Democratic and Popular Republic of Algeria
Ministry of higher education and scientific research
University of Ferhat Abbes Setif 1
Department of Computer Science

Profile based multimedia documents adaptation

A novel approach for adaptation using an abstract document model

By

BELHADAD YEHYA

For the fulfilment of the requirements for the degree of

Doctor (LMD)

In computer Science
Option: Intelligent and communicating information systems SIIC
Presented publicly: 03/11/2018

Thesis committee:

President	**ALIOUAT Makhlouf**	**PROF, UFA Sétif 1**
Supervisor	**REFOUFI Allaoua**	**MCA. UFA Sétif 1**
Co-supervisor	**ROOSE Philippe**	**MCA. IUT Bayonne, FR**
Examiners	**GHOUALMI-ZINE Nacira**	**PROF, Université Annaba**
	ALTI Adel	**MCA. UFA Sétif 1**

2018

Democratic and Popular Republic of Algeria
Ministry of higher education and scientific research
University of Ferhat Abbes Setif 1
Department of Computer Science

Profile based multimedia documents adaptation

A novel approach for adaptation using an abstract document model

By

BELHADAD YEHYA

For the fulfilment of the requirements for the degree of

Doctor (LMD)

In computer Science
Option: Intelligent and communicating information systems SIIC
Presented publicly: 03/11/2018

Thesis committee:

President	**ALIOUAT Makhlouf**	**PROF, UFA Sétif 1**
Supervisor	**REFOUFI Allaoua**	**MCA. UFA Sétif 1**
Co-supervisor	**ROOSE Philippe**	**MCA. IUT Bayonne, FR**
Examiners	**GHOUALMI-ZINE Nacira**	**PROF, Université Annaba**
	ALTI Adel	**MCA. UFA Sétif 1**

2018

بسم الله الرحمن الرحيم

Abstract

The development and innovation in creating new emerging technologies created heterogeneity when accessing the internet nowadays. This heterogeneity induced by both hardware as well as software led multimedia documents to become unstructured and difficult to use. Therefore a growing need for adaptation is raised to adjust existing documents to different states and profiles. This thesis deals with profile based multimedia documents adaptation.

This thesis consists of four major parts : the first part outlines a generic survey for recent multimedia document adaptation. The second part, which is our first contribution, is a novel approach where we demonstrate how two parallel versions of a document can help resolve the complex adaptation issue, and how to create a facet for the web in order to allow seamless adaptation and continuous evolution while maintaining adaptability of both existing documents and new documents for emerging technologies.

The second contribution of this thesis is the abstract specification of multimedia documents for profile based adaptation, using an abstract document as a substitute for the original one. In the proposed abstract document the processing model helps to achieve the separation between specification and content, while creating two parallel documents. The abstract document model called XMS (XML Multimedia specification) allows to pass only the abstract version of a document to the processing engine and generates the adapted version from the original one using an adapted abstract document. A simple minimal profile is proposed to help make the concept clear. The third major contribution is the spatial specification and manipulation paradigm used in XMS, which is based on a composite spatial relation using a combination of a qualitative distance, a qualitative direction and a qualitative topology in order to specify fairly a two dimensional multimedia document. The paradigm used allows spatial manipulations on the whole spatial specification of a document. The manipulation engine is implemented as a PROLOG based inference system. An abstract minimal web page layout sample is used to exhibit the complex results of spatial manipulation algorithms along with a detailed comparison of XMS for adaptation with other adaptation solutions.

Résumé

Tant pour le développement et l'innovation dans la création de nouvelles technologies émergentes, cela à créer une hétérogénéité lors de l'accès à Internet. Cette hétérogénéité induite à la fois par le matériel et les logiciels a eu pour conséquence une déstructuration des documents multimédias qui sont devenus difficiles à utiliser. D'où un besoin croissant de l'adaptation est soulevé pour adapter les documents existants à différents états et profils.

Cette thèse porte sur l'adaptation de documents multimédias basés sur des profils et comprend quatre parties principales : la première partie présente une étude des travaux récents sur l'adaptation de documents multimédias. La deuxième partie, qui est notre première contribution présente une nouvelle approche où on montre comment deux versions parallèles d'un document peuvent aider à résoudre le problème complexe de l'adaptation et comment on peut créer une facette pour le web afin de permettre une adaptation et une évolution continues tout en maintenant l'adaptabilité des documents existants et des nouveaux documents pour des nouvelles technologies émergentes.

La deuxième contribution de cette thèse porte sur la spécification abstraite des documents multimédias pour une adaptation basée sur des profils, en utilisant un document abstrait comme un substitut à l'original. Dans le document abstrait proposé, le modèle de traitement permet à réaliser la séparation entre spécification et contenu, tout en créant deux documents parallèles. Le modèle de document abstrait appelé XMS (spécification multimédia en XML) permet de transmettre uniquement la version abstraite d'un document au moteur de traitement et génère la version adaptée à partir de la version originale en utilisant un document abstrait adapté. Un profil minimal simple est proposé pour clarifier le concept. La troisième contribution majeure est le paradigme de spécification spatiale et de manipulation utilisé dans XMS, qui est basé sur une relation spatiale composite utilisant une combinaison d'une distance qualitative, d'une direction qualitative et d'une topologie qualitative pour spécifier un document multimédia bidimensionnel. Le paradigme utilisé permet des manipulations spatiales sur toute la spécification spatiale d'un document. Le moteur de manipulation est implémenté en tant que système d'inférence écrit en PROLOG. Un échantillon de mise en page de page web abstrait minimal est utilisé pour présenter les résultats complexes d'algorithmes de manipulation spatiale ainsi qu'une comparaison détaillée de XMS pour l'adaptation avec d'autres solutions d'adaptation.

Dedication and acknowledgement

الحَمدُ لِلَّهِ

الَّذي بِنعمَتِهِ تَتِمُ الصَالِحَات

To anyone who helped in making this thesis come to life
To my parents
To you my mother for your prayers
To you my father for your encouragements
To my sisters, their husbands, nieces and nephews for standing by my side
To you my love for your inspiration

I dedicate this humble work of science

AUTHOR'S DECLARATION

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes. Except where indicated by specific reference in the text, the work is the candidate's own work. In case of a missed referencing, the author of this dissertation takes no credit for the missed references and declare that it was only missed and not an act of plagiarism and in such case the author declare to do his best to make any suggested correction if the regulations allows such modifications. Part of this dissertation is based on a published work by the author, chapter 4 and chapter 5 are adapted by permission from Springer Nature Multimedia tools and applications journal. "**Spatial reasoning about multimedia document for a profile based adaptation, Combining distances, directions and topologies**" **Belhadad, Y., Refoufi, A. & Roose, P. Multimed Tools Appl** (2018) 77: 30437. "**https://doi.org/10.1007/s11042-018-6080-8**" under the license number: **4356010499482**. Any views expressed in the dissertation are those of the author.

SIGNED: .... DATE: 03/11/2018

PUBLICATIONS RELATED TO THIS THESIS

- Belhadad, Y., Refoufi, A. & Roose, P.

  "Spatial reasoning about multimedia document for a profile based adaptation, Combining distances, directions and topologies"

  Multimed Tools Appl (2018) 77: 30437. https://doi.org/10.1007/s11042-018-6080-8

TABLE OF CONTENTS

Page

List of Tables

List of Figures

CHAPTER 1

INTRODUCTION

Over using multimedia has become a vital part of our lifestyle nowadays, using all kind of technologies to access news, information and everything related to this age of technological advancement. Using smart-phones, smart TVs, VR technologies is as usual to people as drinking coffee, some technologies were not even dreamt about before the 90s. The diversity of these technologies makes it hard to allow every user to get a full satisfaction of these technologies while accessing multimedia content whether in form of videos on YouTube channels, on movies streaming sites, or on TVs, laptops, phones etc. The vitality of accessibility of multimedia becomes even narrower as new emerging technologies raise to the end consumers.

These advances in the hardware industry, and the slow improvements on the software counterpart create a challenging environment for software developers to keep the race up. A new challenge on these developers is raised, adapting every single media to different use cases becomes a must. Adaptation to different devices, users, contexts started by the appearance of smart-phones, the first idea was to create a new stream of authoring for authors to create multiple versions for both computers and smart-phones. Later on, it was found that not only smart-phones require adaptation. With mobility we find ourselves in different situations from low battery to high contrast in broad day light to driving and working while accessing multimedia on a regular basis.

One can even think that it was not only with the appearance of smart-phones that the adaptation was needed. However, even before smart-phones we already had different standards for different operating systems, for different software applications and for different media types and quality. Accessibility was limited before smart-phones and it continues to become more limited with emerging new technologies, keeping up with new technologies by authors and content creators becomes unbearable and costly. The goal of this thesis is to propose a way out to reach the universal multimedia access (UMA [3][4][5]). Based on profiles describing devices, user preferences or contextual information, the goal is to use inference systems to adapt multimedia documents to difference scenarios.

The adaptation became a vital need for UMA with the appearance of mobile devices [6][7][8][9][10], having small screen created limited usability of the web content in general [8, 9, 11]. Some studies were interested in automatic adaptation to eliminate the need to create two parallel versions of websites and contents, few researches tackle the problem of universal accessibility and most of the existing studies focus on creating a replacement technology for standard not addressing the actual standards for improvement. Other studies focuses on providing a static work-flow for creating and publishing multimedia.

Other than mobile devices and smart-phones, new technologies are used today. With new emerging technologies, the need for more generic approaches for adaptation in order to allow user maximum accessibility on content. Some studies tackle this problem by trying to create novel technologies, mostly using space-time specification and manipulation as in SMIL [11]. Sooner or later, the new technologies or standards fail to provide what it was meant for as in the case of SMIL it was discontinued by W3C.

Some relevant studies tackle this using innovative document processing model as in Madeus [12], XADAPTOR [13] based on existing technologies such as HTML or even SMIL. However, we always end up in the need for a more generic approaches.

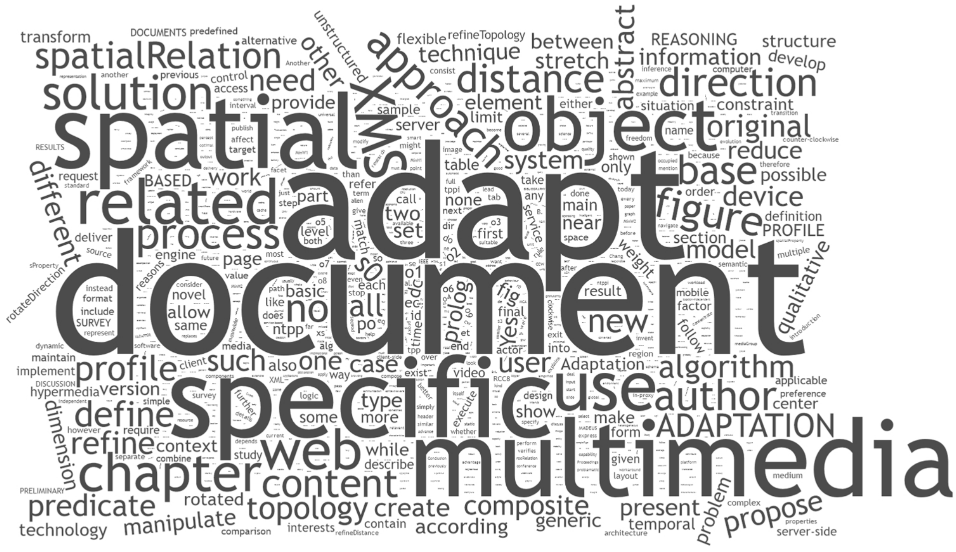
This thesis is organised as follows:

- Surveys the existing solutions and compare different recent studies: in chapter two (2) we give general information about multimedia adaptation, survey some relevant studies. Categories for adaptation are given, we also propose some factors for comparing different adaptation solutions. A comparison of existing studies and solutions is given.

- Discusses why adaptation is needed and possible ways to tackle the adaptation problem: in chapter three (3) we focus on showing reasons throughout the evolution of web on software and hardware levels that led to the adaptation necessity. A hypothetical facet is proposed as a way to eliminate the growing need for adaptation in the future.

- Proposes a practical way to implement the facet proposed in chapter three: Using an abstract document model called XMS to form a facet for the web of multimedia documents in general, in chapter four (4) an abstract document model XMS and an abstract document processing model is proposed for the profile based adaptation. A significant contribution of a novel approach for spatial profile based adaptation is proposed, a composite spatial relation using qualitative spatial distances, directions and topologies is used to manipulate the abstract structure of a multimedia document based on a profile.

- Presents preliminary results of the proposed approach and manipulation algorithms: based on an abstract layout of a sample web page, preliminary results of spatial manipulation algorithms are shown in chapter five (5). A detailed comparison with existing and similar approaches and a discussion are provided.



Begin

Chapter 2;

CHAPTER 2

MULTIMEDIA DOCUMENTS ADAPTATION SURVEY

Recently the increasing use of smart and mobile devices including not limited to: smart phones, tablets, smart TVs etc. and the continuous development of such devices has led to the diversity of constraints in playing and executing multimedia documents such as video, web multimedia, images, sounds, text, virtual reality applications etc. Therefore, constraining the accessibility of information in general, constraints that come from different forms of file representations, heterogeneous capabilities of devices and multiple operating systems, various content providers and multiple available standards that specify the process of authoring and publishing such documents. All these factors led in the last years to the necessity of adapting multimedia documents to different profiles and contexts because it is impossible to create and publish documents that are compatible with all current and future devices, and suitable to all possible situations, heterogeneous software environments and possible contexts.

This chapter gives a general presentation about multimedia documents adaptation, surveys some relevant studies in the field of multimedia document adaptation. This chapter is organized as follows, section one details the multimedia document architecture specification. Section two categorises adaptation according to what dimension should be adapted. Section three categorises adaptation according to where adaptation can be executed and survey some related recent studies, and section four compares and discusses some relevant recent papers surveyed in previous section. Finally, the chapter is concluded.

2.1 Multimedia document architecture specification

As defined in (definitions 2.1,2.2, 2.3) multimedia documents are documents that contain a composition of one or multiple media elements within it. These documents come in different shapes with different qualities and specifications. In this section, we present some basic knowledge about multimedia documents architecture specification.

Definition 2.1. According to Ellis [14] multimedia is defined as “a computer-based product that enhances the communication of information by combining two or more of the following: text, graphic art, sound, animation, video, or interactivity.”

Definition 2.2. According to Li et al. [15], authors define the term ‘multimedia’ as “a name given to a collection of data of multiple types, which include not only “traditional multimedia” such as images and videos, but also emerging media such as 3D graphics (like VRML objects) and Web animations (like Flash animations)”.

Definition 2.3. Li et al. [15] also defines a Multimedia Document as “a natural extension of a conventional textual document in the multimedia area. It is defined as a digital document that is composed of one or multiple media elements of different types (text, image, video, etc.) as a logically coherent unit. A multimedia document can be a single picture or a single MPEG video file, but more often it is a complicated document such as a Web page consisting of both text and images”.

Different specifications and standards define how multimedia documents are structured, specified, and stored, Such as the well-known MPEG [16, 17] standards, SMIL [11] or XHTML [18]. On another hand, anyone can design new formats for multimedia documents based on their own needs. The specification of the architecture of multimedia documents is based mainly on three dimensions that are temporal, spatial, hypermedia.

The specification of multimedia documents can be implicit in the definition of the document itself; in other cases, the specification is extracted or generated from the document. Another possibility for this is to make the authors express the specification explicitly, in the document or in a separate file.

Definition 2.4. In multimedia document specification, a document D is defined as: $''D{=}\langle O,S\rangle$ where O a set of N media objects and S a set of M specification$''$. Each specification concerns one of the dimensions of the document; which can be either spatial, temporal, hypermedia or logical dimensions [19].

$O = \{O_i | 0 < i \leq N, i$ and $N \in Z\}$ a set of N media objects O_i, an object can be another multimedia document in complex documents such as in web or simply a basic media element.

$S = \{S_k | 0 < k \leq M, k$ and $M \in Z\}$ a set of M specification, each one is a couple of two sets $S_k = \langle I,R\rangle$ where:

- $I = \{i \in Z | O_i \in O, 0 < i \leq N\}$ the set of identifiers of objects concerned in the specification, it may or may not concern all the object in O.
- $R = \{R_l | R_l = \langle i,j,C_n\rangle, i,j \in I, C_n \in C\}$ is the set of binary relations between these objects. C is a set of constraints that are specified usually using logic based on the dimension of the specification such as (RCC8, Qualitative distances etc.).
- Usually, the relations in R are exclusive, so if a constraint is not met, the specification is not either. Therefore, we get another different composition rather than the original intention of the author.

2.1.1 Temporal specification

The temporal specification of multimedia documents defines the temporal constraints between the media objects, these constraints are to be respected in the organization of the final document.

All kinds of temporal constraints may be as basic as 'at the same time as', 'before', 'after', 'on that particular time' or more complex including more than two objects. In the last case, the constraints might be decomposed between the objects into binary relations, this depends on the logic of the specifications.

Temporal specification [1, 20] can be done using any logic that can express temporal constraints and relations, the quality and the expressiveness of the specification then is based on the expressiveness of the used logic. There are many possibilities when it comes to describing temporal constraints.

The used logic also allows us to use its algebra to express or refine complex constraints or find new possible ways to express the same constraint, also to deduce implicit constraints. The richer the logic and its algebra are the richer the specification, we can cite as an example Allen interval logic [20] and Allen interval calculus [1].

