Multi-Step Media Adaptatio: Implementation of a Knowledge-Based Engine

Peter Soetens
VRT
Brussels, Belgium
peter.soetens@vrt.be

Matthias De Geyter
VRT
Brussels, Belgium
matthias.degeyter@vrt.be

ABSTRACT

Continuing changes in the domains of consumer devices and multimedia formats demand for a new approach to media adaptation. The publication of customized content on a device requires an automatic adaptation engine that takes into account the specifications of both the device and the material to be published. These specifications can be expressed using a single domain ontology that describes the concepts of the media adaptation domain. In this document, we provide insight into the implementation of an adaptation engine that exploits this domain knowledge. We explain how this engine, through the use of description matching and Semantic Web Services, composes a chain of adaptation services which will alter the original content to the needs of the target device.

Categories and Subject Descriptors

J.7 [**Computers in Other Systems**]: [Publishing]; I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods; H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems—*Evaluation/Methodology*

General Terms

Design, Languages, Experimentation, Algorithm

Keywords

Multimedia, content adaptation, semantic web, services, device independence, standards, OWL

1. INTRODUCTION

A broad range of consumer devices and media formats makes it difficult to publish content on such a device. To automatically adapt content for these target platforms, a description language and an adaptation engine is needed. Such a description language will need to be extensible to cover the evolutions in the media landscape. New types of devices and formats are continually introduced so it is not sufficient to agree on a certain standard to describe content and devices [2]. An adaptation engine on the other side, will also have to cope with these changes. Not only does it have to be able to understand new descriptions but it also has to integrate and use new adaptation components that support the new formats and devices. These components can be regarded as the building blocks for processing media. Each component performs a basic operation, but by chaining them, complex adaptations become possible. Ideally, the functionality of these software components is described using the same description language as the devices and the content. In previous work, we proposed such a description language [4] in OWL using a domain ontology that describes the concepts in the media domain. We also proposed an architecture for an adaptation engine in [3] that uses this domain ontology to perform content adaptation. We present in this document the implementation of such an adaptation architecture and evaluate the performance.

WWW 2005, May 10–14, 2005, Chiba, Japan.
ACM 1-59593-051-5/05/0005.

2. MULTI-STEP MEDIA ADAPTATION WITH SEMANTICS

Our approach is based on a single ontology describing all relevant concepts of the media adaptation domain. This shared vocabulary defines the description language that is used to describe *target platforms, media presentations* and *adaptation components.* It may describe for example the capabilities of a PDA, details about a video presentation and an adaptation component transforming the video presentation to a lower resolution. The ontology is implemented using OWL and modelled after elements from both MPEG-21 and UAProf. Such a representation in OWL allows us to annotate the target platforms using the CC/PP framework, describing the client's hardware, software and browser capabilities with concepts from the domain ontology. The media presentation description is specified in OWL, but may be integrated with a format like MPEG-7. Finally, the adaptation components in our framework are Web Services and hence their functionality can be described (with concepts from the domain ontology) using OWL-S in terms of Inputs, Outputs, Preconditions and Results (IOPR). With this approach we also introduce the flexibility and distributed nature of Web Services in our architecture.

Next to this description language we have designed an adaptation engine based on a Service Registry, a matching process and an adaptation strategy (see fig. 1). The purpose of the engine is to transform given content to a desired platform based on their respective descriptions and the available services. The Service Registry, where available services need to register or sign off, is hierarchically structured based on the domain ontology and the services' IOPR descriptions. The registry allows us to increase performance when searching for a specific adaptation service. The adaptation process is performed in three phases : *matching* of the

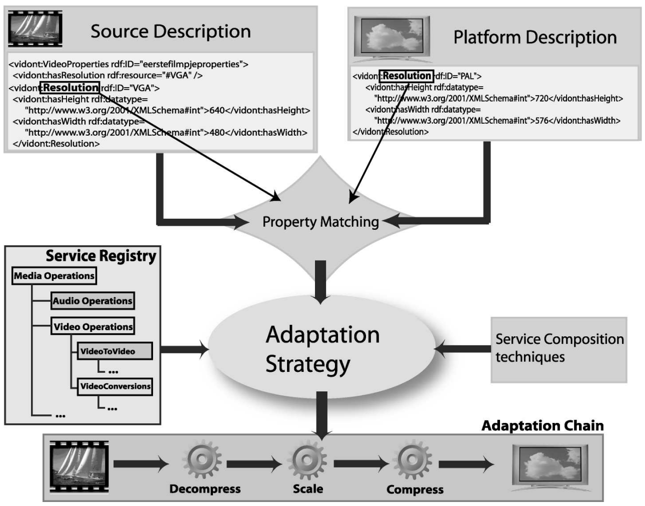

Figure 1: An overall view of the design of our adaptation engine.

