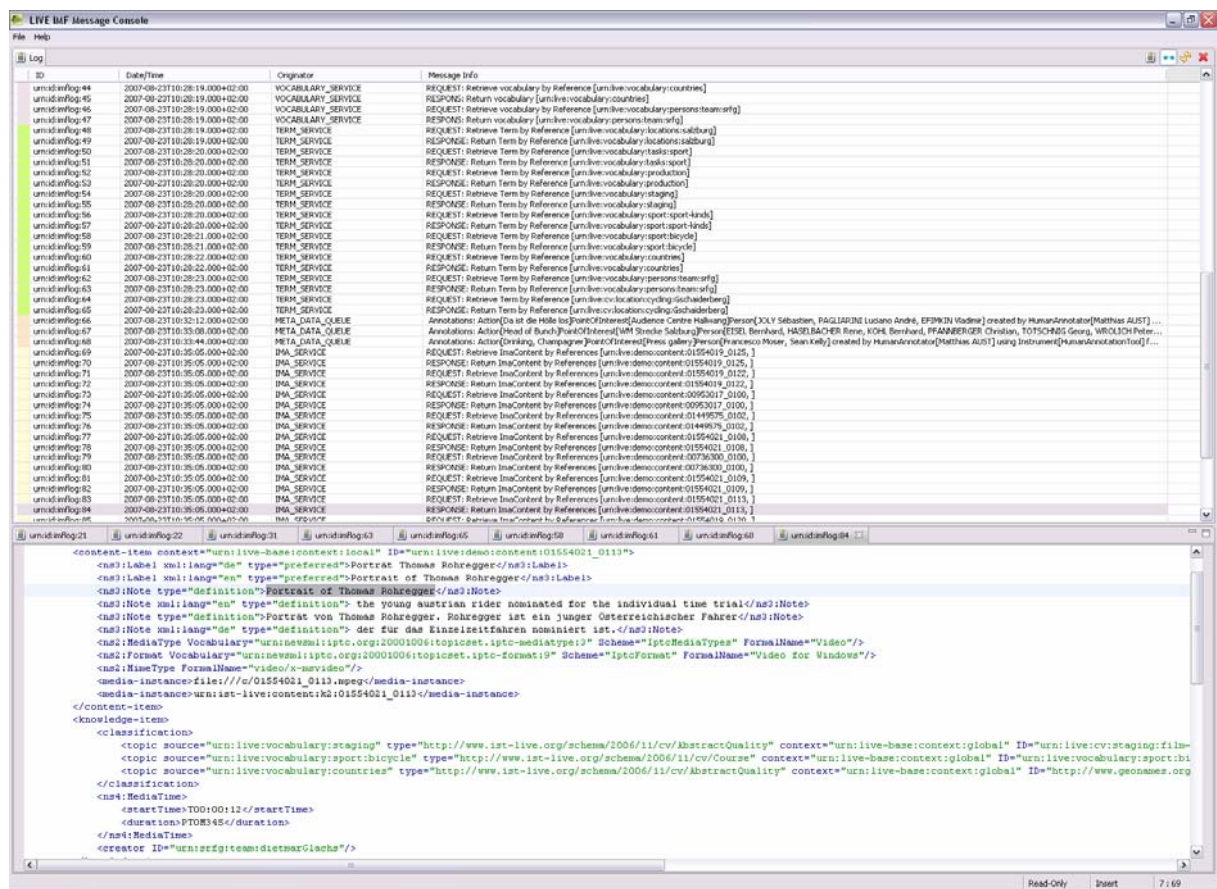


Figure 16: The IMF console

In the first phase a prototype of the IMF was implemented consisting of the following components:

- The **Knowledge Service**: This component currently provides a controlled vocabulary to the LIVE production support system. The first implementation of the provided interfaces supports storing and retrieving of vocabularies and terms.
- The **Intelligent Media Asset Information System (IMAIS)**: The aim of the IMAIS is to provide the clip based information to the connected components like the Recommender Engine or the Video Conducting User Interface respectively. The IMAIS already supports the basic IMA model as described in section 4.4. Both Content Annotation and Subject Matter Annotation, are supported.
- The **Messaging System** is also available although there is currently no message enrichment. The messages however are sent to the outgoing queue and can therefore be processed by the consumers. A so called “IMF Messaging Console” was implemented and serves as a demonstrator for the monitoring of action messages (see Figure 16).