2.1.2 Spatial specification

As in temporal specification, the spatial specification defines the spatial constraints between the media objects. Doing so, we define the spatial layout of the multimedia document.

The temporal constraints are linear meanwhile; spatial constraints can define a minimum of two-dimensional space, can be extended for virtual reality applications and content where the specification is defined in a three-dimensional space [21] [22] [23].

Same as temporal specification, in order to define spatial constraints [1], we need to choose a mathematical logic that allows the specification of the spatial constraints. Such logic would also allow us to use its algebra so we deduce other constraints or refine the existing one.

There is a version of Allen interval logic called Allen 2D [1] which consists of two-dimensional spatial relations, where every object is considered a two intervals (horizontal and vertical axis). However, Allen 2D provides a combination of a 169 relations, which is much to be processed, based on the thirteenth temporal relations. There are other interesting alternatives to this such as

RCC8 [1], which stands for Region Connection Calculus. RCC8 provides eight basic binary relations. In the case of the RCC8, it allows a flexibility in the processing capacity needed meanwhile the expressiveness is limited compared to Allen 2D.

A three-dimensional spatial specification more complex in expressiveness therefore, more complex in processing, by providing an extension from Allen interval logic to a 2D Allen interval logic takes us from basic thirteenth relations to the square of this number. Extending this logic into three-dimensional space will give us the cube of that number of possible relations which is even more complex to use such as for advanced virtual reality applications.

2.1.3 Hypermedia specification

Not like spatial and temporal specification, hypermedia specification addresses a more complex dimension of a multimedia document [24], as it might supersede the definition of the document itself and can go beyond it into a set of external documents. Hypermedia specification is about the hypermedia dimension that is the ability to navigate in the document itself and even go beyond that and navigate from the document to another document, or vice versa forming a complex structure. Linking from a multimedia document to another can be done using URIs [24] like the unidirectional hyperlink in HTML [18, 25], or more specific like XLINK [26] in the XML world. These URIs can address directly another external multimedia document, or in some cases only a part of the document itself.

The navigation in the content of a multimedia document itself can be expressed using some standards, as for the HashURI [27] which consists of using of a hash '#' to denote a part of a document given with its URI. Another interesting standard for this is the XPATH [28, 29] that is specific to the XML world where we get a specific portion of an XML document instead of the document itself.

External navigation allows us to compose a set of multimedia document into a more complex multimedia presentation. Meanwhile, inner navigation is interesting to allow users of a given document to navigate within the document itself, or create a complex presentation not composed of a set of documents instead, a presentation composed of a set of portions of documents.

A real representation of the dimensions of a multimedia document can be projected in spatial and temporal dimensions from the document itself that is somehow static. However, it is harder to find a realistic projection of the hypermedia dimension, the hypermedia can represent the will of the user to go through the pages of a book or review a scene in a movie, and this is dynamic because it is more based about the undefined will of the user. We cannot know what the execution path of a given document from a given user prior to the execution; this makes the case of adapting such dimension more delicate and more complex to deal with especially with external navigation.

2.2 Multimedia documents adaptation

As we said in the introduction, different end-users use different smart devices while accessing multimedia content, whether on the web or in any multimedia system. These devices may constrain the execution of such documents whether in whole or in part, also the mobility of some smart devices may constrain the execution based on the context that becomes even restrictive while on mobility. Some of these constraints are related to performance problems, others to quality [30].

Performance problems reside in the capability of a device to execute a given object in an acceptable amount of time. Such as a low-resolution screen, that does not permit the viewing of high-quality videos and images or even result in the incapacity of execution. On other hands, a low processor frequency that does constrain the time of processing, which in this case the problem requires an adaptation of the video to make it suitable for playing on this device, this adaptation is about converting the original video to a video with lower resolution as an example.

Quality problems are when having for example a screen with a very low resolution, so when converting a video to such resolution it will not serve to what it was intended, or simply the information present in the video will be invisible, or barely visible to the end-user. In such situation, there is no need to show the video at all.

Based on the previous mentioned problems we may classify the adaptation techniques based on what dimension of the document will be adapted. In the following elements we describe temporal, spatial, hypermedia adaptation and mixed adaptation.

2.2.1 Temporal adaptation

In temporal adaptation [6, 19], the temporal dimension of a document is manipulated according to either a profile, a context or both. The manipulation aims to reduce or stretch the final presentation over time, this can be done by manipulation temporal relations or temporal properties.

2.2.2 Spatial adaptation

In spatial adaptation, the spatial dimension [12, 31] is manipulated according to a profile, a context or both. The manipulation aims to either reduce or stretch the resolution of the composition in space, it can also be used to adapt the position of every element in the composition in space. Spatial adaptation can be done by manipulating either the spatial properties or the spatial relations.

2.2.3 Hypermedia adaptation

In hypermedia adaptation [12], the hypermedia dimension of the document is manipulated to change the possible navigation paths that the user can use in order to navigate in or out of the document. It can also be used for the composition of multiple documents or decomposition of a unique document, decomposing a document into multiple sub-document interlinked with hypermedia links allows the user to choose the interesting information for him, navigate the document according to his preferences and personalises his own experience.

2.2.4 Mixed adaptation

In a mixed adaptation [12, 31], the system adapts a document over more than one dimension combined, there exist multiple ways to combine multiple dimensions into an adaptation system. The most combined dimensions are those of space and time, Spatio-temporal manipulations in adaptation systems are used to change the behaviour of the composition or the presentation to form a personalised and adapted final document according to a special given situation.

2.3 Where should adaptation take place

The adaptation process can be done on different places, this depends on the specification of the situation where adaptation is needed and also on the design of the adaptation technique. In this section, we classify the adaptation techniques based on the location of the execution of the adaptation process.

In a network where multimedia documents are exchanged, adaptation can be done in three different places. A server-side adaptation, a client-side adaptation and between them using a proxy perhaps a third-party one where the adaptation is scheduled and executed. We also survey some relevant studies to adaptation techniques designed as a location-independent solution, these are solutions that can be implemented the way the developer of such solutions would.

2.3.1 Server-side adaptation

The server that contains and delivers the multimedia documents is responsible of the adaptation. Additional resources and software should be allocated for the adaptation process. In this case, there are two possible scenarios for the adaptation solution, the first one consists of making available a batch of different versions of the content for possible contexts, the second one consists of keeping only one original version of the document and based on the context or the profile a special process applies the needed adaptation operations.

Figure (fig. 2.1) shows a basic and typical architecture of a server-side adaptation solution.

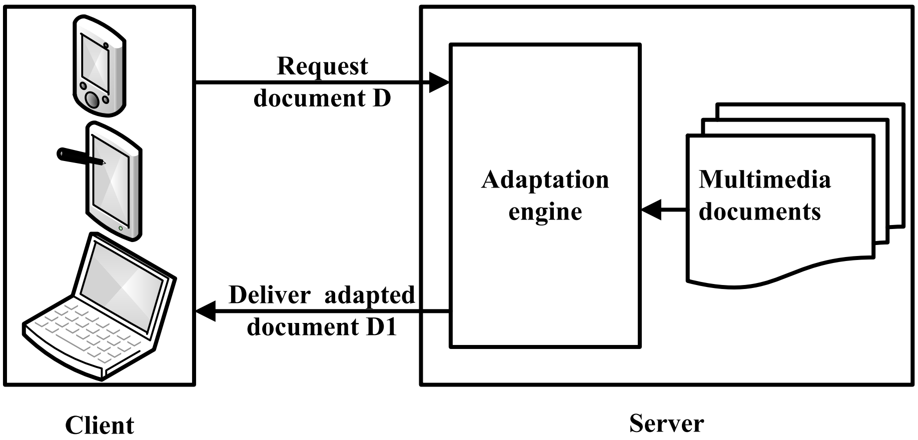


Figure 2.1: Server-side adaptation

Most researchers concentrate either on a server-side adaptation [32] [33] [6] or on a generalised location-independent solution. however, most location-independent solutions best fit a server implementation, this is caused by the higher performance of servers. The following surveyed papers uses a pure server-based adaptation technique where authors of the papers implemented their solution directly on servers or designed them to be executed on servers.

Habib Louafi et al. [6] propose a framework for robust QoE-aware prediction based dynamic content adaptation applied to enterprise slide documents for mobile web conferencing. The authors' framework concentrate on the conversion of slide documents into a web page. They tested the framework in two scenarios. The first is converting the slide into JPEG files (rasterized images) embodied in a web page skeleton because of the wide support of the JPEG format in most mobile devices, in spite its lack of interoperability especially with the absence of the keyword searches and other functionalities to the rasterized content. The other scenario consists of converting the slides into XHTML pages where each element in the slide is converted into a corresponding element in the XHTML page. The authors show a comparison of the two scenarios where the optimality is better reached when using XHTML rather than the use of JPEG. In this work, a less-treated problematic is taken into account despite the importance of such application in the field of multimedia document adaptation, less work addressed such problematic and this what make such research even interesting

Davy Van Deursan et al. present a fully integrated platform for format independent multimedia content adaptation and delivery using semantic web technologies called NinSuna [33]. A server-based platform implemented using JAVA EE, and claimed to fully deal with the significant amount of heterogeneity in terms of coding formats, delivery formats, and contextual playback environment. In order to test their platform, the authors used news sequences; the platform enables the user to select news fragments that match their preferences and personal interests. The authors rely on format agnostic packaging engines (engines independent of the underlying delivery formats). The main advantage of NinSuna is its extensibility because of its format independency, however; a global deployment of such techniques on a large scale application still requires lots of time. Convincing developers and users to follow technique requirements would make the most problematic feature of a such solution.

Davy Van Deursen et al. again present a new semantic format independent multimedia content adaptation system called RDF-driven content adaptation [32]. This technique has been used in the previously cited work; it relies on the scalable coding formats that is inspired by the principles of XML-driven content adaptation techniques, which is format independent. It also recommend that existing coding and metadata formats have to be mapped to a proposed model that describes structural, semantic, and scalability information of media bit streams. The authors used semantic web technologies to implement their technique such as RDF, SPARQL, OWL. This technique is interesting from the point of view of the authors because of its independence from the format. Whilst it might be considered as interesting because it is a generalized solution for the multimedia content adaptation, however, it might not be as realistic as it seems because of the proliferation of profiles and content variants we must deal with as mentioned by the authors. It is a very delicate problem to deal with, and proposing such a system is always interesting while in order to succeed with its deployment we need to map every existing coding and metadata format to the proposed model.

2.3.1.1 Static server-side adaptation (Offline Adaptation)

This case of server-side adaptation consists of creating a process for creating different versions of the same document for different situations or contexts at the time of creation of the document [34][35][36][37][38]. As an example, creating different versions of an image or a video with different resolutions, when a client requests the document, another process occupies of the decision about what version of the document to deliver.

This type of adaptation requires more resources in term of storage whilst its computing complexity depends only on the number of possible contexts. Other limitations of such technique reside in the impossibility of the determination of all the possible target contexts and profiles whether currently available or to become available in the future. In addition, the time required from the first publication of the document to the creation of the versions of the document cannot be suitable for real-time applications such as in web real-time conferencing [6].

Figure (fig. 2.2) shows how static server-side adaptation works in a simple case where a client requests a content from a server.

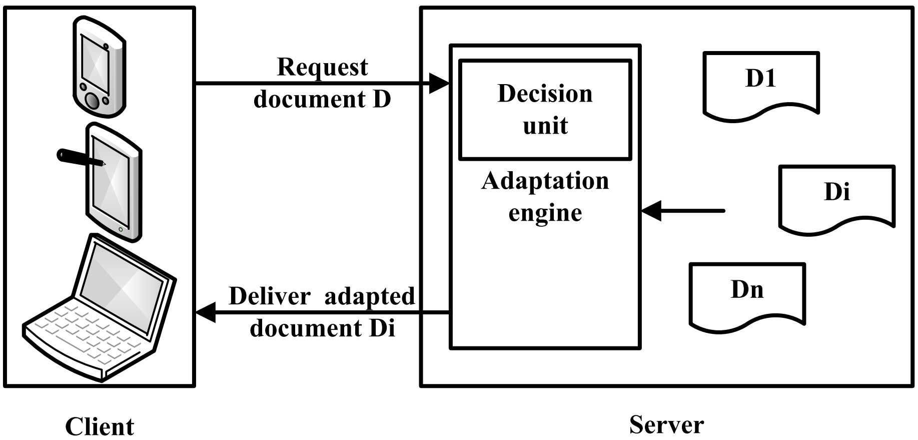


Figure 2.2: Static server-side adaptation

2.3.1.2 Dynamic server-side adaptation (Online or On the fly adaptation)

Dynamic server-side adaptation [39] [40] [41] consists of keeping only the original version of the content, while a specific process schedule whether or not an adaptation is needed whenever a client requests a document and what type of adaptation should be performed, another process occupies with the operations of adaptation.

In this case of server-side adaptation, the storage resources are preserved but the operations of adaptation that may be performed require a higher processing capacity. This technique provides a more agile solution in term of future possibilities, we may simply add or detect a useful new adaptation operation and to forget about all the dependencies of the update.

Figure (fig. 2.3) shows how dynamic server-side adaptation works in a simple case where a client requests a content from a server.

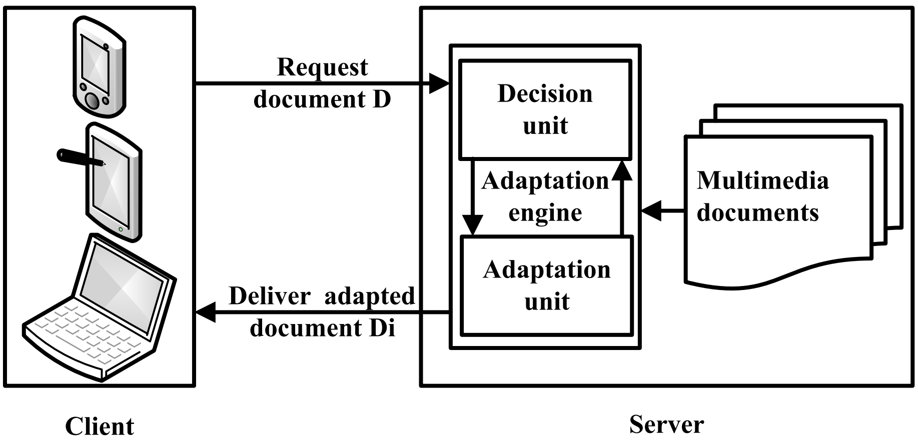


Figure 2.3: Dynamic server-side adaptation

2.3.1.3 Mixed approach (Cashing):

In mixed solutions of adaptation using both online and offline techniques, the solution consists of implementing a caching policy[9]; Depending both on the limitations of the server used for the adaptation, and the size of the data to be processed. This policy of caching is used to store the result of online adaptation so it can be used in the same scenario in the future request of the same resource on the server (a multimedia document). In this case, the adaptation engine delivers a cached version of the requested document created in a previous adaptation process (if available) instead of creating a new version from scratch. This is useful to get both advantages of the techniques while maintaining control over the allocated resources. At the same time of taking the advantages of the techniques used, we also take the same problems as well, an example to this, if such solution is needed in real-time applications such as web conferencing. In such applications, we may have look for better ways to solve the processing time issues.

A typical scenario where we want to maximize the benefits of this technique is to maintain in the same time the cashing technique and a priority technique where documents versions with low request rate shall be deleted in case the cache is full and none of the existing versions corresponds to the actual request case. The more a version is requested the more it will persist in the cache.

2.3.1.4 Service-based adaptation

One particular application of the server-side adaptation is the deployment of the basic processes in the adaptation system as services [35][42][43] that we call this application service-based adaptation system. In such a system, the workload of the adaptation process will be distributed on multiple servers. Each one contains a specific process; a main server contains a composition of services using web services technologies. The main advantage to this is the final performance of the adaptation process and the time needed from when the adaptation starts till the delivery of the adapted content. Another advantage to service-based adaptation system is the decomposition of the adaptation process. Since the basic processes already exist, the author of the composite service requires less effort in developing the final system.

A downside to service-based adaptation systems is the possible underlying cost of the final system, service providers may over-price the use of their services. another downside to this kind of systems is the availability of the basic services, this may lead the authors of such systems to implement their proper basic services. Some recent studies concentrate on a Service Oriented Approach (SOA) for implementing their proposed adaptation solutions, some relevant studies are surveyed.

Quang Pham Hai et al. [43] propose a service-based architecture for on the fly multimedia document adaptation, the services in the proposed architecture may be available locally on the server or remotely on multiple platforms. The authors propose a procedure of selection of the relevant adaptation services for the adaptation process based on profiles. This is one of the interesting applications of the service-based approach in the multimedia document adaptation, the solution does not only rely on the existing services on the server and can be extended to a remotely available solution which makes the implementation of an adaptation solution even easier by composing only the interesting services. According to the authors, adding semantic to the profiles would allow their selection procedure to provide a better way to select the needed adaptation services.

Alti Adel et al. proposes an ontology for on the fly semantic multimedia documents adaptation [44], the proposed semantic rules for the automatic generation of dynamic and quality composition of heterogeneous components. The interesting part of this work is the independence of applications and documents; it provides a core basic for automatic multimedia document adaptation.

