A Support Framework for Automated Video and Multimedia Workflows for Production and Archive

Robert Manthey, Robert Herms, Marc Ritter, Michael Storz, and Maximilian Eibl

Technische Universität Chemnitz,
Chair Media Informatics,
Straße der Nationen 62, D-09111 Chemnitz
{robert.manthey,robert.herms,marc.ritter,
michael.storz,maximilian.eibl}@informatik.tu-chemnitz.de

Abstract. The management of the massive amount of data in video- and multimedia workflows is a hard and expensive work that requires much personnel and technical resources. Our flexible and scalable open source middleware framework offers solution approaches for the automated handling of the ingest and the workflow by an automated acquisition of all available information. By using an XML format to describe the processes, we provide an easy, fast and well-priced solution without the need for specific human skills.

Keywords: Ingest, Framework, XML, Archiving, Middleware.

1 Introduction

Today's archives of audiovisual media, especially video tapes, are still at least partly insufficiently documented. Making this media searchable is a challenging task concerning capacity and time. The more specific the metadata vocabulary is, the greater are the possibilities for information retrieval and the automation of processes [1, p.94]. A comprehensive description of the media requires the collection of as many as relevant metadata [2].

This challenge can be overcome by an early intervention into the workflow, for instance during the ingest process. Ingest means transferring media into a system including encoding and file creation. The automated ingest is a complex workflow, which requires appropriate hard- and software components. [3]

We assume that metadata concerning technical constraints of the ingest process is a benefit for the media lifecycle. This includes the used technologies as well as hard- and software. Based on this approach, media can be documented more comprehensively by enabling an increasing transparency for the end user. Furthermore, more available options concerning search requests are provided like searching for original video tape formats or video software and their requirements, for instance, devices that naturally only supports low bandwidth only would get low bandwidth results. An established workflow yields the opportunity to detect faulty equipment or to correct errors automatically in the ingested material, because the type of the video tape or the video player is known that may have caused a specific noise. [4]

S. Yamamoto (Ed.): HIMI/HCII 2013, Part III, LNCS 8018, pp. 336–341, 2013.

Nowadays, commercial systems in production and ingest often use proprietary standards, closed and fixed black boxes or are very expensive like Multicam [5] or Airspeed 5000 [6]. On the other side non-commercial systems like Ingex [7] are open source and therefore modifiable, but more focused on production processes. In contrast to that our solution allows an automated tape-ingest and the extraction of metadata to support further processing.

2 Imtecs-Framework

As a part of our research, a flexible and open source framework called Imtecs (*I*ngest *m*iddleware including extraction of metadata from *te*chnical *c*onstraint*s*) has been developed providing a collection of components in order to implement an automated ingest workflow, which in addition enables the extraction of metadata of the technical constraints for each job. Basically, the framework is extensible and can be integrated into existing workflows, such as archiving. Furthermore, it can be applied individually or in combination with an automatic content analysis as well as intellectual annotation to grant a holistic solution.

In order to adapt to such an ingest workflow, as much information as possible should be acquired. For this purpose, we define the Ingest Workflow Description Format (IWD), an XML format to describe all devices and services that are involved in the workflow as well as their properties and relationships to each other.

Using this description, the framework works as a middleware layer for an ingest workflow scenario storing the collected metadata of the technical constraints of each ingested job.

2.1 System Architecture

The Imtecs-Framework provides a collection of software components to meet the requirements to be open source, modifiable, using an extensible format as well as gathering as much information as possible. The handling of external devices and services is included in the system core. After configuring the framework, it acts as a middleware-layer to control the participating devices and services. Basically, also devices can provide any kind of services like a webservice. The classification at user level facilitates the configuration of workflows making them more transparent, especially for other people in the collaboration.