Event Information and Staging Information are not part of the first prototype.

6 Conclusions and Future Work

This report provides a basic specification of the Intelligent Media Framework (IMF) which will be refined and extended in the course of the LIVE project. The framework acts as a middleware for the overall LIVE production support system and is based on a combination of the classical three-tier architecture with principles of service oriented architectures (SOA). The IMF provides interfaces and services for the management of the intelligent media assets in support of other components under real-time conditions (e.g. it provides the knowledge infrastructure for the *Metadata Generation System*, the *Recommender System* and the *Video Conducting System*).

In the following sections we assess our design principles according to the knowledge and framework requirements and provide an outlook of the future work

6.1 Assessment of IMF Requirements

6.1.1 Assessment according to Knowledge Requirements

In this section, the presented knowledge structure for the Intelligent Media Framework is assessed according to the knowledge requirements defined in section 3.1 (“Knowledge Requirements”, p. 17).

Make information machine processible

The knowledge model for the IMA is formalised to make information created while planning- as well as staging activities processible by machines. The model covers several levels of expressiveness: A simple term model in combination with classification descriptions establishes compatibility with external information systems. Expressing more formal knowledge about the current course of an event is possible by using the event domain model in combination with event situation descriptions. Grounding based on the DOLCE foundational ontology ensures that the different knowledge models used within the IMF and external models can be integrated in a consistent manner.

Support cross-domain interoperability

Based on the idea of a facet based design introduced by the KCO model (see [Beh2005], [LIVE D7.1]), the different knowledge models used within the IMF use parts of existing standards: vCard is used to describe Names, Addresses, Contact information, etc. SportsML is used to describe metadata about events and athletes, NewsML and MPEG-7 are used to provide metadata for the content items of IMAs. Thus it is possible to share (import and export) knowledge with other software system within and outside of the LIVE TV production domain.

Allow multiple interpretations

LIVE’s Intelligent Media Asset model allows us to attach different interpretations of different users to one and the same content item. The reflection of the different “levels of meaning” within the IMA model is specific for the LIVE domain. This representation is ensured by the definition of LIVE specific *propositional description* types: The *classification description* can

be used to express meanings on the particular level, the *event situation description* to express local level descriptions, and the *staging situation description* will be used to express interpretations of agents working on the global level of meaning. However, universal level interpretations are not planned to be formally described, but will only be referred to with terms.

As an example let us look at the universal concept “David against Goliath” which will be presented within the IMA by a term. That means that the knowledge behind that concept – an outsider wins a fight against an overwhelming favourite – is not accessible to the IMF, but only to the mental models of the Video Conductor using this term to interpret (annotate) a content item. But still the Video Conductor can search the IMA based on this term to find situations corresponding to his mental model behind the concept “David beats Goliath”.

Unleash hidden information

The set of formal statements defined within the domain models provide an easy to use language to formally describe staging situations. The controlled vocabulary based on the term model defines a list of terms, which can be used within these statements. The LIVE production support system can use these knowledge structures provided by the IMF to implement tools supporting professional users to create annotations about event and whole show: e.g. an individual identified by the human annotator can be selected from the list of participants within the human annotation tool. An action performed by this individual can be specified by selecting one of the possible tasks defined for the current kind of sport. The IMA can then interlace these bits of information in the available knowledge for the event. An agent is created by combining the specified individual with the given roles of this individual for the current event. An activity is created for the specified task and by connecting the created agent as the one performing this activity. In addition the activity can be marked as part of the current event. This example shows how only two bits of information provided by the human annotator can open the door to a wealth of hidden information by applying knowledge about the current context to the original information item. In addition the IMA model provides the necessary propositional description types to link these knowledge structures to the content item.