Vanessa El-Khoury et al. propose the Personalized vIdeo Adaptation Framework (PIAF) for high-level semantic video adaptation [45] which is a fully integrated framework that provide all the requirements for semantic video adaptation, this work is based on the definition of a video annotation model and a user profile model based on the standard MPEG-7 and MPEG-21. This work is a detailed one with a significant amount of technical data, and is used for video adaptation, which is a complex multimedia form; hence, the algorithms used in the framework can be extended easily to other simple forms of multimedia content. Another important of aspect of this work as mentioned also by the authors is its user-centric approach, this we do not usually see in other studies or rarely. Usually, other studies concentrate only on minor user preferences, in this work the authors created a feedback mechanism to test the quality of experience of the users (observers as named by the authors) by providing their opinions about the result of the adaptation of observed the videos according to each method used for comparison in the work.

M. Shamim Hossain et al. propose an ACO based multimedia content adaptation approach [46], a technique that inherits the adoption of ACO-based path selection behavior in the path computation for appropriate learning content customization. This technique is used to alleviate the challenge of learning content mismatch that needs adaptation to satisfy the user in term of QoS. The study is interesting and it would be more interesting if implemented as a generalized solution for multimedia document and content adaptation.

J. H. Abawajy and F. Fudzee propose a dynamic multi-criteria path determination policy that selects an optimal path to the content adaptation services that best meet the user preferences and QoS requirements for distributed multimedia content adaptation [47], the solution tends to select the appropriate adaptation services to be applied on the multimedia content before delivery. This approach of service oriented architecture solution for multimedia content adaptation is interested in in-proxy adaptation.

2.3.2 Client-side adaptation

Client-side adaptation [48] requires that the end-user device is the responsible for adapting the document received from the server to execute it properly. This solution can be implemented by the manufacturer of a device, or any another third-party software provider like the operating system company; A web browser specifically designed for a device or a class of devices that implement a support to the client side adaptation of multimedia documents over the web.

The limitation of such technique is that a device may not be able to perform the specific adaptation process in an acceptable amount of time. Because of this, such technique should be limited to minor adaptations instead of higher cost adaptation processes. Meanwhile, an advantage of this technique is that multimedia document designer needs only to create a unique original version of the documents.

Figure (fig. 2.4) shows a basic architecture for client-side adaptation solution.

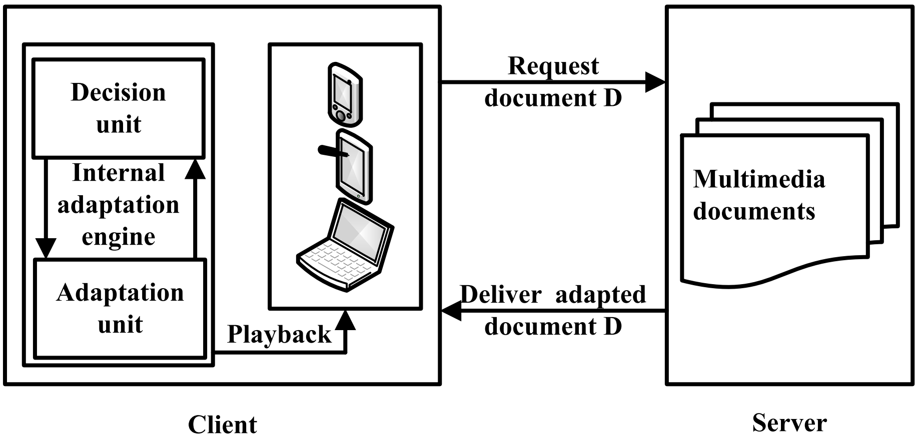


Figure 2.4: Client-side adaptation

Less work on client-side adaptation is found in literature because of the usually time consuming adaptation processes. Researchers avoid such solutions for impractical reasons however, the next surveyed work is one of the rare found in the literature that concentrates on client-side adaptation.

Abdelkrim and Nadjib propose a client based context-aware adaptation technique for SMIL encoded multimedia documents [48]. The proposed technique deals with the adaptation of SMIL multimedia documents at compile-time (before the execution with the player) and claimed to be feasible to be implemented for a solution at run-time when adjustments are needed on run-time (while playing the document). The technique details the contexts and their formats that have been studied in the work. The proposed algorithm can also be used to evaluate and monitor the QoS of a presentation, the limitation of this solution is that it relies on the processing capacity of the client; it also features the aforementioned limitations of client-based multimedia document adaptation.

2.3.3 In-proxy adaptation

Unlike server-side and client-side adaptation, in-proxy adaptation relates on neither the server delivering the document nor the client requesting it. This technique consists of an intermediate solution where installing a proxy server that acts as a medium for delivering documents from the server to the client in an adapted form.

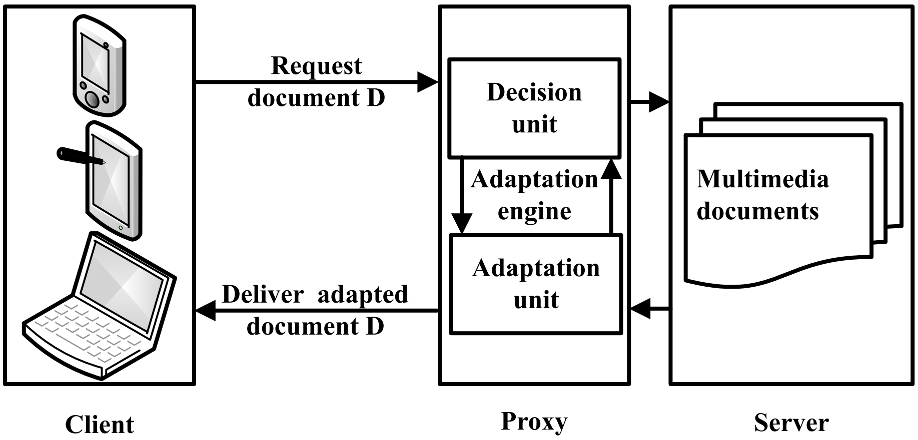


Figure 2.5: In-proxy adaptation

In-proxy adaptation solutions [8] [49] are implemented usually as a dynamic adaptation (on the fly). However, just like server-side adaptation, offline adaptation and cashing solutions based in-proxy can be implemented. In such cases, the choice of the storage or cashing location (in-proxy or in-server) has to be made according to the developers need. The only difference with server-side adaptation is the adaptation process, selection of the resources to be adapted and communication with the client are based in the case in the proxy instead, and the server is responsible for delivering the original content to the proxy or in some cases also act as a cache.

The previous figure (fig. 2.5) shows how in-proxy adaptation works in a simple case where a client requests a document from a server throw the proxy.

Following are relevant works about in-proxy adaptation surveyed.

Dongsong Zhang et al. study provides insights for research on context-aware mobile computing, adaptive interfaces, and mobile HCI . They propose a

Context-Aware, Dynamic Approach to Multimedia Adaptation (CADAMA) for mobile Web according to specific context [8]. The adaptation in this study is user-centred where each user has his/her profile stored on a web server, this profile might include information about the user from the preferences to the disabilities. The study is interesting in the point that it deals with web pages instead of a specific document model or format. However, it can impose that users of such system must be registered in order to benefit from the adaptation provided by the implemented proxy. This can lead in a platform-dependent solution that is the case studied Android. It would be more interesting if the profiles are stored on the browser level, and associate a user account to them to make the solution platform independent.

Soocheol Lee et al. present some methods for adapting multimedia documents to match user preferences and propose the user preference content value model (u-CVM) [49] that relates the content value of different modalities with resources. One very interesting part of this work is the concentration on the modality conversion. However, modality conversion can only be useful when the aim of the multimedia document is the transmission of information, like in the case of news. Meanwhile, in the case of multimedia content as visual art or in the case of movies and videos, the purpose of the video or audio cannot usually be met when converted to other modalities and therefore cannot evoke the same quality of experience or not serve the intention of the authors at all.

2.3.4 Location-independent adaptation

In this category, researchers would just give a design for an adaptation system [9][31]. The implementation of such system depends solely on the developers based on their particular needs. The developers can either implement the solutions to be executed on the server, proxy or in some rare designs, the solution might be able to fit into a client based implementation.

Following are some recent location-independent works surveyed.

Sébastien Laborie et al. propose a concrete application of an earlier framework they proposed for semantic adaptation of multimedia documents [31].

An interesting part of this work is that the algorithm designed is a generic solution based on an abstract structure that captures the spatio-temporal and hypermedia dimensions of a multimedia document, and then they apply the algorithm on a real example using SMIL documents. The algorithm transforms the structure minimally according to the device constraints; according to the authors, their solution is location independent and can be implemented on the client-side or server-side, and even in an intermediate solution (proxy). This is essential for a global solution, however like we said about other generic solution the deployment of such technique the major limitation.

N. Saravanaselvam et al. propose a cost-effective technique that reduces the data consumption in web mobile browsing where the billing is usage-based pricing [9]. It sounds not related to the problematic of adaptation when we talk about prices; however, the key idea behind the reduction of the costs of browsing is to adapt each web page before being transmitted to the mobile device. The authors used a simple and effective content adaptation technique to decide if an object best fits the mobile specifications, and better presents the original content like in an image with the optimal resolution. The proposed technique is useful for those countries where internet services are usage-based billed and can be used as a user-preference when the user wants to reduce the usage of the internet bandwidth for other reasons like preserving the capacity of their batteries.

2.4 Discussion and comparison

In the following table, we opted on the same comparison scheme done in an earlier survey on multimedia document adaptation for mobile devices [50], this allow us to take an overview of both surveyed papers and provide a global comparison.

We consider the following parameters for comparison between the surveyed works:

- Where: (S—Server, SVC—Service, P—Proxy, C—Client)

- When: (P—Preprocessed/Stored, R—Real-time/On the fly)
- What: (T—Text, I—Images, M—Multimedia)
- Level: (S[1-3], S1 = Physical, S3 = Semantic)
- User Controlled: (U[1-3], U1 = No User Control, U3 = Full User Control)
- Feedback: 0/1—Is it resource adaptive?
- Caching: 0/1—Implemented?

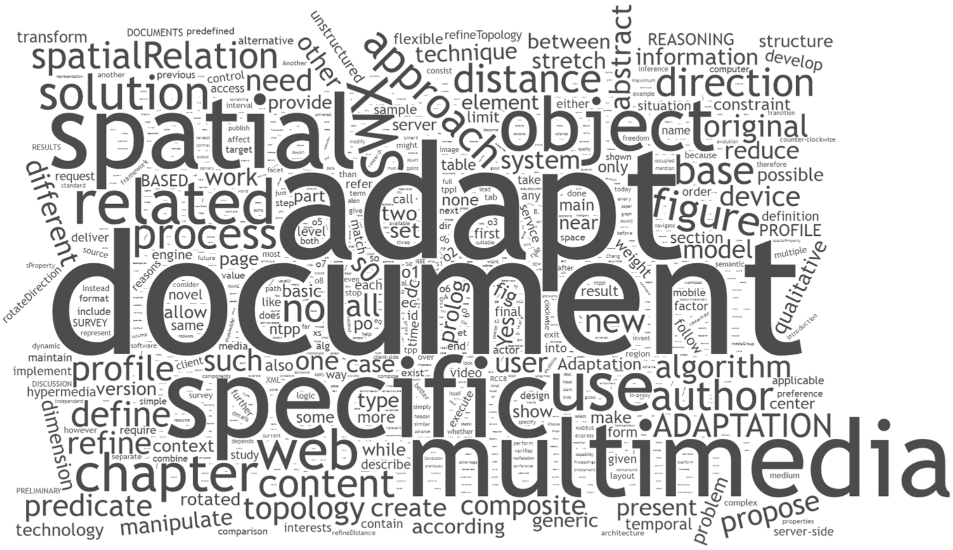
Table 2.1: Comparison between similar adaptation solutions

Paper	Where	When	What	Level	User control	Feedback	Caching
[6]	S	R	M	S1	U1	0	0
[33]	S	R	M	S3	U2	0	0
[32]	S	R	M	S3	U2	0	0
[48]	C	R	M	S1	U1	0	0
[31]	S,C,P	R	M	S3	U1	0	0
[43]	SVC	R	M	S2	U2	0	0
[44]	SVC	R	M	S3	U2	0	0
[49]	P	R	M	S1	U2	0	0
[45]	SVC	R	M	S3	U3	1	0
[46]	SVC	R	M	S1	U2	0	0
[8]	P	R	M	S1	U3	0	0
[47]	P	R	M	S3	U2	0	0
[9]	S,P	R	M	S1	U1	0	1

In this chapter we surveyed, multimedia documents adaptation studies and some related problems. We discussed multimedia document specification, adaptation techniques were categorised and compared.

The studied works in this chapter are only a small part of the existing studies in the field of multimedia documents adaptation. It is just a step in order to make a full study, however we noted that most of the works are specific and does not address the problematic from a real generic viewpoint and just address partial problems. It is not as easy as it seems but a generic solution or a workaround to the adaptation problematic is needed.

In the following chapter, we discuss a possible workaround to the adaptation problematic. Using a web facet that stand as a parallel adaptable version of the web existing documents while maintaining some factors we discuss along the chapter.



End

Chapter 2;

Begin

Chapter 3;

Chapter 3

A novel approach for adaptation

In the previous chapter and as explained, the increased use of smart technologies led to the necessity of adaptation to improve the accessibility of multimedia in documents in general. Different factors have transformed the web in particular into a chaotic and an unstructured nature, therefore, a growing need for adaptation with each new invention in multimedia. Different context and usage of multimedia imposes different techniques and solution, however, a good look into the surveyed papers in the previous chapter leads us to think about the need for a more generic approach to adaptation.

This chapter proposes a workaround the problematic of adaptation and is organised as follows.

Section 3.1 explains some expectations from the web of multimedia and why adaptation is needed. Section 3.2 explains the behaviour of different actors on the web of multimedia, section 3.3 proposes a novel approach and concludes the chapter.

3.1 Adaptation of multimedia documents

Multimedia documents are any form of content including text, images, videos, audios in an electronic form as defined in (definitions 2.1, 2.2, 2.3), and multimedia documents adaptation is to make this content suitable for the context of its execution [12, 19]. A context is the state and characteristics of the end-user and the device running the document and the environment of the device [19, 51]. The characteristics or the user preferences [19], and the device capabilities form a profile for adaptation, it is used to help decide which adaptation process is needed in order to adapt the document to the current context.

In [52] the authors try to define the terms of context and context awareness, by exposing some existing definitions and proposing a new one for each term.

Definition 3.1. A Context as defined by the authors is 'any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves '.

Definition 3.2. Meanwhile context-awareness according to the authors is "any information that can be used to characterize the situation of an entity, where an entity can be a person, place, physical, or computational object" and "the use of context to provide task-relevant information and/or services to a user, wherever they may be."

Each context and profile defines the state of the user based on information collected by a software on the end-device. Whether specified by the user via a preferences menu or by using dedicated sensors to help perceive the environment, also by listing the specifications of the device characteristics including capabilities and available codecs and the compatibility between the original document and the device.

3.1.1 Expectations from the Web of Multimedia Documents

The inventors of the web [53] thought it was a good platform for sharing information, at first it was meant for academic purposes. The generalization of this technology and the birth of the internet with the simplest forms of information sharing technologies was thought to start with a structured document form which is HTML [25], and leads to create a structured web of information.

What happened is the need for more out of this technology, different opinions, technologies, and problems, all this led to an unstructured web where each document is structured but in its own form.

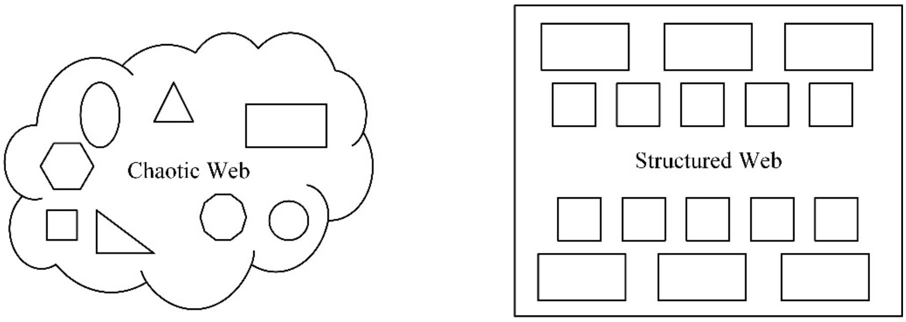


Figure 3.1: Unstructured vs. structured web

Figure (fig. 3.1) shows what was expected to be structured web and what is the web today as an unstructured web.

In a structured web, we should be able to know the dimensions of the web, we can point to any information, and any object in the web just by using URI [24], and we can access the web anywhere at any time using any device.

What we have today is simply contrary to what we expected the web to become, because the architecture of the web does not permit evolution in a good way.

3.1.2 Motivations for adaptation

As we said in the introduction, different end-users use different smart devices while accessing a multimedia document, these devices may constrain the execution of such documents contents whether in whole or in part [19, 54], some of these constraints are related to these two kinds of problems: quality problems and performance problems as explained in the previous chapter in section (sec: 2.2).

In different terms, adaptation is intended to make a fine line between the unstructured nature of the actual web and better serve the accessibility by maximizing the ability of users to view and play a document on theoretically any multimedia device, anywhere and at any time and thus according to the user current context. Different studies have been conducted in order to make this possible, besides it is clear that the task is very complex and important for the evolution of the web of today. Most of the available works concentrate on specific adaptation related problems as presented in the previous chapter, no general approaches to tackle the real problematic.

But what is the real cause that leads to the necessity of adapting multimedia documents? Is it just because multiple devices and software are being developed almost every day? Different models? New types? New things probably no one would imagine them before?