The architecture of the framework (Fig. 1) prescribes that devices are represented by Device Objects and services by Service Objects. Device and Service Classes are abstractions of devices. Each class provides metadata parameters which are set by each corresponding object. The adjusted Controller takes over the control of the workflow and arranges the scheduling of the devices and services, so that one or more audiovisual media can be are successfully ingested. The Metadata Collector gathers all the metadata of each Device and Service Objects that are involved in the ingest process of a certain medium. [4]

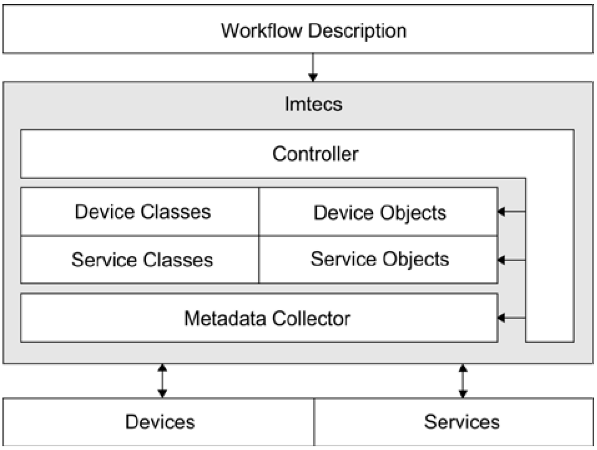

Fig. 1. Scheme of our framework

2.2 Ingest Workflow Description Format (IWD)

To describe a given workflow, the components and relations between them, we need a flexible, customizable, scalable and easy to handle format. We decided to create a format based on XML which fulfills these requirements. We separate these descriptions into several files to allow their distribution through several hosts and administrative authorities. Because of this, one file handles the configuration of relatively general settings as shown in Fig. 2. Others hold the workflow-dependent settings for the given workflows like in Fig. 3. This distribution and the possibility to use any kind of network to operate the framework, provides the opportunity and adaptivity to realize any given ingest workflow as presented in the next section, for instance.

Fig. 2 shows a configuration file from our Imtecs-framework. It contains relatively general, workflow-independent settings. For example, the amount of time that can be added to each ingest operation to be robust against overlong input materials or unpredictable timing issues. The *TestController* is used to operate the whole framework in a simulation mode with communication between the components, workflow operation, metadata acquisition and so on, but without interaction to real hardware devices or external services. After this manner, configurations of all kinds of workflows can be tested and verified, easy and fast and without any touch to the valuable material. Because of the distributed nature of the framework all locations occurs as local or remote paths, as shown in the entries *Devicelist*, Log and *PathList*. Here, they point to the descriptions of the workflow and the location of the logging file. The final destination of each ingest, the *Storage*, displays the connection to other systems pointing to different hosts with different destination paths on that hosts, to allow them to be used by different workflows or stages of workflows.

```xml
<?xml version="1.0" encoding="utf-8" ?>
<Configuration>
  <TestController>false</TestController>
  <RecordTolerance unit="minutes">10</RecordTolerance>
  <DeviceList id="1">C:\Users\validax\Imtecs\Imtecs\WorkflowDescription.iwd</DeviceList>
  <Log id="40">\\validax.informatik.tu-chemnitz.de\ingest\Log.txt</Log>
  <PathList>
    <Path id="51" type="Storage">
    \\validax-storage1.informatik.tu-chemnitz.de\validax\ValidaxIngest</Path>
    <Path id="52" type="Storage">
    \\validax-storage2.informatik.tu-chemnitz.de\validax\ValidaxIngest</Path>
    <Path id="53" type="Storage">
    \\validax-storage3.informatik.tu-chemnitz.de\validax\ValidaxIngest</Path>
    <Path id="70" type="Metadata">
    \\validax-metadata.informatik.tu-chemnitz.de\validax\ValidaxMetadata</Path>
  </PathList>
  <Source_Identification type="manual"/>
</Configuration>
```