Make knowledge independent of content

The introduction of Information Items in the IMA model provides the counterpart of content items in the knowledge domain. To exemplify, an incident such as a crash in a race is no longer represented by an annotation of a content item, but represented by an information item which is realised by the content item. This enables first that multiple users within the LIVE staging process can conceive different interpretations about that information item and second it also allows to link different content items with one and the same information item. Additionally this knowledge structure can support the planning activities, by first creating information items for planned parts of the show and later on assigning (or recommending) content to them.

6.1.2 Assessment according to the Framework Requirements

This sub-section assesses the presented system architecture for the Intelligent Media Framework according to the framework requirements defined in section 3.2 (“Framework requirements”, p. 18).

Knowledge and information integration

Information integration is supported by the Intelligent Media Asset Information System (IM AIS) and can be separated into two aspects: Firstly, to integrate event information gathered from external information systems with the LIVE production support system. External information systems are typically provided by the broadcasters or by the organisers of the event. Secondly, to integrate annotations about observed situations created by automatic and human annotators. Both internal and external information need to be integrated with prepared information about the event, such as participants, locations, the time table, etc. Typically, information from external sources has to be mapped to the controlled vocabulary.

Management of intelligent media assets

This functionality is provided by the *IMA Store*, a component of the IM AIS. By keeping flexibility as well as scalability and performance in mind, a data model suitable to store all LIVE related data is developed. The model allows access to the knowledge base and the intelligent media assets in a traditional relational manner but also offers storage and retrieval of XML data in the provided data formats such as MPEG-7 or NewsML. Search in XML data is then performed with XQuery. Future work will extend this functionality with the possibility to store RDF graphs and query them with SPARQL.

Real-time delivery of knowledge

To meet the requirements of real-time delivery of knowledge a messaging infrastructure was designed and implemented as part of the IM AIS. The *metadata queue* is used to receive real-time information from internal and external sources. The *action message queue* supports the distribution of real-time knowledge to other components of the LIVE production support system. Information received from the metadata queue is processed by the IM AIS. This includes the alignment to the IMF knowledge models as well as the storage of knowledge in the *IMA Store*.

6.1.3 Conclusions

Initial experimentation with the first prototype supports the hypothesis that the introduction of an intelligent content model for the LIVE staging domain (the “*IMA model*”) based on the clear separation of content and knowledge conceptually fulfils the main requirements of the LIVE staging domain: The IMA model integrates information about the event, the content, the users involved in the production process and the concepts of presenting the event (“staging”) on a semantic level. It uses the descriptive schemes of international and industrial standards and thus guarantees cross-domain interoperability. By providing the necessary propositional description types to link knowledge structures to the content items, the model allows different interpretations of the same content. The model allows access to the knowledge base and the intelligent media assets in a traditional relational manner, but also offers storage and retrieval of XML data in the provided data formats. The message services developed as part of the Intelligent Media Asset Information System (IM AIS) support the real-time requirements of the LIVE production support system.

6.2 Future Work

Some open issues have to be tackled in the second iteration cycle (from July 2007 to June 2008) to serve the requirements of the field trial of the LIVE production support system. One of the main objectives of the future work will be the further improvement of the information services to become compatible with external news information systems e.g. by adopting the IPTC News Architecture (NAR)¹³.

Furthermore the knowledge model will be further extended by a model for the staging domain. Therefore results achieved in the staging analysis of the LIVE project will be further aligned to DOLCE and the Descriptions and Situations Pattern (DnS; see [Gan2003], [Mas2002]). The *Intelligent Media Asset Information System* (IMAIS) storage will be further extended to support RDF based data and the IMF will be evaluated according to real time requirements.

A future challenge includes the enrichment of partial information and the alignment of information available via the metadata queue to the current context of the event, e.g. when a human annotator annotates media related to a person, the IMF has the responsibility to check if “knowledge” about this person is already available within the context of the event. If the person is a participant in the current event, the IMF needs to align the person to the event domain model. The goal is to develop a component which is optimised for the semantics defined in the different knowledge models of the IMF.