The following reasons do not engage the developers to use the current trends in development other than creating their own methods and technologies:

- The proliferation in the creation of heterogeneous devices and systems.
- The lack of a global approach of development of both devices and systems.
- The lack of standards.
- The lack of methodologies.
- The lack of a better and suitable solution for almost all situation.

All that we mentioned above leads to the total freedom for developers of smart devices and software industry to make their own models, to make all design choices, to not have a single general model or approach to respect.

This freedom is a good thing and a source for creativity, excellence, and evolution but the total freedom leads to the transformation of a world wide web which is intended to be a huge set of structured documents and information into a web of an unstructured set of structured documents. This creates the problem of inaccessibility, different documents models, and context types result in the difficulty of creating a suitable document for all possible situations or a suitable solution for everyone.

Does what we said before means we should create a cage for creativity or not allow development freedom? Of course, this is not an option and definitely not a solution to the problem of adaptation. But how can we create the kind of approach that will guarantee the freedom of developers while maintaining an even structured and accessible web?

3.1.3 A Growing Need for Adaptation

We may think somehow about today raising technologies and say in the near future there would be no need for adaptation, because of the capabilities of new devices and developers are more likely to choose existing standards rather than creating their own. Research for better solutions is ongoing every day and like we thought high definition HD video is good enough when it first appeared, today we have 4K, 8K displays and we will probably have better displays in the near future. Not to mention the probability of the generalization of the use of multi-sensorial media [54], therefore a need to adapt today's document that is created for today's technologies to be suitable for tomorrow's technologies is a must.

This would create an increasing need for adaptation solutions, so whenever a new technology is developed; all content providers and all developers will have to make their content suitable for. Or in the other case, the developers of the technology should provide a bridge from every single possible source and make their solution the ultimate one. This is not practical and probably impossible for anyone whether for developers, researchers, content providers, service providers or end users.

How can we overcome such a problem?

3.2 Different actors on multimedia documents

In multimedia whether on the web or on any other information publishing or diffusing system, there are basic actors on the process of publishing a given document. The first actor is the content provider, who need to publish some kind of information in a multimedia form. In order to allow multimedia publishing technologies are created, software technologies, hardware technologies and all other kinds of technologies involved in the creation and processing of multimedia documents. therefore, software developers and device makers are also both actors in the process.

In order to allow such publishing process by content providers using such technologies made by software developers and hardware makers, there must be a service of publishing let say as an example internet service in general terms.

So the actors on the multimedia documents are as follows:

- Content providers.
- Software developers.
- Hardware makers.
- Publishing service providers.

All these actors are involved in the process of publishing multimedia documents, each one has a part of responsibility in creating the most accessible form of multimedia documents, and their behaviours can affect this process.

What kind of behaviours that make it difficult to allow a smooth and evolutionary process?

We believe that how everyone in the system acts defines the actual problems we have because we are the one who created the web, we are the one who invented multimedia, we are limited and non-perfect human beings and our inventions are as limited as we are. We cannot create perfect inventions, so we have to do the best we can to invent and create the best things we can.

The behaviours that are causing the different actors to create heterogeneous inventions in the same field are:

- Everyone wants to be unique
- Everyone wants to be the first who invent something
- Everyone wants to privatize something that should belong to the public in order to make more profits.

All these behaviours make the web today in an unstructured nature. Say someone wants to create a new format of multimedia documents to represent an image, he can simply use XML for this purpose and when he comes across the definition of the dimensions of the image. He has the ability to name the attribute the way he wants. Whether D, Dimension, Dimensions, Length and width, resolution and all possible names that represent almost the same thing. Even more, he may choose a different language, and therefore a new problem to understand this format by other technologies developers.

3.2.1 Limitations Versus Freedom

It is clear that we have not yet invented the best or the ultimate solution that makes it possible to not converge to the need for adaptation, the real question is how can we limit actors behaviours.

If we create a solution that is the ultimate one for the process of publishing multimedia documents on the web, should we oblige actors to use this solution? Or give them the freedom they already have and therefore we solve nothing?

We started with a structured web we made it an unstructured web, we cannot transform it back to its first state like shown on the next figure (fig. 3.2), where the second transition is the needed one. The solution is to restructure the unstructured web that we made, it is easy saying than doing because of a large amount of data on the web today and all the restrictions we mentioned, hence, what we should do to solve this problem is trying to find a workaround the unstructured web and try to make the different system actors converge toward using this workaround. If such workaround is possible the web will become structured again. Is that even possible? Such solution would have to pass lots of tests before it may be qualified for the mission of restructuring the unstructured web. So if we were able to find such solution, and we make the leap from an unstructured web to a newly structured web, would this be able to face the freedom of system actors?

Very likely the problem will become recursive no matter what the solution we made passed tests, because of the previously mentioned reasons.

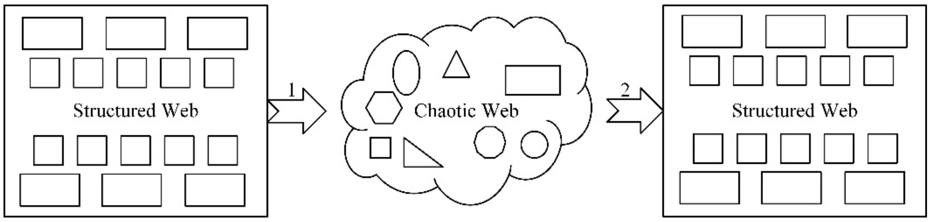


Figure 3.2: 1- First transition from structured to the unstructured web, 2- The needed transition

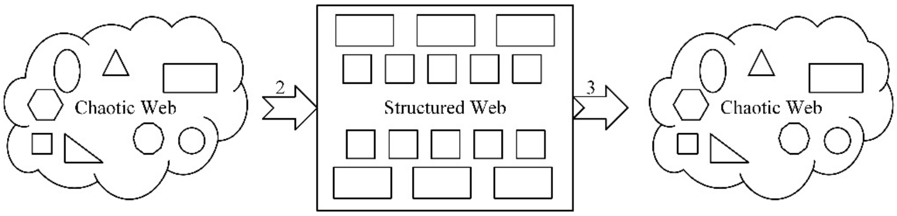


Figure 3.3: Auto-transition from a newly structured web to a new unstructured web

And lots of other reasons, so the result would be like shown on the previous figure (fig. 3.3) which is an extension of the figure (fig. 3.2) where the third transition represents the transition from a recently restructured web to a new form of an unstructured web.

This kind of recursive problem requires a different approach to deal with it, in the next elements we describe a proposition of a novel way to workaround the problem.

3.3 A Novel approach

In this section, we present a novel approach to look at the freedom of actors, propose a workaround the actors limits on multimedia documents contents so that we can make an efficient solution to the problem and suggest a facet that might restructure the web.

We showed different actors freedom and how it transformed the structured web to the actual unstructured web, one that needs to be restructured in order to allow global accessibility of information and minimize the needs for adaptation of multimedia documents. We also showed how could such limits requires a tremendous work and finally, it might lead to the same initial problem. Because of this, a middle approach between freedom and limitations should be taken where no total freedom and no total limitations.

In this approach, we guarantee that actors are somehow satisfied with the solution maintaining their creative freedom at the same time. Actors can do whatever they want, this approach is about creating a parallel face of the web without transitions from an unstructured web to a structured web. However, we maintain each type and keep both structured and unstructured web in coexistence; where both of these two extremities can support each other. In order to maintain this coexistence, we have to create a parallel-structured web and maintain the unstructured web as it is today. We should then put boundaries on the structured side, and we make different actors converge toward using the technologies of the new structured web. At the same time, using any other technology or new technologies of their choice in the unstructured web.

This new web is only a facet of the unstructured web not a new independent one, it is only a virtual way to structure the web and maintain its evolution for a long term, the next figure (fig. 3.4) shows how this facet should look. A structured facet of the web should allow different actors to access the multimedia or any other contents on the web through a structured way while allowing them to freely access and manipulate the unstructured web the way they want, technical details of the facet would be part of a later work.

All the actors who contribute to the evolution of the web should follow the guidelines of the facet and maintain its integrity, it should allow evolution in term of future technologies and should maintain its structured form to not fall in a recursive problem and make the facet another unstructured surface of the web. Users are able to contribute to the evolution of the facet in a way that they make the facet even strong against the unstructured nature of the web, if such solution is made, it should give a new and an emerging web while all actors are satisfied.

Sometimes we start by inventing something, it becomes revolutionary and it sounds something great, it may be, but as we mentioned above that we are

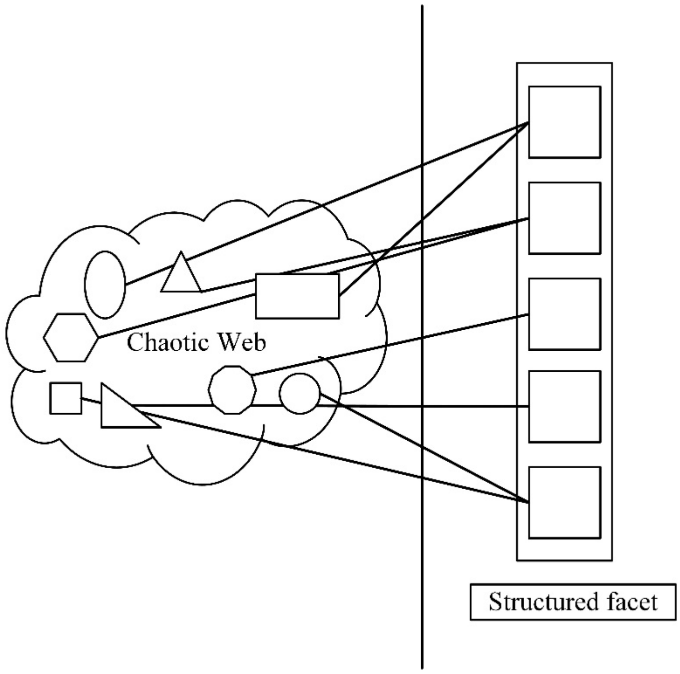


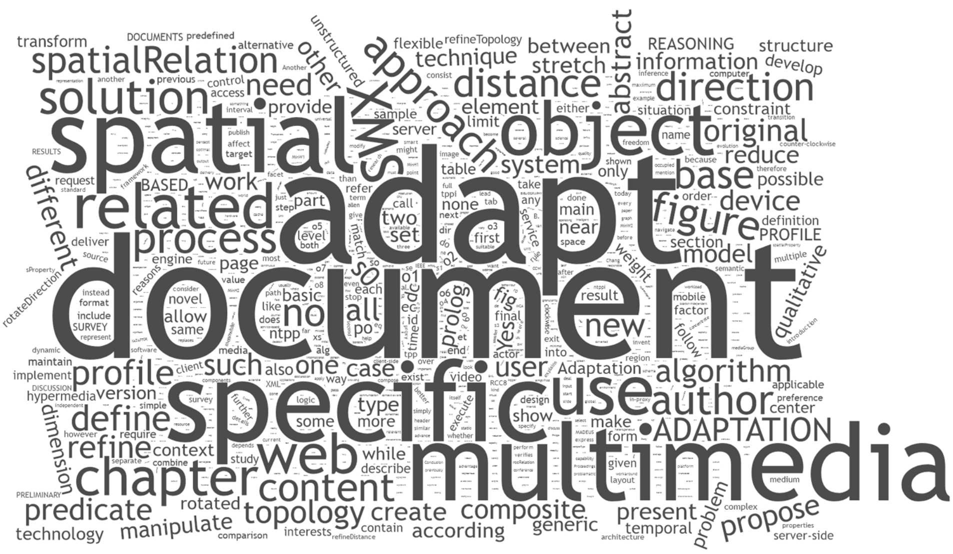
Figure 3.4: A structured facet of the unstructured web

non-perfect human beings and we simply are not able to create something perfect, however, sometimes imperfections make things perfect.

Our inventions can create new problems, therefore we look for solving those problems in order to make them better, and sometimes we just need to take a step back and discuss what was created first.

In this chapter we discussed the status of the web today, and adaptation related problems, we found that in order to make an efficient adaptation solution we have to restructure the unstructured web first so the adaptation work is at its minimum, we proposed a workaround the problem by using a facet for the actual web where to allow different actors on the web to have the freedom of creativity while maintaining the structured form of the web intact.

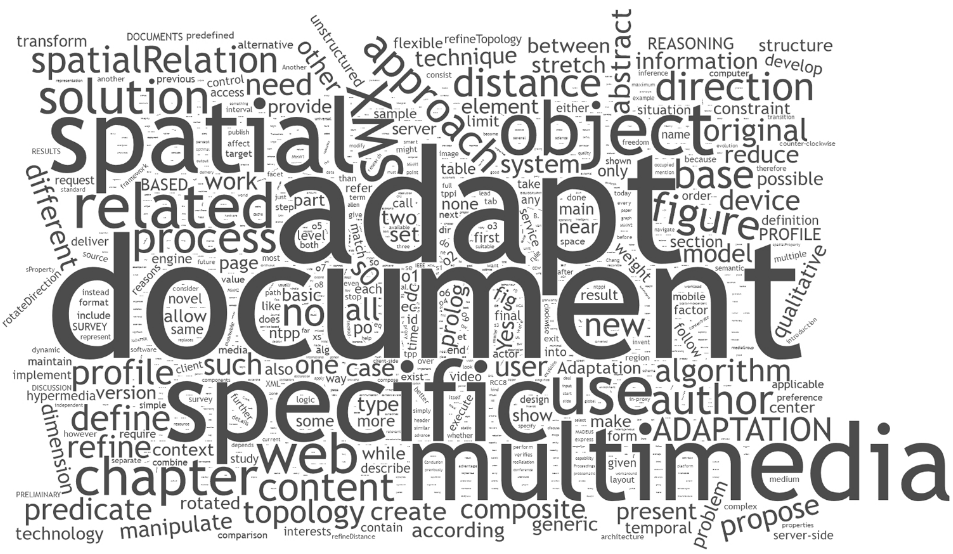
This chapter is a first step in the motivation to make this proposition a practical solution for the web's future, and allow better accessibility and adaptability of multimedia documents. The following chapter provides a practical way to implement the facet proposed in this chapter, using an abstract version of a multimedia document as a base construct.



End

Chapter 3;

{



Begin

Chapter 4;

}

CHAPTER 4

SPATIAL REASONING FOR PROFILE BASED ADAPTATION

Mainly with the latest technological advancements, new and emerging technologies are seen every day on a continuous basis. In each of these advancements, the difficulty of creating new uses for these technologies becomes harder than the previous level. Mainly because of the growing speed of creating new devices that cross the edges and limitations of the previous ones. Multimedia devices are no exception to this, every day we get a step further from the earlier technologies; however, the software industry improvements and evolutions are usually slower than the hardware industry. In this wide evolutionary environment, multimedia design and publishing services struggle to create a universally accessible multimedia (UMA: Universal multimedia access) [3][4][5]. Several studies tackle this issue using different approaches; the main used approach is to maintain alternatives such as for mobile devices [6][7][8][9][10].

Today, we are continually approaching the creation of new devices such as virtual reality technologies. Not only mobile devices in these days are touched by considered an important multimedia terminal. Web browsing still not used as part of the virtual reality industry, this may lead shortly to the need not only to adapt web multimedia to limited devices. In the first years when adaptation was proposed as a problem [55] only mobile devices were considered. Shortly adaptation would be needed to adapt multimedia documents designed for conventional computers to multimedia and web content for more advanced devices. For this, based on device characteristics or user context automatic reasoning and adaptation of multimedia is needed to provide customized multimedia content to users. In this chapter, we show how we can use spatial reasoning methods to adapt a given original document to a new version according to a target output.

The first section of this chapter provides a glimpse view over the multimedia document adaptation and adaptation approaches. The second section shows some adaptation approaches factors. The third one proposes a novel multimedia document processing model XMS.

The Fourth section presents how the abstract document model XMS allows creating an abstract spatial specification using a combination of qualitative distances, directions, and topologies. The fifth section explains our abstract spatial specification manipulation paradigm and provides the basic manipulation algorithms.

Section six presents the main algorithm that uses algorithms from section five to adapt a given abstract spatial specification and concludes the chapter.

4.1 Multimedia document adaptation

Based on definitions (def. 2.1, def. 2.2) one can understand that a multimedia document can contain any form of data that can be interpreted, presented by the terminal device and perceived by the final user. Therefore, to interpret and present such data, we will face the diversity of possible situations, where users can view or use these multimedia documents; from a basic two-dimensional static or dynamic output to three-dimensional static or dynamic content see multi-sensorial multimedia technologies [54, 56]. These situations vary on several levels, on a hardware level, a software level, and a user level which comprise in the context and preferences or even his incapability

[57][58][59][60][61]. Executing a possible presentation involve interpreting the information within the document. The several usage situations may interfere with the interpretation sometimes by constraining the execution, and sometimes by the misinterpretation of the structure of the document resulted from the misuse of the coding standards by software or hardware developers.

All these variables generate a need to create a solution for adapting such documents to fit in all these situations. Several techniques have been proposed; they can be classified according to [62] into these main approaches: an approach based on alternatives specification [11, 16, 17, 63]. Another based on transformation rules [13, 19, 64], and another approach based on flexible document model [65, 66].

4.1.1 Adaptation based on alternatives specification

In the proposed approach of adaptation solutions, the authors have to specify the appropriate alternatives in case of incompatibility of the default multimedia document with either the terminal or the user context. This solution requires that the author knows in advance all possible situations where and when their default document or some components cannot be presented properly. They can specify the triggering constraints that allow the adaptation engine to find a suitable version, or simply define the target profile properties such as the screen size or the operating system.