Fig. 2. General configuration file

```xml
<?xml version="1.0" encoding="utf-8" ?>
<WorkflowDescription>
    <DeviceList>
      <List name="Robot">
        <Robot id="1" active="true" controllable="true">
          <Address type="TCP/IP">123.110.220.121:8001</Address>
          <DeviceInfo>
            <Name>L2012</Name>
            <Type>Lego</Type>
            <Interface>
              <Input>Bluetooth</Input>
            </Interface>
          </DeviceInfo>
        </Robot>
      </List>
      <List name="Camera">
        <Camera id="31" active="false" controllable="true">
      </List>
      <List name="Player">
        <Player id="62" active="true" controllable="false">
          <DeviceInfo>
            <Name>Panasonic AG-7700</Name>
            <Type>SVHS</Type>
            <Interface>
              <Output>SVideo</Output>
            </Interface>
          </DeviceInfo>
          <Workflow position="2">
            <Next type="Device">92</Next>
          </Workflow>
        </Player>
      </List>
      <List name="Converter">
        <Converter id="92" active="true" controllable="false">
          <DeviceInfo>
            <Name>Electronic-Design TBC-Light</Name>
            <Type>Time Base Corrector</Type>
            <Interface>
              <Input>SVideo</Input>
              <Output>SVideo</Output>
            </Interface>
          </DeviceInfo>
          <Workflow>
            <Previous type="Device">62</Previous>
            <Next type="Device">93</Next>
          </Workflow>
        </Converter>
        <Converter id="93" active="true" controllable="false">
          <DeviceInfo>
            <Name>Blackmagic Design Mini Converter</Name>
            <Type>Analog to SDI</Type>
            <Interface>
              <Input>SVideo</Input>
              <Output>SDI</Output>
            </Interface>
          </DeviceInfo>
          <Workflow>
            <Previous type="Device">92</Previous>
            <Next type="Device">122</Next>
          </Workflow>
        </Converter>
      </List>
      <List name="CaptureDevice">
        <CaptureDevice id="122" active="true" controllable="true">
          <Address type="HTTP">123.110.220.66:8004</Address>
          <DeviceInfo>
            <Name>Telestream Pipeline HD</Name>
            <Type>HD Hardware-Network-Encoder</Type>
            <Interface>
              <Input>SDI</Input>
              <Output>Ethernet</Output>
            </Interface>
          </DeviceInfo>
          <Workflow>
            <Previous type="Device">93</Previous>
            <Next type="Storage">53</Next>
          </Workflow>
        </CaptureDevice>
      </List>
    </DeviceList>
    <ServiceList>
</WorkflowDescription>
```

Fig. 3. XML description of a workflow

The sample in Fig. 3 shows the description of one of our workflows used for the automated ingest of video tapes. The *WorkflowDescription* is divided into two parts the *DeviceList* and the *ServiceList*. The *DeviceList* contains information to the devices and their part of the workflow. In a similar way a *ServiceList* is organized and used. They contain for instance lists of groups of similar devices that are uniquely identified by the name of the group. Each device also has a unique id to be identified within the workflow. The attribute *Active* enables or disables the device. The term Controllable describes the capability to control and receive information on-demand in order to be stored as metadata. If the device is uncontrollable the metadata must be predefined. Besides, the name, the type of the device and information about the connectivity are stored. The *ID*s of the previous and subsequent devices complete the description.

As shown in the example, the robot-device is a part of the workflow, because it changes the tapes, but it is not connected to the chain of devices operating the ingest itself to get the content of the tapes.

3 Application in the Field of Digital Archiving

We use the proposed framework within a complete automated workflow from ingest over analyzation and distribution to storing operations. The ingest of our application domain comprises a big amount of VHS, DV and Betacam tapes. About one thousand hours of content combined with associated technical constrains and metadata have already been processed.