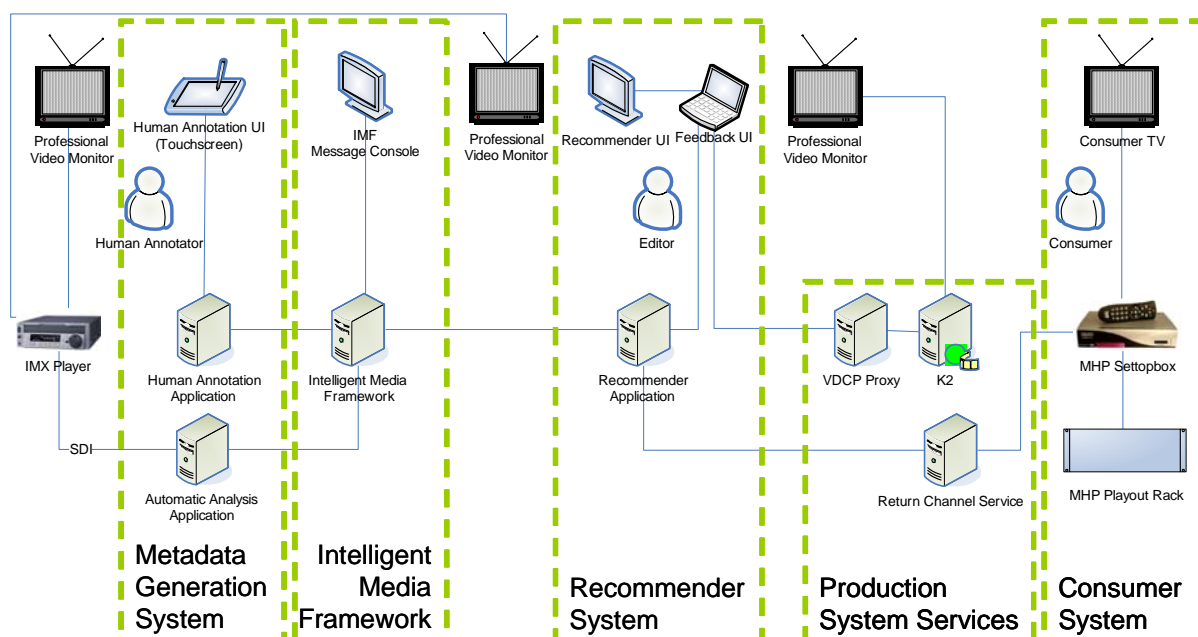


Figure 17: The LIVE production support system (prototype setup)

The delivery of the first prototype of the LIVE production support system in June 2007 served as a proof of concept not only for the intelligent content model (the “*IMA model*”) and the middleware layer for the handling of intelligent media assets (the “*Intelligent Media Framework*”) developed on top of this model in work package 7, but also for the automatic detection tools and the *Human Annotation Tool* (the “*Metadata Generation System*”) developed in work

¹³ News Architecture (NAR) for G2-Standards: <http://www.iptc.org/NAR/> - last visited: 01.08.2007

package 5 and for the “*Recommender System*” developed in work package 6. The overall system integration and a initial “*Consumer System*” were provided by work package 8 whereas the design of the technological infrastructure received fruitful input from the team of work package 4 who provided ideas for the envisaged content formats going beyond the current, linear approach to TV broadcasting of live sporting events. Figure 17 shows the setup of the first prototype of the LIVE production support system. In this setup the deployment and human interfaces of the prototype are shown in three “work spaces”: (i) the “Annotation Space” on the left side of the diagram (comprising the *Metadata Generation System*); (ii) the “Editor Space” in the centre of the diagram (comprising the *Intelligent Media Framework*, the *Recommender System* and the *Production System Services*); and (iii) the *Consumer Space* at the right side of the diagram.