4.1.2 Adaptation based on transformations rules

Instead of defining a whole alternative based on constraints or properties. In the proposed approach, the author has to specify rules for the adaptation engine to perform a set of transformations on the default document or a part of it, just like the alternatives specification the author has to know all the possible situations that need adaptation in advance.

4.1.3 Adaptation based on a flexible document model

Instead of forcing authors to predict all possible situations where their documents should be executed, in solutions based on this approach, the author specifies a set of decomposed or semi-composed objects then the adaptation engine is the responsible for composing the final presentation based on these objects [19]. Doing this, the authors have not to know the possible situations of execution of the presentation. Meanwhile, they have no control over the final presentation.

4.2 Adaptation approaches factors

Previously, a glimpse view of adaptation approaches has been provided. Each of the presented techniques has a proper application, table (tab. 4.1) shows some differences based on the following five factors are summarized:

Table 4.1: Adaptation factors for used adaptation technique

Adaptation technique	**Abstraction**	**Workload**	**Separation**	**Flexibility**	**Control**
Based on alternatives	0	2	0	0	2
Based on transformation rules	1	1	-	1	0
Based on flexible document model	2	0	0	2	0
Ideal approach	2	0	2	2	2
Our approach	1-2	1-2	2	1-2	1-2

- The first factor is about the level of abstraction of the adaptation approach, a highly abstract one would exclude the author from the process of adaptation, this means that the author should not care about the adapted version of his document. Instead, the developers of the approach are the one who takes full responsibility for the adaptation part; this means that the approach is entirely automatic and the authors have no control over it.

- The second factor is about the author workload; a good approach should minimize the workload of the author so he would concentrate more on the content rather than the process of adaptation. Most of the existing approaches give either full control with lots of work to the authors; or minimize the work of the authors while giving him no control over the adaptation process.

- One of the important factors here is the level of separation. By separation, we mean separating the original content and adaptation that might be in the description of alternatives or the layout and formatting of the content of a multimedia document, as we can see in table (tab. 4.1), the usually used approaches do not consider this. Existing approaches are based on the content meanwhile, in this work, the adaptation parameters are either expressed with the content explicitly in the form of rules or something else, or in other cases are automatically extracted from within the content based on the properties of objects or the document itself.

- Another important factor is the level of flexibility which defines how much flexible the approach can be. A flexible approach should allow maximum flexibility over time without the intervention of the author, given a document created to be widely adaptable, with the appearance of new technologies the author of this document according to a flexible approach should not worry about the continuous adaptability.

- An underestimated factor in multimedia document adaptation approaches is the control of the author; authors need to have a maximum control while maintaining the rest of other factors at optimality. Usually, the adaptation approaches try to minimize the author workload and not think about the control of the author; the approach developers have no authority over the content of the multimedia document that has been created by an author. Some authors might consider consenting the developers such authority while others simply want the full control over their document, which leads them to only one option is to create a solution based on alternatives; where this approach is the one that puts the more workload on authors.

The next qualitative values are considered for comparison between the approaches:

- 0 = low value
- 1 = mediocre value
- 2 = high value
- - = depends on the implementation.
- 1-2 = optional depends on the authors' choice.

Table (tab. 4.1), the ideal approach is used as a benchmark; it should maximize all the factors while minimizing the author workload. It is clear that each approach has a strength in one or two factors. However, it may not be possible to create one ideal solution that works for every situation while maintaining other factors at optimality. To develop such solution, a priority of consideration for these factors is needed. The most important factor considered in our approach is that of the author control because he is the one who knows better the content of his multimedia document and how it should be adapted to different situations.

The proposed approach in the table sets the goals we want to reach; to maximize the author control over his multimedia document a flexibility on the other factors is a must. This way allows us to maintain a balance in all the factors we proposed, the author has to consent us to have control over his content, but if he chooses to have the full control, then we should allow him to use our solution and yet maintain the optimality of other factors.

To allow such flexibility while maintaining the authors' control, our approach is based on an abstract document. This document (XMS) is used in the processing model instead of the original document; a basic version of the abstract document might be generated from the original one. Advanced options can be defined by the author to give him the control he wishes over the document either by using alternatives, restricting or directing transformations.

4.3 A novel multimedia document processing model

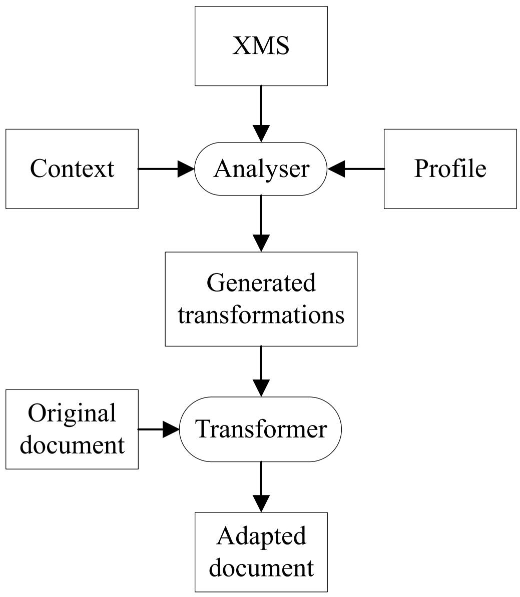


Figure 4.1: Approach architecture

Our processing model takes as input a description of the original document we call XMS (XML Multimedia Specification); this file written in XML contains an abstract version of the original multimedia document. The fact that the processing model takes as input an XMS document instead of the original document helps us to maintain some factors at optimality (Separation, abstraction, flexibility). The components of this abstract document are the key elements for all other factors we set as a goal (author workload, author control, abstraction, flexibility).

In the web in general or any other multimedia delivery system, all begins with a user requesting a document on a client device; the multimedia server then delivers the requested document. In this processing model (see fig. 4.1), the author creates or generates an abstract version for his multimedia document. After a request, the server takes into account a set of parameters and properties that describe the state of the requesting user device or even the user preferences and context, and instead of processing the document directly the system takes as an input the abstract document XMS. It allows the processing engine to decide if an adaptation is needed, and eventually, it delivers the original document as it is or generate a set of transformation and passes it to the transformation engine that uses the original document and a set of transformations to produce the adapted version.

4.3.1 XMS: an abstract document model

The core of the XMS document model is based on definition 2.4 of a multimedia document. As explained in the defintion a document has four dimentions so make the XMS more generic, the logical dimension is considered as a part of the original document and is not defined explicitly in the abstract document. This also helps us to maintain a level of abstraction to give a space for the creativity of the author.

Definition 4.1. A media element is the basic element used by the author to compose his final multimedia document, it can be a text, images, videos or any other medias.

Definition 4.2. An object or an abstract object in XMS to describe the composition of a multimedia document is a set of at least one basic media element; it is an indecomposable part of the composition. An object can be defined as an essential element of a multimedia composition created by an author. Given a document D composed of N media elements, the number of objects is M where $M \leq N$.

Definition 4.3. Specification granularity: the number of objects in a specification by the number of basic elements defines the level of granularity of a specification, a granularity equals to one is the most detailed specification of a multimedia composition. The granularity can be part of the interval $]0,1]$, the bigger the granularity is the more detailed the specification we get.

In XMS, a basic specification can be generated from the original document itself. Another possibility for this is to make the authors provide a specification explicitly in it, or giving a separate file to generate the final specification as an XMS document.

Definition 2.3 is for a multimedia document; meanwhile, an abstract multimedia document does not contain the set of objects. Instead, the abstract document contains only the set of identifiers referring to the actual objects in the original document. The figure (fig. 4.3) shows an abstract UML design of an XMS document.

4.3.2 XMS components

```
<xs:element name="XMS">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="header" type="headerType"/>
      <xs:element name="objects" type="objectsType"/>
      <xs:element name="specifications"
                  type="specificationsType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Figure 4.2: XMS main components in XSD

XMS is composed of three main elements, a header containing the main declarations, the objects elements defining and referencing the objects of the composition and the specifications element that contains information about the dimensions of the document. The main XMS elements written in XSD schema are shown in figure (fig. 4.2).

4.4 Abstract spatial specification using XMS

To define a spatial composition of a multimedia document or its layout, a look at the used formalisms to describe spatial relations is needed. An abstract spatial specification needs to define a set of qualitative composite relations using

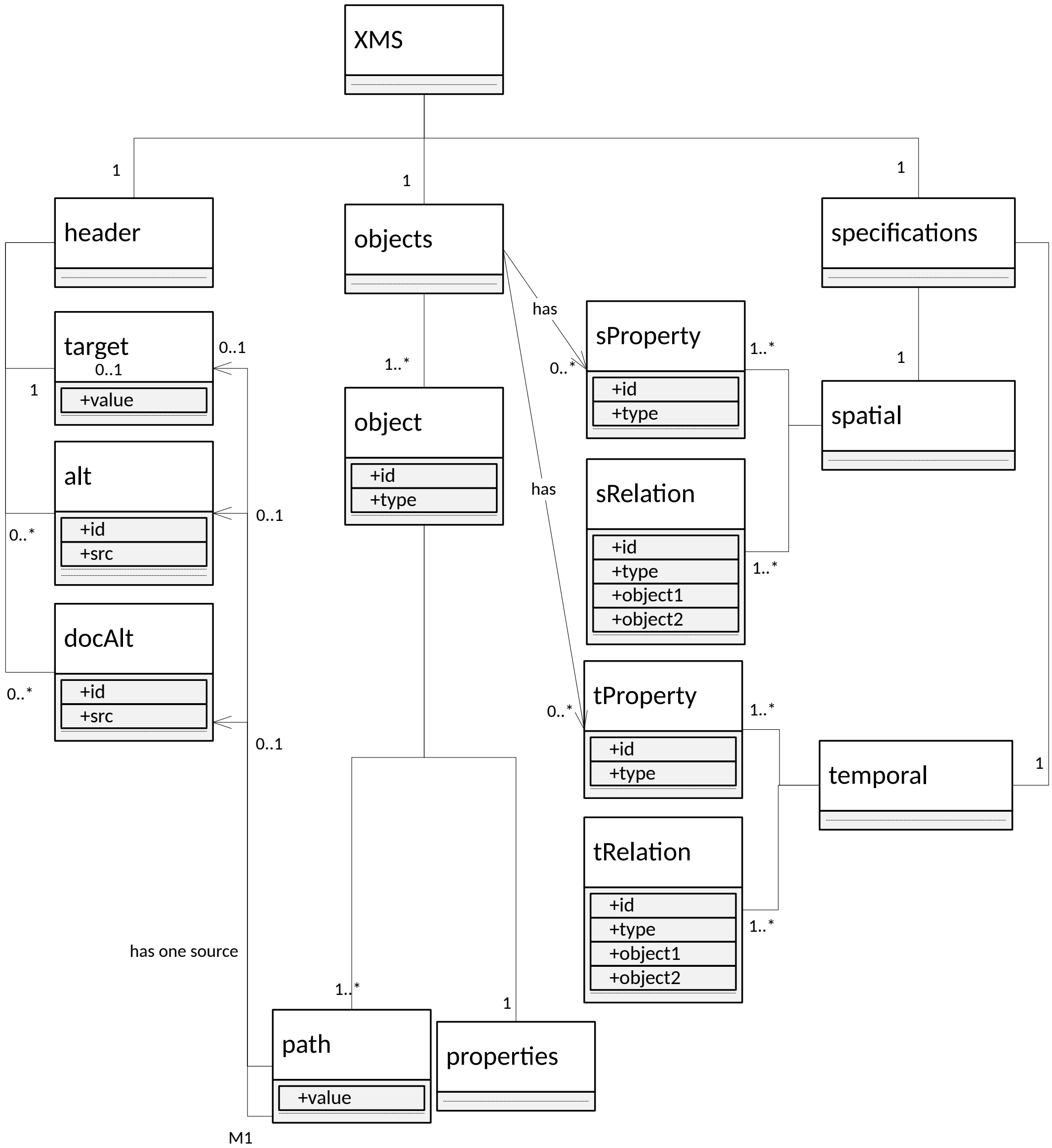


Figure 4.3: XMS abstract UML design

both topological, and directional relations refined with qualitative distances [67][68][69].

The next elements defines three different spatial information used in XMS to define an abstract spatial composition.

4.4.1 Qualitative topological specification

To represent topological information in XMS, the well-known RCC8 [1] are used (fig. 4.4). This paradigm covers eight basic topological relations in a two dimensional space. These can be changed according to a conceptual neighborhood graph (fig. 4.14) in order to adapt a composition.

Each basic relation in RCC8 describes the topology between two regions A and B.

- DC (is disconnected from): the two regions are disconnected the one from the other, the region A and B does not contain one another nor the borders are in contact.
- EC (is externally connected): the two regions are not included the one in the other but the external borders of region A are in contact with those of region B.
- PO (partially overlaps): the two regions are overlapped by one another; the region B overlaps the region A.
- TPP (tangential proper part of): $A\ TPP\ B$ the two regions are included one in another in this case A is included in B, and the borders of region A are in contact with the borders of region B.
- $TPPi$ (tangential proper part of inversed): $A\ TPPi\ B$ same as $A\ TPP\ B$ but in this case B is included in A, this is the equivalent of $B\ TPP\ A$.
- $NTPP$ (nontangential proper part of): $A\ NTPP\ B$, the region A is included in the region B and the external borders are not in contact.
- $NTPPi$ (nontangential proper part of inversed): $A\ NTPPi\ B$, same as $A\ NTPP\ B$ but in this case the region B is included in the region A, this is the equivalent of: $B\ NTTP\ A$.
- EQ (equals): $A\ EQ\ B$ or $B\ EQ\ A$, the two regions have the same position and the same size and borders.

4.4.2 Qualitative directional specification

Same with the topological description of a spatial composition, in order to define an accurate position of objects in a spatial composition we need to add

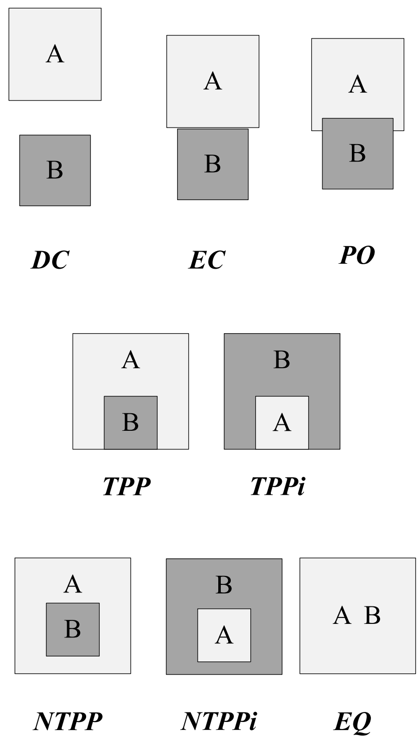


Figure 4.4: The eight topological relations in RCC8 [1]

information about the direction [2][70][71] of an object according to a reference object. The direction is defined by the direction of the center weight of each objects. According to the previous figure, we count nine distinct directional relations. A directional relation between two objects A and B is described as follows: A Relation B, the nine relations are explained in the following items:

- **None** (A NON B): no direction, same as B NON A. In the case the center weight of each objects are on the same position. This is a neutral zone and simply describes the concept of here.

- **North** (A N B): the center weight of the object A is on the north of the center weight of object B.

- **North-east** (A NE B): the center weight of the object A is on the north-east of the center weight of object B.

- **East** (A E B): the center weight of the object A is on the north-east of the center weight of object B.

- **South-east** ($A\ SE\ B$): the center weight of the object A is on the south-east of the center weight of object B.
- **South** ($A\ S\ B$): the center weight of the object A is on the south of the center weight of the center weight of object B.
- **South-west** ($A\ SW\ B$): the center weight of the object A is on the south west of the center weight of object B.
- **West** ($A\ W\ B$): the center weight of the object A is on the west of the center weight of object B.
- **North-west** ($A\ NW\ B$): the center weight of the object A is on the north west of the center weight of object B.

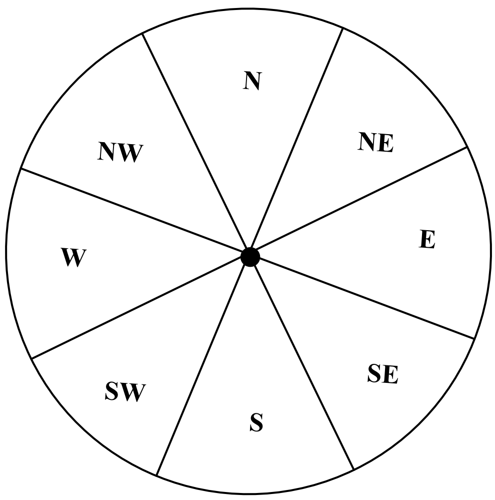


Figure 4.5: The qualitative directional relations [2]

4.4.3 Qualitative distance specification

To define the spatial composition, information about the space between two objects needs to be considered in the description of the spatial relation in a specification. Here we describe the qualitative distance [2][70][71] specification of a spatial relation between two objects.

There are many paradigms when it comes to qualitative spatial specification, and qualitative distances are no different. The distance in this work is described as the distance from the center weight of the two objects A and B, and not the distance between the borders of the objects. To maintain a comprehensive specification, the authors have to define a set of intervals where the qualitative values might be, the next figure shows a set of four basic qualitative distances.

Four different qualitative distances are used in XMS, the first is the *none* where no distance is present between the two center weights of the two objects A and B, and are on the point 0 in the next figure. There are three different intervals where the rest of the qualitative distances can vary. The second qualitative distance is the near distance where the center weight of the object A is on the point 0, and the center weight of the object B vary in the interval $]0,1]$. When the center weight of the object B surpasses the point 1; the qualitative distance then takes a medium value. In the medium qualitative distance, the center weight of object A still at point 0 and the center weight of the object B vary in the interval $]1,2]$. By surpassing the point 2 by the center weight of object B and till the point 3 or more, the qualitative distance is described as far. The center weight of object 1 still at point 0 and the center weight of the object B vary in the interval $]2,3]$.

4.4.4 A composite spatial relation between two objects

A composite spatial relation between two objects A and B is a relation composed of multiple types of spatial relations; it is used to enhance the information about a relative spatial description between a reference object A and another object B. In this abstract spatial description, a combination of the three previously mentioned spatial information is used to provide a full description of a spatial relation. Given two objects A and B, the composite spatial relation between these two objects is defined as follows:

R={Top,Dir,Dis} where:

- Top represents the qualitative topology according to RCC8 of the given two objects

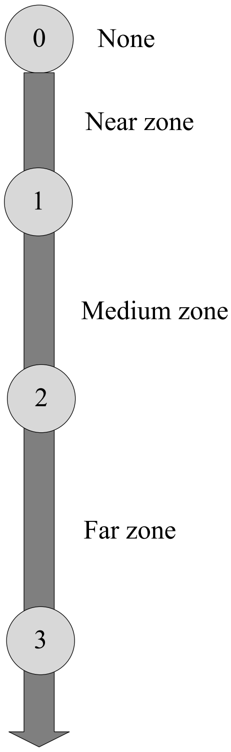


Figure 4.6: The qualitative distance relations

- Dir represents the qualitative direction of the center-weights of the two objects
- Dis represents the qualitative distance between the center-weights of the two objects

The next figure shows a sample representation of a spatial relation between two disjoint (according to RCC8) objects A and B.

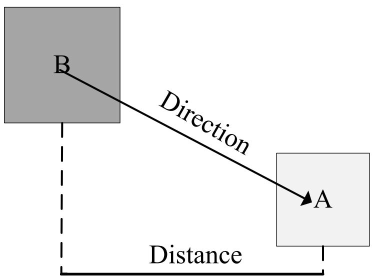


Figure 4.7: A composite spatial relation

4.4.5 An XML to PROLOG representation

In order to process spatial information available in an XMS document by our inference system, a transition from XML elements to PROLOG predicates is needed. The transition can be done while reading the XMS source file so the XML specification is translated into predicates to be processed by PROLOG later. In the next element, we provide how composite spatial relations are defined in our XMS document and how are they presented in our inference system.

In XMS, the composite spatial relations are represented according to the next portion of the XSD schema:

```
<xs:complexType name="spatialRelation">
  <xs:sequence>
    <xs:element name="source" type="xs:positiveInteger"/>
    <xs:element name="rccRelation" type="rccRelation"/>
    <xs:element name="distance" type="distanceType"/>
    <xs:element name="direction" type="directionType"/>
    <xs:element name="target" type="xs:positiveInteger"/>
  </xs:sequence>
</xs:complexType>
```

Figure 4.8: A composite spatial relation XMS schema

Where:

- source: defines the id of the source object.
- rccRelation: defines the topology of the two objects
- distance: defines the qualitative distance between the center-weights of the two objects
- direction: defines the qualitative direction of the center weight of the target object according to the center weight of the source object.
- target: defines the id the target object.

According to the previous schema the relation between object A and object B represented in figure (fig. 4.8) is defined in XMS as follows:

Given the id=1 for the object A and the id=2 for the object B, the XML portion that describe the composite spatial relation is shown in figure (fig. 4.9).

```
<spatialRelation>
      <source> 1 </source>
      <rccRelation> DC </rccRelation>
      <distance>Medium</distance>
      <direction>SE</direction>
      <target>2</target>
</spatialRelation>
```

Figure 4.9: A sample composite spatial relation in XMS

A composite spatial relation that is present in XMS like we mentioned previously is translated to a PROLOG predicate according to the form presented in figure (fig. 4.10).

```
spatialRelation( RelationId,
                 SourceObjectID,
                 RCC,
                 Direction,
                 Distance,
                 TargetObjectID ).
```

Figure 4.10: The form of a composite spatial relation in PROLOG

The previous example of the XML composite spatial relation is translated according to the previous figure into a PROLOG predicate show in figure (fig. 4.11).

```
spatialRelation( s1,
                 1,
                 dc,
                 se,
                 medium,
                 2          ).
```

Figure 4.11: A sample of a composite spatial relation in PROLOG

4.5 Spatial specification manipulation

A specification should be manipulated and adapted in order to make it suitable to a certain situation or profile; manipulations vary on several levels and according to several situations. The most used manipulation is to reduce either the spatial dimensions or temporal dimensions of a multimedia document so it fits and can be presented in a target device or situation. In the proposed approach, both reduction and stretching of a specification is considered in the inference system. Here we show how to reduce a spatial specification or stretch it to fit a certain profile.

According to the proposed spatial specification model, three main steps are used to either reduce or stretch the specification.

4.5.1 Distances refining

The first step in manipulating is the distance refining by either augmenting or reducing the distance depending of the situation. To do so, two ways browsing of the distances line allows us to achieve the needed transformation as shown in figure (fig. 4.12).

According to figure (fig. 4.11), we can count four main manipulations of distances in each direction. These manipulations are represented in PROLOG predicates in table (tab. 4.2). The type stop defines the end of either reducing or stretching a distance of a spatial relation.

In the case where after all the initial distances are reduced to a minimum 'none' or stretched to a maximum 'far', if the resulted specification does not match the profile then the adaptation of the spatial specification takes the next step.

Algorithm (alg. 1) is a simplified form of the spatial distances refining algorithm.It is based on the next basic steps:

- It starts by verifying the first spatial relation in the specification where the distance can be refined. This can be done by verifying if the result of refining the distance in the spatial relation is the same and no further

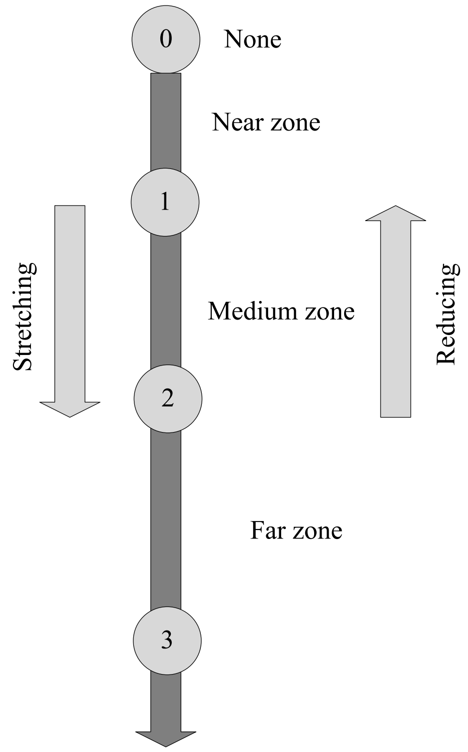


Figure 4.12: Distances manipulations

Table 4.2: Distances manipulation predicates

Type	**Predicate**
Stretching	**refineDistance**(stretch,none,near).
Stretching	**refineDistance**(stretch,near,medium).
Stretching	**refineDistance**(stretch,medium,far).
Stop	**refineDistance**(stretch,far,far).
Reducing	**refineDistance**(reduce,far,medium).
Reducing	**refineDistance**(reduce,medium,near).
Reducing	**refineDistance**(reduce,near,none).
Stop	**refineDistance**(reduce,none,none).

change can be done. The first most spatial relation that can be refined is then taken as r.

- r is then refined into a new spatial relation r'.
- Another part of the algorithm verifies if there are any affected distances in the specification and create a new spatial specification s', it contains

all non-refined relations and the refined ones, also all affected relations.

- Check if the new specification s' does in fact match the profile and delivers the specification and exit.
- In case the new specification does not match the profile, the algorithm then checks if all the relations in s' are refined to maximum and exit.
- If at least on spatial relation in s' can be refined the specification s' is then refined to a new one.

Algorithm 1 Distance refining

Input: profile p, document d, spatial specification s, adaptation direction dir
Output: adapted specification s'

$i \leftarrow 0$
$n \leftarrow$ number of relations in s
while refineDistance($dir, s[i](distance), s[i](distance)$) & $i < n$ **do**
  $i \leftarrow i{+}{+}$
end while
$distance = s[i](distance)$
refineDistance(dir,$distance$,$refinedDistance$)
$r' \leftarrow s[i]$
$r'(distance) \leftarrow refinedDistance$
generate the affected distances
$s' \leftarrow s$
$s'[i] = r'$
update all affected distances in s'
if match(p, s) **then**
  exit
else
  $j \leftarrow 0$
  $r \leftarrow s'[j]$
  while refineDistance($dir, s'[i](distance), s'[i](distance)$) & $j < n$ **do**
    $j \leftarrow j{+}{+}$
  end while
  if $j = n$ all distances are refined to maximum **then**
    exit
  else
    $s' \leftarrow$ refineDistances(p, s', dir)
  end if
end if

4.5.2 Directions rotation

If after refining all the distances to a maximum, the spatial specification still does not match the profile. The next step of the manipulation consists of rotating the different directions; either the width or the length of the specification is affected. The rotation of a directional relation is done according to either a clockwise direction or a counter-clockwise direction as shown in the figure (fig. 4.13).

Different possibilities when it comes to spatial directions, the final possible rotation of a given relation depends on the first state of the direction. We count from north to north a full rotation meanwhile; the adaptation requires only basic steps of rotation. According to the previous figure we count nine different basic possible rotation steps in each way (clockwise, or counter-clockwise) including the return step (the step that takes us back to the initial direction) and the non-rotation step where the initial objects have the same center weight.

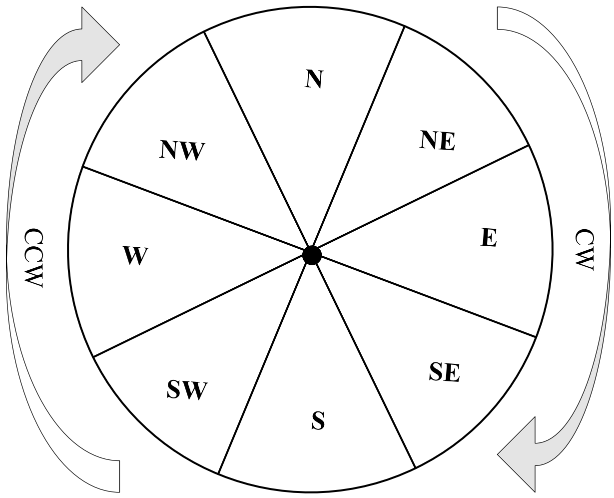


Figure 4.13: directional relations manipulations

The basic rotation steps are represented PROLOG predicates in tables (tab. 4.3,tab. 4.4) starting from the east direction. Table (tab. 4.3) for clockwise rotation and table (tab. 4.4) for clockwise rotation.The type stop defines the end of either clockwise or counter-clockwise rotation of the direction of a spatial relation.

Table 4.3: Clockwise directions manipulation predicates

Type	Predicate
Clockwise	**rotateDirection**(cw,e,se).
Clockwise	**rotateDirection**(cw,se,s).
Clockwise	**rotateDirection**(cw,s,sw).
Clockwise	**rotateDirection**(cw,sw,w).
Clockwise	**rotateDirection**(cw,w,nw).
Clockwise	**rotateDirection**(cw,nw,n).
Clockwise	**rotateDirection**(cw,n,ne).
Clockwise	**rotateDirection**(cw,ne,e).
Stop	**rotateDirection**(cw,none,none).

Table 4.4: Counter-clockwise directions manipulation predicates

Type	Predicate
stop	**rotateDirection**(ccw,none,none).
Counter-clockwise	**rotateDirection**(ccw,e,ne).
Counter-clockwise	**rotateDirection**(ccw,ne,n).
Counter-clockwise	**rotateDirection**(ccw,n,nw).
Counter-clockwise	**rotateDirection**(ccw,nw,w).
Counter-clockwise	**rotateDirection**(ccw,w,sw).
Counter-clockwise	**rotateDirection**(ccw,sw,s).
Counter-clockwise	**rotateDirection**(ccw,s,se).
Counter-clockwise	**rotateDirection**(ccw,se,e).

Algorithm (alg. 2) like algorithm (alg. 1) is a simplified form of the spatial direction rotation algorithm.It is based on the following basic steps:

- It starts by verifying the first spatial relation in the specification where the direction can be rotated. this can be done by verifying if the result of rotation of the distance in a spatial relation is the same and no further change can be done. The cases where no further change can be done are the predicate having the type "STOP" in tables (tab. 4.3, tab. 4.4). The

Algorithm 2 Directions rotation

Input: profile p, document d, spatial specification s, adaptation direction dir
Output: adapted specification s'

```
i ← 0
n ← number of relations in s
while rotateDirection(dir,s[i](direction),s'[i](direction)) & i < n do
   i ← i++
end while
r ← s[i]
direction ← r(direction)
rotateDirection(dir,direction,rotatedDirection)
r' ← r
r'(direction) ← rotatedDirection
s' ← s
s'[i] ← r'
update all affected directions in s'
if match(p,s) then
   exit
else
   j ← 0
   while rotateDirection(dir,s'[j](direction),s'[j](direction)) & j < n do
      j ← j++
   end while
   if j = n all direction are rotated to maximum then
      exit
   else
      s' ← rotateDirections(p,s',dir)
   end if
end if
```

first most spatial relation that can be rotated is then taken as r.

- r is then rotated into a new spatial relation r'.
- Another part of the algorithm verifies if there are any affected directions in the specification and create a new spatial specification s', it contains all non-rotated relations and the rotated ones, also all affected relations.
- Check if the new specification s' does in fact match the profile, deliver the specification and exit.
- In case the new specification does not match the profile, the algorithm then checks if all the relations in s' are rotated to maximum and exit.

- If at least on spatial relation in s' can be rotated the specification s' is then refined to a new one.

4.5.3 Topological refining

In a final try to adapt the spatial specification and if all the initial directions are rotated to an end direction and the resulted specification does not match the profile the inference system takes the specification to a final step. It consists of adapting the topology of the spatial specification according a neighborhood graph [19]
Just like the distances line, the direction of browsing the neighborhood graph defines either stretching or reducing the space occupied by the two objects. The PROLOG representation of the RCC8 neighborhood graph requires complex clauses in order to define the conditions for the transition between each two nodes; it depends on the size of the source object compared to the target object in the following cases:

- Reducing from DC to EC and from EC to PO
- Reducing from PO to either: TPP, EQ or TPPi depending on the two objects of a spatial relation
- Stretching from NTPP to TPP
- Stretching from EQ to PO
- Stretching from NTPPi to TPPi
- Stretching from either TPP or TPPi to PO
- Stretching from PO to EC and from EC to DC

Table (tab. 4.5) shows predicates for topological reduction of a spatial relation, there are five different possible reductions explained as follows:

- From DC to EC without further conditions
- From EC to PO without further conditions

- From PO to TPP in the case of $O1$ occupies lesser space than $O2$
- From PO to TPPi in the case of $O2$ occupies lesser space than $O1$
- From PO to EQ in the case of $O1$ occupies the same space as $O2$

Table 4.5: Topological reducing predicates

Type	Predicate
Reduce	**refineTopology**(dc,ec,_,reduce).
Reduce	**refineTopology**(ec,po,_,reduce).
Reduce	**refineTopology**(po,tpp,S,reduce):- **spatialRelation**(S,O1,po,_,_,O2), **spatialProperty**(_,O1,[MinH1,MinW1],_), **spatialProperty**(_,O2,[MinH2,MinW2],_), MinH1<MinH2,MinW1<MinW2.
Reduce	**refineTopology**(po,tppi,S,reduce):- **spatialRelation**(S,O1,po,_,_,O2), **spatialProperty**(_,O1,[MinH1,MinW1],_), **spatialProperty**(_,O2,[MinH2,MinW2],_), MinH1>MinH2,MinW1>MinW2.
Reduce	**refineTopology**(po,eq,S,reduce):- **spatialRelation**(S,O1,po,_,_,O2), **spatialProperty**(_,O1,[MinH1,MinW1],_), **spatialProperty**(_,O2,[MinH2,MinW2],_), MinH1=MinH2,MinW1=MinW2.

Table (tab. 4.6) shows the predicates where further reducing of the space occupied by two objects using topology refining, the following cases are the equivalent of the predicates:

- The topology is PO (partially overlapped) and neither object A nor object B is bigger than the other.
- The topology is either: TPP, TPPi, EQ, NTPP, NTPPi.

Stretching according to the neighborhoud graph (fig. 4.14) is simpler than reducing. Table (tab. 4.7) shows the following stretching predicates:

- From NTPP to TPP

Table 4.6: Topological stopping for when reducing predicates

Type	Predicate
Stop	**refineTopology**(po,po,S,reduce):- **spatialRelation**(S,O1,po,_,_,O2), **spatialProperty**(_,O1,[MinH1,MinW1],_), **spatialProperty**(_,O2,[MinH2,MinW2],_), ((MinH1<MinH2,MinW1>MinW2); (MinH1>MinH2,MinW1<MinW2)).
Stop	**refineTopology**(tpp,tpp,_,reduce).
Stop	**refineTopology**(tppi,tppi,_,reduce).
Stop	**refineTopology**(eq,eq,_,reduce).
Stop	**refineTopology**(ntpp,ntpp,_,reduce).
Stop	**refineTopology**(ntppi,ntppi,_,reduce).

- From NTPPi to TPPi
- From EQ to PO
- From TPP to PO
- From TPPi to PO
- From PO to EC
- From EC to DC

Table (tab. 4.7) also shows where no further stretching can be performed according to topology refining, when stretching the DC topology (disjoint) can no longer be stretched by changing the topology.

The algorithm (alg. 3) is a simplified form of the spatial topology refining algorithm.

Algorithm (alg. 3) is just like algorithms (alg. 1 and alg. 2) a simplified form of the spatial topology refining algorithm.It is based on the next basic steps:

- It starts by verifying the first spatial relation in the specification where the topology can be refined. This can be done by verifying if the result of refining of the topology in a spatial relation is a stop and no further

Table 4.7: Topological stretching predicates

Type	Predicate
Stretch	**refineTopology**(ntpp,tpp,_,stretch).
Stretch	**refineTopology**(ntppi,tppi,_,stretch).
Stretch	**refineTopology**(eq,po,_,stretch).
Stretch	**refineTopology**(tpp,po,_,stretch).
Stretch	**refineTopology**(tppi,po,_,stretch).
Stretch	**refineTopology**(po,ec,_,stretch).
Stretch	**refineTopology**(ec,dc,_,stretch).
Stop	**refineTopology**(dc,dc,_,stretch).

reduction or stretching can be done. The cases where no further change can be done are the predicate having the type "STOP" in tables (tab. 4.6, tab. 4.7). The first most spatial relation that can be topologically refined is then taken as r.

- r is then topologically refined into a new spatial relation r'.
- Another part of the algorithm verifies if there are any affected topologies in the specification and create a new spatial specification s' which contains all non-refined relations and the refined ones, also all affected relations.
- Check if the new specification s' does in fact match the profile, deliver the specification and exit.
- In case the new specification does not match the profile, the algorithm then checks if all the relations in s' are refined to maximum and exit.
- If at least on spatial relation in s' can be further refined the specification s' is then refined to a new one.

The figure (fig. 4.14) represents a modified RCC8 neighborhood graph used to refine the topology of a spatial specification. The dashed ends of the edges in

Algorithm 3 Topology refining

Input: profile p, document d, spatial specification s, adaptation direction dir
Output: adapted specification s'