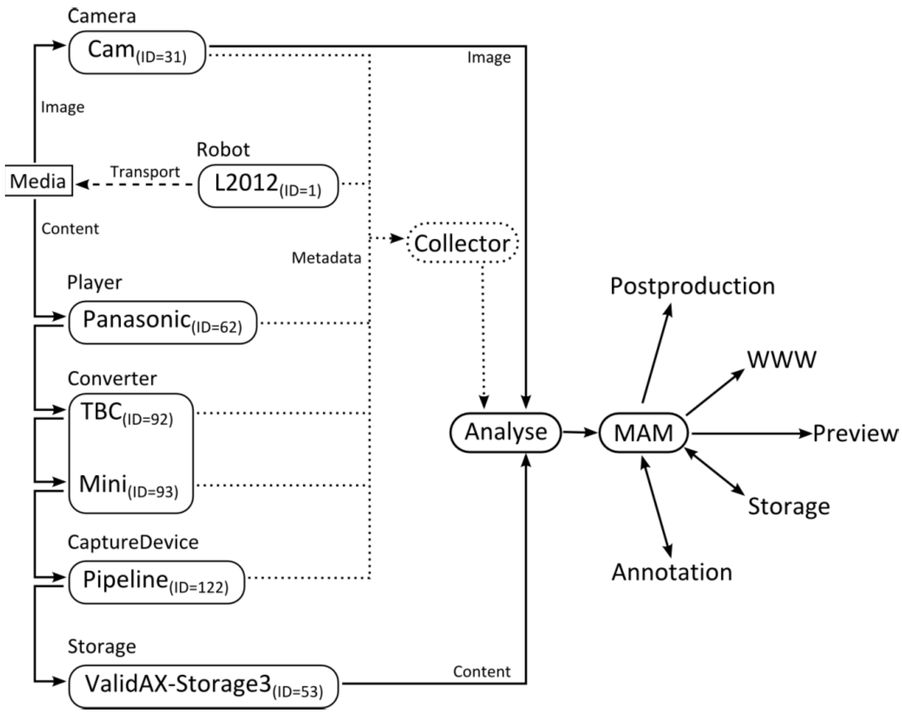

Fig. 4. Scheme of the workflow and its application inside our media management process

The left side of Fig. 4 shows the workflow described in the XML of Fig. 3. The dotted line represents the flow of the metadata extracted during the ingest, the collection and transfer to our analysis system. Together with the content of the media and the images from the envelop of the media, a comprehensive analyzation is realized.

The results are passed to our Media Asset Management System (MAM) and prepared to support subsequent steps like postproduction, distribution to Websites or to store in archiving systems.

4 Future Work

The results in this paper indicate that our framework can be used for automated pre-processing and distribution of the audiovisual media. Since the description of the ingest-workflow is separated, the applicability of the framework appears very flexible in creating arbitrary even non-linear processing chains that can be interactively altered during run-time.

The next steps include investigations of the human-machine interface concerning monitoring and controlling of a workflow with the intention to distinguish between professional and non-professional users by certain characteristics and parameters. At last, there is also a need for the examination of the deployment to mobile applications.

Acknowledgements. This work was partially accomplished within the project ValidAX – Validation of the AMOPA and XTRIEVAL framework (Project VIP0044), funded by the Federal Ministry of Education and Research (Bundesministerium für Wissenschaft und Forschung), Germany.

References

1. Hercher, J., Mitzscherlich, A., Sack, H.: Bestandsanalyse, Metadaten und Systematisierung. Digitalisierungsfibel, Leitfadenfüraudiovisuelle Archive. transfermedia, Potsdam, pp. 82–101 (2011)
2. Airola, D., Boch, L., Dimino, G.: Automated Ingestion of Audiovisual Content (2003), http://www.broadcastpapers.com/whitepapers/IBCRAIAutoIngestAVContent.pdf (February 15, 2013)
3. Borgotallo, R., Boch, L., Messina, A.: Automated Industrial Digitizationof Betacam tapes - with MXF generation andvalidation (2011), http://tech.ebu.ch/docs/techreview/trev_2011-Q4_betacam_digitiztion_borgotallo.pdf (July 10, 2012)
4. Herms, R., Manthey, R.,Eibl, M.: A Framework For Media Ingestion- Adding Data About Technical Constraints. (2012), http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-103912 (January 15, 2013)
5. EVS Broadcast Equipment SA: Multicam - Version 11.01 – Operating Manual (2012), http://www.evs.tv/sites/default/files/download_area/package/compatible_11.01.77_addon/Multicam%2011.01.77/MULTICAM_operationman_11.01_ENG_121221_web.pdf (January 21, 2013)
6. Avid Technology Inc.: AirSpeed 5000 Datenblatt (2012), http://www.avid.com/static/resources/common/documents/datasheets/airspeed_5000/AirSpeed_5000_ds_A4_sec_de.pdf (February 5, 2013)
7. Tudor, P.N., McKinnell, J.S., Pinks, N.P., Casey, S.: Developments in Automated Tapeless Production for Multi-Camera Programmes (2010), http://www.bbc.co.uk/rd/publications/whitepaper186.shtml (February 10, 2013)