The future development will be based on a revised requirements analysis, a revised system architecture and an additional design iteration cycle for the Intelligent Media Framework. The following quote from the first review in March 2007 indicates the potential of LIVE and the approach taken for the Intelligent Media Framework:

*“There are clear possibilities for exploitation of components of the systems under development as well as the entire integrated workflow system. **The real innovative part is the Intelligent Multimedia Framework** and combined with the recommender (an intelligent browser of information), it should be relevant even in conventional broadcast domain.”*

(Quote from the review report, March 2007)

7 References

- [Beh2005] Behrendt, W. et al. "Towards an Ontology-Based Distributed Architecture for Paid Content" In: A. Gomez-Perez and J. Euzenat (Eds.): *ESWC 2005*, LNCS 3532, pp. 257-271, 2005.
- [Cas2003] Cassens, J. "A Work Context Perspective on Mixed-Initiative Intelligent Systems". In: *Proceedings of the IJCAI 2003 Workshop on Mixed-Initiative Intelligent Systems*, Acapulco (2003) pp. 30-35, 2003.
- [Gan2003] Aldo Gangemi and Peter Mika "Understanding the Semantic Web through Descriptions and Situations". In: *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*, pp. 689-706, 2003
- [Gan2005] Aldo Gangemi "Ontology Design Patterns for Semantic Web Content". In: *Proceedings of the International Semantic Web Conference 2005*, pp. 262-276, 2005.
- [Hab1981] Jürgen Habermas "The Theory of Communicative Action". Trans. by Thomas McCarthy, 2 vols. Cambridge: Polity, 1984–87 (trans. from "Theorie des kommunikativen Handelns", 2 vols, Frankfurt am Main: Suhrkamp, 1981).
- [Har1968] Gilbert H. Harman "Three Levels of Meaning". *The Journal of Philosophy*, Vol. 65, No. 19, Sixty-Fifth Annual Meeting of the American Philosophical Association Eastern Division (Oct. 3, 1968), pp. 590-602, 1968.
- [LIVE D4.3] Richard Wages, Stefan Grünvogel, Jaanis Garancs. "Methods, design guidelines, workflows for online staging". Public LIVE deliverable D4.3, 2006.
- [LIVE D7.1] Georg Güntner, Wernher Behrendt, Tobias Bürger, Alberto J. Cruz, Christian Eckes, Gerhard Stanz, Rupert Westenthaler, Janez Zaletelj, Felix Zielke. "State of the Art Report IMF". Public LIVE deliverable D7.1, 2006.
- [LIVE D9.3] Heike Fischer, Oliver Lucht. "Public synopsis of the LIVE System Architecture" (2006). Public LIVE deliverable D9.3, 2006.
- [Mas2002] Claudio Masolo et al. "The WonderWeb Library of Foundational Ontologies Preliminary Report" (WonderWeb Deliverable D17), 2002. Available online: <http://wonderweb.semanticweb.org/deliverables/documents/D17.pdf> (last visited: 31.07.2007)
- [Mas2003] Claudio Masolo, Stefano Borgo, Aldo Gangemi, Nicola Guarino, Alessandro Oltramari. "WonderWeb Deliverable D18 Ontology Library, December 31, 2003. <http://wonderweb.semanticweb.org/deliverables/documents/D18.pdf> – last visited 25.06.2007
- [MKP2002] Martinez-Sanchez, J. M., Koenen, R. and Pereira, F.: MPEG-7: The generic multimedia content description standard, part 1. *IEEE MultiMedia*, 9(2): pp. 78-87, 2002.

Appendix: Index of Figures

Figure 1: Building blocks of the LIVE production support system (acc. to [LIVE D9.3])	6
Figure 2: IMF layered architecture.....	7
Figure 3: Overview of the IMF service layer	7
Figure 4: The LIVE domain	11
Figure 5: The four levels of meaning (acc. to [LIVE D4.3])	14
Figure 6: LIVE staging environment of an editor	16
Figure 7: Knowledge structure of the IMF.....	19
Figure 8: SKOS model	21
Figure 9: Term type hierarchy.....	22
Figure 10: Event domain model	23
Figure 11: IMA model.....	26
Figure 12: LIVE system components	30
Figure 13: The components of the IMF	31
Figure 14: Real-time information chain	33
Figure 15: Relationship event - event setting.....	33
Figure 16: The IMF console.....	34
Figure 17: The LIVE production support system (prototype setup)	39