```
i ← 0
n ← number of relations in s
while refineTopology(s[i](topology), s[i](topology), dir)) & i < n do
    i ← i++
end while
topology = s[i](topology)
refineTopology(dir,topology,refinedTopology)
r' ← s[i]
r'(topology) ← refinedTopology
generate the affected topology
s' ← s
s'[i] = r'
update all affected topologies in s'
if match(p, s) then
    exit
else
    j ← 0
    r ← s'[j]
    while refineTopology(s'[i](topology), s'[i](topology),) & j < n do
        j ← j++
    end while
    if j = n all topologies are refined to maximum then
        exit
    else
        s' ← refineTopologies(p, s', dir)
    end if
end if
```

the figure (fig. 4.14) are modified in the original neighborhoud graph [19], they mean we can not go in that direction. The cases where topological adaptation can do nothing are explained previously in the 'STOP' types of predicates.

4.6 Spatial adaptation algorithms

Previously, some basic manipulations over a spatial specification is discussed in algorithms (alg. 1, alg. 2 and alg. 3), the main algorithm (alg. 4) is an assembly of the previous algorithms.

The algorithm passes the initial spatial specification on a three-stage refining on the spatial dimension, these stages are based on algorithms (alg. 1, alg. 2

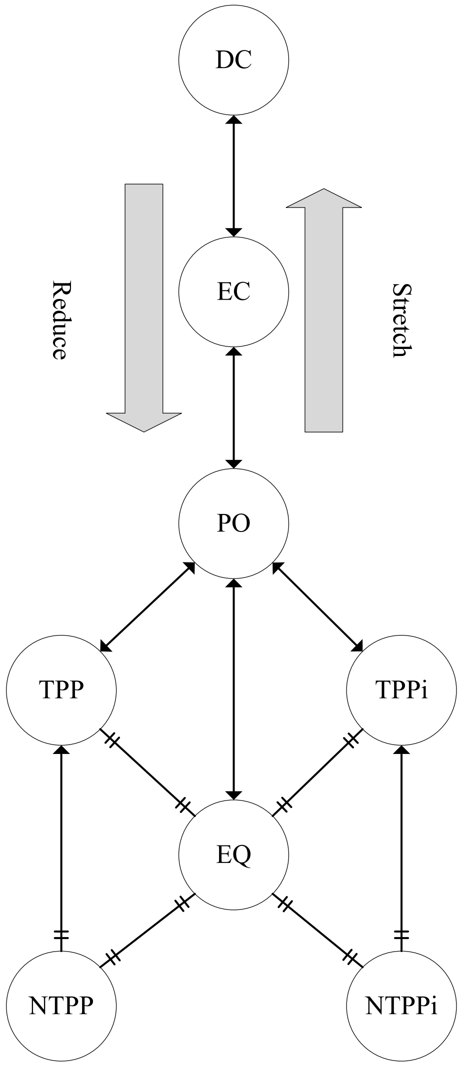


Figure 4.14: Modified RCC8 [1, 2] neighborhoud graph

and alg. 3) for spatial manipulation. Algorithm (alg. 4) is a simplified version of the adaptation algorithm. The algorithm consists of these main stages:

- The first stage consist of verifying if the spatial specification is already conform to the profile and exits the algorithm in that case. Otherwise, the algorithm passes to the second stage.

- In the second stage the algorithm refines the distances in the spatial specification in order either to reduce or stretch the space occupied by the final percept.

Definition 4.4. A percept according to the Oxford dictionary is a mental

Algorithm 4 Main adaptation algorithm

Input: profile p, document d, spatial specification s, adaptation direction dir
Output: adapted specification s'

```
if match(p,s) then
  exit
else
  s' ← refineDistances(p,s,dir)
  if match(p,s') then
    exit
  else
    s' ← rotateDirections(p,s,dir,0)
    if match(p,s') then
      exit
    else
      s' ← refineTopologies(p,s',dir)
      if match(p,s') then
        exit
      end if
    end if
  end if
end if
delivers the most refined version or take further actions and exit
```

concept that is developed as a consequence of the process of perception. In our abstract document, a percept is the final composition of a set of media elements in space and time; this composition is the one that the final user is going to perceive. The original document specification is the target percept, which the author intends originally to deliver to the final user. After applying the adaptation algorithm, the result might or might not achieve a consistent composition in space and time that fits the constraints of the final user. This leads to create something in between that neither give the specification as defined by the author nor respect all the constraints of the user and delivers a consistent composition that is called a percept.

- In case after refining distances the resulted specification s' matches the profile, s' is simply delivered. Otherwise the algorithm takes thee adaptation to the next level.

- If refining the distances would not produce a fairly matched specification, the specification s' resulted from previous step is passed through the directions rotation.
- In case after rotating the directions the resulted specification s' matches the profile, s' is then delivered. Otherwise the algorithms passes to the final try to adapt s'.
- The final stage consists of adapting the topology of the spatial specification s' resulted in the previous stage.
- If the resulted specification of topology adaptation s' does not match the profile, either the final spatial specification s' is delivered or according to the final user consent from the profile the original specification is delivered instead.

Advanced mechanisms might be used after all the tries of adapting of the spatial specification. Some mechanisms to refine a specification further including and not limited to: temporal decomposition, hypermedia decomposition as if the specification is split into two or more slides where every slide matches the spatial requirements [6].

4.6.1 A simple profile for adaptation

In order to adapt a multimedia document a reasoning about the document and the situation in which the document will be used is required. To do so, a profile must be used to get information about the device, location and some user preferences [57][58][59][60]. In this work, we used a simple profile that is translated from XML to PROLOG. The profile to be used in the inference system defines the screen, battery, and memory information of the target device so the system can decide what transformations and manipulations can be done on the specification in order to match the profile. Further information are required in a real life scenario, several studies have tried to define a profile like the well known CC/PP standard [59], others based on this standard or tried to suppress its limitations

The next set of predicates defines a simple profile:

```
profile(profileName).
device(profileName,screenID,memoryID,batteryID).
screen(screenID,physicalSizeInches,widthPixel,
        heightPixel,pixelDensity).
battery(batteryID,batteryPercentage).
memory(memoryID,availableMemory).
```

Figure 4.15: A simple profile PROLOG predicates

A sample smart-phone profile should look like:

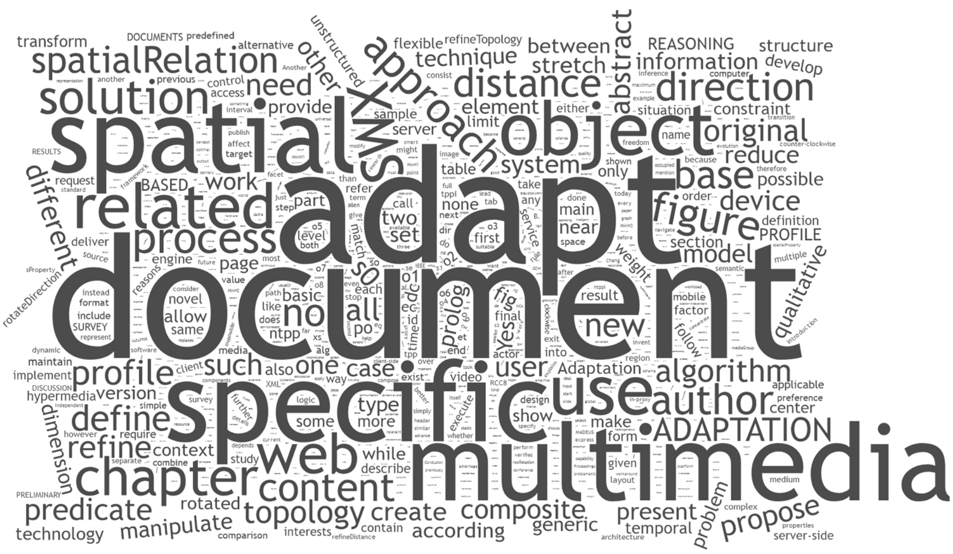
```
profile(smartPhone).
device(smartPhone,s1,m1,b1).
screen(s1,4.8,360,640,2).
battery(b1,20).
memory(m1,60).
```

Figure 4.16: A simple smart-phone profile PROLOG predicates

The goal of this chapter is to provide a practical way to implement the facet proposed in chapter three. An abstract document model was used to form a facet of the web without interfering with the original existing content. A spatial manipulation paradigm in an abstract document processing model was proposed to manipulate and adapt the abstract document according to a simple profile. Manipulation algorithms were given for the processing model, based on a combination of spatial information provided in the abstract version.

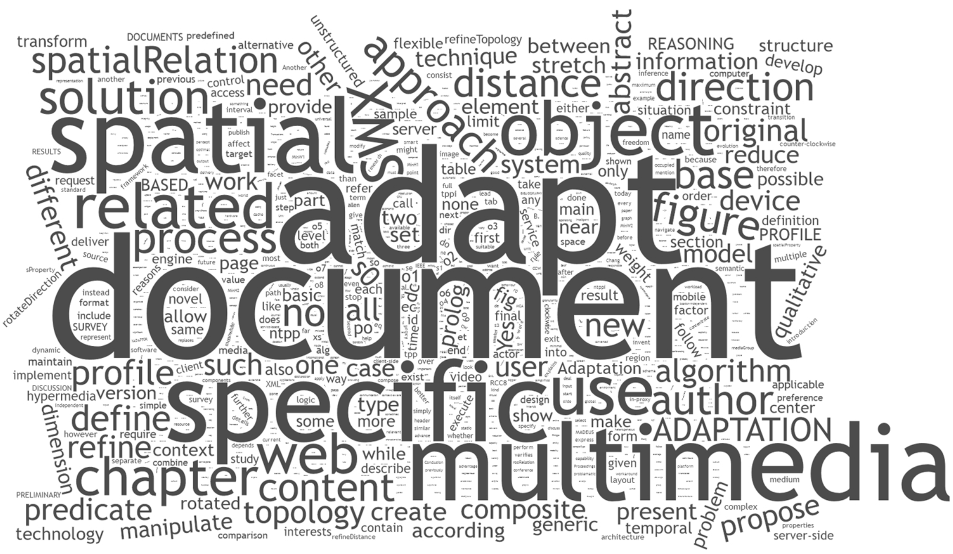
The spatial manipulation based on the composite spatial relations is limited in term of adaptation, the real problems under the adaptation are more of a spatio-temporal and hypermedia related. Only the spatial dimension was taken into account to showcase the approach, a more complex and delicate processing model would combine spatial, temporal and also hypermedia information both in the specification and the manipulation algorithms. The following chapter provide basic results of the manipulation algorithm.

The following chapter provides preliminary results of the manipulation algorithms presented in this chapter. Due to the complexity of the algorithms, only a unique step is shown in order to simplify the understanding of the manipulation paradigm.



End

Chapter 4;



Begin

Chapter 5;

CHAPTER 5

PRELIMINARY RESULTS AND DISCUSSION

A document processing model is a vital part of any adaptation system. In our approach based on XMS abstract documents, we rely on a novel processing model based on an abstract document that replaces the original document in the processing engine. XMS document processing model takes as input in the first phase of the adaptation an XMS document instead of the original document, this is what gives our processing model its power. The processing model as introduced in section (sec. 4.3) comprises in two phases of processing:

- Analysing the XMS abstract document.
- Transforming the original document based on the new version of the XMS abstract document.

In this chapter, we present some preliminary results using our adaptation PROLOG transformation engine and manipulation paradigm as a vital part to the XMS processing model. This chapter is organised as follows.

Section one presents the components of the XMS processing model. Section two shows a sample abstract web page layout to be used in our processing engine, section three presents the corresponding XML and PROLOG specification of the sample abstract layout. Section four shows some sample results of applying algorithms on the spatial dimension of the sample document. Section five compares XMS to other existing solutions and concludes the chapter.

5.1 Components of the XMS processing model

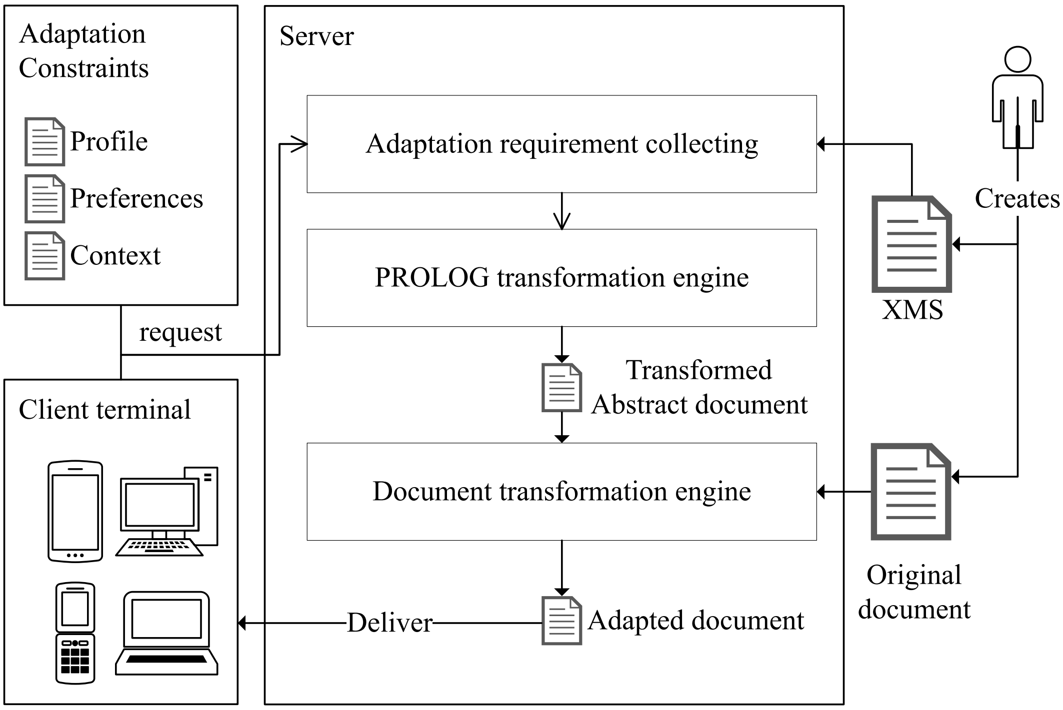


Figure 5.1: XMS processing model schema

Figure (fig. 5.1) shows a detailed version of the XMS processing model shown in figure (fig. 4.1). The schema shows the life cycle of the request and delivery of a document in a client-server architecture.

The first phase of the life cycle of the document begins when the client requests a document from a server, the request comes along with a set of adaptation constraints comprises in a profile, preferences and a context. A process takes care of collecting adaptation requirement from the constraints along with the abstract XMS document and passes them to the PROLOG transformation engine.

The PROLOG transformation engine takes as input a PROLOG version of the abstract XMS document and the adaptation constraints. The engine processes and manipulate the specification accordingly, generates a new version of the abstract XMS document as a set of Prolog predicates. The engine then passes new version of the abstract XMS document along with the original document to the document transformation engine.

The document transformation engine transforms the original document according to the adapted abstract version of XMS document and delivers the adapted final document to the client.

An important part of this thesis focuses on the design and implementation of the PROLOG transformation engine. The engine is based on the spatial manipulation algorithms as explained in section (sec. 4.4), samples of how the manipulation algorithms work are shown in the following sections.

5.2 The sample document

In figure (fig. 5.2) a sample abstract layout of a web page is shown, this layout can be extended to a different layout with more details and therefore giving its specification a higher granularity (see def. 4.3. To keep the example simple for understanding we chose only to keep the outer layout, the following elements presents samples of applying the spatial manipulation algorithms on the layout.

5.3 The XMS specification

In our approach the document author starts by writing or generating a specification of the original document using XMS, a sample specification corresponding to the simple page layout shown in figure (fig. 5.2) is composed of three main main elements:

- The XMS header containing general information about the document and possible alternatives.
- The objects declaration containing references and details about objects in the original document.

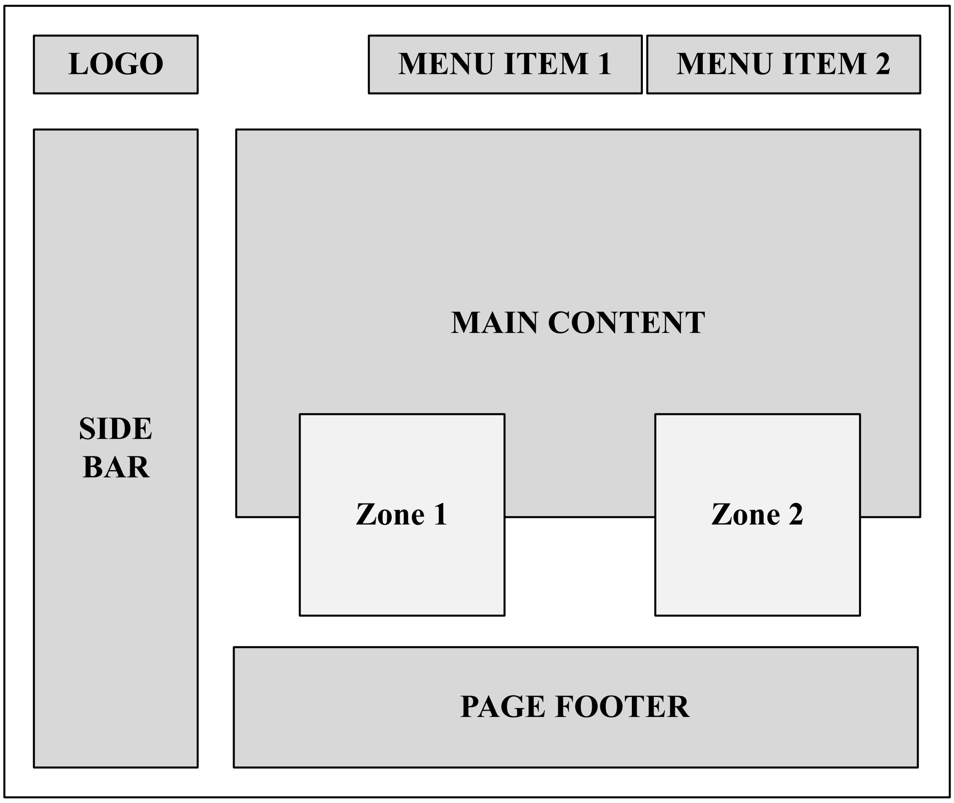


Figure 5.2: A simple web page layout

- The specification part containing both spatial properties and relations.

5.3.1 The XML specification

The header of the XMS specification of the simple page layout is shown in figure (fig. 5.3). The header in this case contains only the target original document.