platform and content descriptions, *composition* of the adaptation chain and execution of this chain. The (property) matching process of the descriptions is needed to establish the current state of the media content and the goal state (i.e. what the target platform can handle). This process involves comparing the concepts that describe the source document against similar concepts describing the platform (for example a video's resolution against the resolution of a PDA screen). The result of this process is then used in the adaptation strategy that composes the adaptation chain. This strategy is responsible for selecting a chain of services that will generate an optimal result for the given target platform.

3. IMPLEMENTATION OF AN ADAPTATION ENGINE

For the work on a news distribution system for Belgian broadcaster VRT, we have modelled three target platforms (PDA, PC and TV with set-top box) and several adaptation services for video (decode, scale, encode, deinterlace, convert to audio,...). We implemented a system that adapts video material from the broadcaster based on information about the connecting users' platform and the available services in the network. The adaptation engine first compares the platform and content description and generates a current and a goal state as defined in our architecture. The implementation of this property matcher is based on a reasoner that detects possible equivalence or subsumption relations between concepts describing both parties. Based on this result, an adaptation strategy will try to generate the best chain of services to reach the goal state. Searching services for this chain is performed in the Service Registry that provides functionality to quickly retrieve a service based on it's IOPR-descriptions. If a service has unfulfilled preconditions, for example a scaling service requires the video format to be YUV, the strategy will try to find an appropriate service that fulfills this request. Each time a service is selected, the current state description is changed by applying the services' results definition to the current state. The algorithm will iterate over the current state description until it matches the goal state. It is quite possible that multiple services are found to perform a certain adaptation : in this case the algorithm will evaluate the result definitions of each service against the goal state and select the best result. The description of this result is done similar to the SWRL-formalism [1] and may specify an Object- or DataProperty value (in case of a DataProperty this may also be a mathematical expression). Below is an Object- and a DataProperty example in a simplified form (*subject,property,value*):

- (#Video,#hasVideoFormat,#MPEG-2)
- (#BitRate,#hasBitRateValue,#BitRate/90)

This first implementation successfully adapts video material for each platform (changing encoding, resolution, framerate,...) and is even able to convert the video material to audio or a set of keyframes if bandwidth constraints are too tight. Our engine manages to convert the source document to the client platform's needs as efficient as possible; i.e. by chaining a minimum amount of adaptation components. This was verified experimentally by comparing the resulting chain to the optimal one –manually created by taking into account all services and their IOPRs– for a large number of different media and platform descriptions. In all cases, both chains were equal. So far, our engine only works in the narrow application field of video. Ongoing work involves integrating a STRIPS-like planning methodology to achieve a more generic adaptation engine which will allow us to generate more complex adaptation plans. The difficulty here lies within the ability to successfully transfer our domain knowledge to a representation used by an available planner.

4. CONCLUSION

A domain ontology provides the semantics and the flexibility needed to describe the dynamic environment of multimedia presentations, adaptation steps and target platforms. When performing media adaptation, an engine will need to process this information and select the appropriate adaptation steps. We have built such an adaptation engine that effectively uses semantically rich descriptions to compose an optimal chain of adaptation services. Experiments have shown that this algorithm generates the correct chain for many different media and platform descriptions.

5. ACKNOWLEDGEMENTS

This work is the result of a joint collaboration between VRT, IMEC, Vrije Universiteit Brussel and Universiteit Gent, funded by the Flemish government.

6. REFERENCES

[1] I. Horrocks, P. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean. *SWRL : A Semantic Web Rule Language Combining OWL and RuleML*. http://www.daml.org/2003/11/swrl/.

[2] ISO/IEC. *Information technology – Multimedia framework (MPEG-21) – ISO/IEC 21000*, 2003.

[3] P. Soetens and M. D. Geyter. Applying domain knowledge to multi-step media adaptation based on semantic web services. In *WIAMIS 2005*, 2005.

[4] P. Soetens, M. D. Geyter, and S. Decneut. Multi-step media adaptation with semantic web services. In *Third International Semantic Web Conference (ISWC 2004)*, 2004.