```
<header target="sample_web_page.xhtml">
</header>
```

Figure 5.3: XMS specification header

The object declaration part of the specification is shown in figure (fig. 5.4). The object declaration references objects in the original document, in this case using HTML identifiers $\#id$. The type attribute defines whether the object being defined is a simple object or a media group. A media group type means the object is a complex object that may contain other components such as multiple HTML elements embedded in a DIV element.

```
<objects>
    <object id="o1" type="mediaGroup" name="body">
        <path value="#o1"/>
    </object>
    <object id="o2" type="mediaGroup" name="logo">
        <path value="#o2"/>
    </object>
    <object id="o3" type="mediaGroup" name="menuItem1">
        <path value="#o3"/>
    </object>
    <object id="o4" type="mediaGroup" name="menuItem2">
        <path value="#o4"/>
    </object>
    <object id="o5" type="mediaGroup" name="sideBar">
        <path value="#o5"/>
    </object>
    <object id="o6" type="mediaGroup" name="mainContext">
        <path value="#o6"/>
    </object>
    <object id="o7" type="mediaGroup" name="zone1">
        <path value="#o7"/>
    </object>
    <object id="o8" type="mediaGroup" name="zone2">
        <path value="#o8"/>
    </object>
    <object id="o9" type="mediaGroup" name="pageFooter">
        <path value="#o9"/>
    </object>
</objects>
```

Figure 5.4: XMS specification object declaration

A sample of the spatial specification is shown in figure (fig. 5.5) both the spatial properties and the spatial relations are shown, eight spatial properties are used to describe the objects of the specification other than the reference object $o1$ which is the main body of the sample web page. The body is as flexible as the whole composition, it is used to better describe positions of different elements.

For the spatial relations, we have found that twenty one spatial relation can describe fairly the sample web page. Not all properties and relations are shown in the figure for ease of understanding.

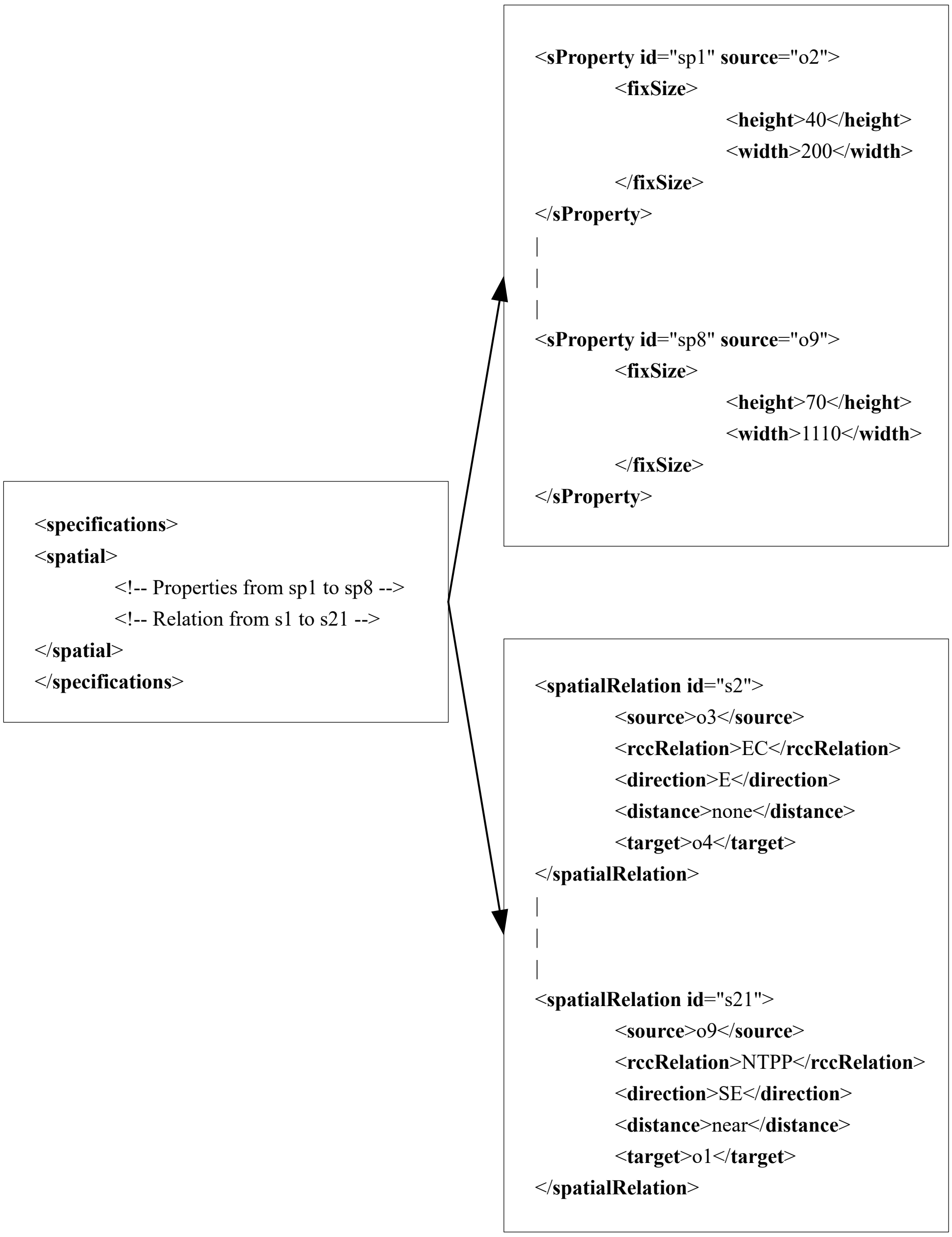


Figure 5.5: XMS spatial specification sample

5.3.2 Prolog representation

In order to process the XMS specification, a minimal version of the specification is used as a set of PROLOG predicates (facts). Just like XMS specification, same main elements exists in PROLOG predicates as follows:

- The XMS main predicate containing general information about the document, possible alternatives are defined in other predicates.

- The objects declaration containing references and other details are represented in a set of predicates.
- The specification part containing both spatial properties and relations is represented using a set of predicates.

The main predicate describing the document is shown in figure (fig. 5.6), the third term 'original' is used to describe the specification version. In case alternatives are given to the entire document, such as a different specification for a different profile is given or an adapted version of the specification is generated this term is named accordingly.

```
document(
        document1,
        d1,
        original
                ).
```

Figure 5.6: XMS main PROLOG predicate

The entire XMS specification in PROLOG predicates is shown in figure (fig. 5.7), except the spatial relations were simplified for ease of presentation in part (d) of the figure. Part (a) shows the main predicate where the entire specification is defined, part (b) shows the definition of the set of predicates which defines objects of XMS specification. Part (c) shows the spatial properties according to the XMS specification for each of the objects, except the reference object which is the 'body' object.

5.4 Sample results

In this part we show some sample results using a unique iteration of each manipulation algorithm presented in the previous chapter (alg. 1, alg. 2,alg. 3). Figure (fig. 5.8) shows the original relations before any manipulations.

Figure (fig. 5.9) shows a sample query of the distance refining algorithm (alg. 1). The resulted new document specification $d1refined$ contains the set of new refined relations, we concatenated the original document ID with the

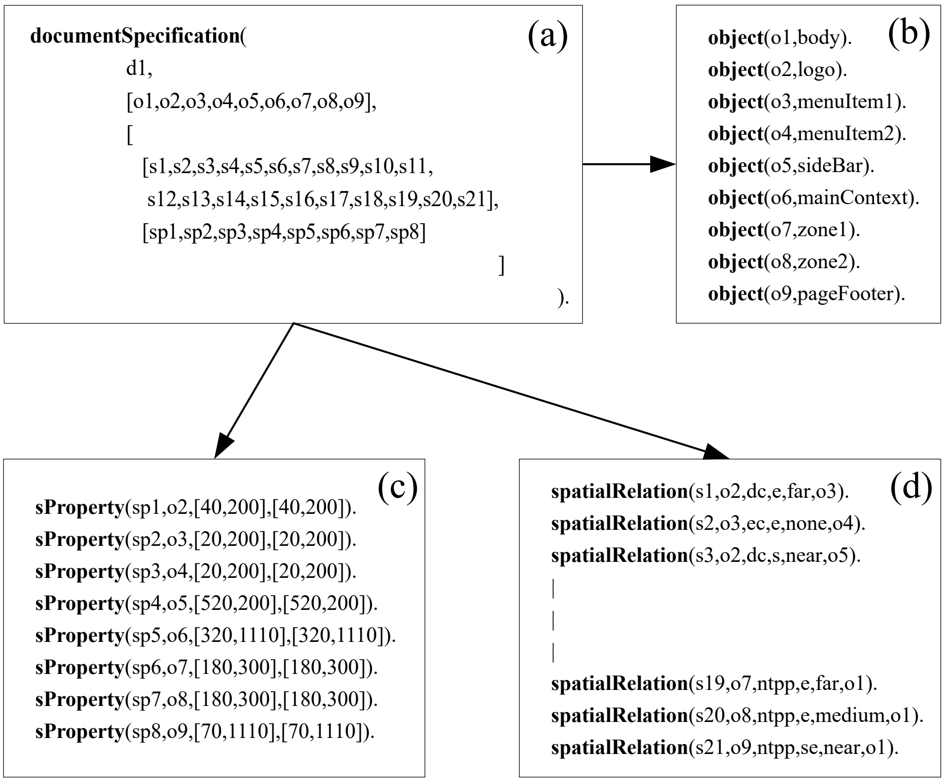


Figure 5.7: XMS PROLOG specification

```
spatialRelation(s1, o2, dc, e, far, o3).
spatialRelation(s2, o3, ec, e, none, o4).
spatialRelation(s3, o2, dc, s, near, o5).
spatialRelation(s4, o3, dc, s, near, o6).
spatialRelation(s5, o4, dc, s, near, o6).
spatialRelation(s6, o5, dc, e, near, o6).
spatialRelation(s7, o6, po, s, near, o7).
spatialRelation(s8, o6, po, s, near, o8).
spatialRelation(s9, o7, dc, e, near, o8).
spatialRelation(s10, o5, dc, e, medium, o7).
spatialRelation(s11, o5, dc, e, near, o9).
spatialRelation(s12, o7, dc, s, near, o9).
spatialRelation(s13, o8, dc, s, near, o9).
spatialRelation(s14, o2, ntpp, nw, near, o1).
spatialRelation(s15, o3, ntpp, n, near, o1).
spatialRelation(s16, o4, ntpp, ne, near, o1).
spatialRelation(s17, o5, ntpp, sw, near, o1).
spatialRelation(s18, o6, ntpp, e, near, o1).
spatialRelation(s19, o7, ntpp, e, far, o1).
spatialRelation(s20, o8, ntpp, e, medium, o1).
spatialRelation(s21, o9, ntpp, se, near, o1).
```

Figure 5.8: Original spatial relations before any manipulations

term 'refined' to differentiate the original and the refined specification. Figure (fig. 5.12) shows the new set of refined relations in the refined specification, the profile 'smartPhone' used in the query is presented in the knowledge base and can refer to the sample profile presented in figure (fig. 4.16).

```
?- time(perceptDistancesRefine(reduce,d1,smartPhone,X)).
% 220 inferences, 0.000 CPU in 0.002 seconds (?% CPU, Infinite Lips)
X = d1refined.
```

Figure 5.9: Distance refining predicate call in PROLOG

```
?- time(perceptDirectionsRotation(cw,d1,smartPhone,X)).
% 409 inferences, 0.000 CPU in 0.000 seconds (?% CPU, Infinite Lips)
X = d1rotated .
```

Figure 5.10: Direction rotation predicate call in PROLOG

```
spatialRelation(s1rotated, o2, dc, se, far, o3).
spatialRelation(s2affected, o2, ec, e, none, o4).
```

Figure 5.11: Spatial relations rotated or affected after direction rotation call

Figure (fig. 5.10) shows a sample query of the direction rotation algorithm (alg. 2). The resulted new document specification "$d1rotated$" contains the set of new rotated and affected relations, we concatenated the original document ID with the term 'rotated' to differentiate the original and the rotated specification. Figure (fig. 5.11) shows the new set of relations in the resulted specification. Not all relations are shown, only the rotated and the affected relations are asserted in the system for better optimisation. In the query of the rotation algorithm predicate we used the original document specification, usually the rotation predicate is used after calling the distance refining algorithm predicate according to algorithm (alg. 4).

Figure (fig. 5.13) shows a sample query of the topology refining algorithm (alg. 2). The resulted new document specification $d1toporefine$ contains the

```
spatialRelation(s1refined, o2, dc, e, medium, o3).
spatialRelation(s2refined, o3, ec, e, none, o4).
spatialRelation(s3refined, o2, dc, s, none, o5).
spatialRelation(s4refined, o3, dc, s, none, o6).
spatialRelation(s5refined, o4, dc, s, none, o6).
spatialRelation(s6refined, o5, dc, e, none, o6).
spatialRelation(s7refined, o6, po, s, none, o7).
spatialRelation(s8refined, o6, po, s, none, o8).
spatialRelation(s9refined, o7, dc, e, none, o8).
spatialRelation(s10refined, o5, dc, e, near, o7).
spatialRelation(s11refined, o5, dc, e, none, o9).
spatialRelation(s12refined, o7, dc, s, none, o9).
spatialRelation(s13refined, o8, dc, s, none, o9).
spatialRelation(s14refined, o2, ntpp, nw, none, o1).
spatialRelation(s15refined, o3, ntpp, n, none, o1).
spatialRelation(s16refined, o4, ntpp, ne, none, o1).
spatialRelation(s17refined, o5, ntpp, sw, none, o1).
spatialRelation(s18refined, o6, ntpp, e, none, o1).
spatialRelation(s19refined, o7, ntpp, e, medium, o1).
spatialRelation(s20refined, o8, ntpp, e, near, o1).
spatialRelation(s21refined, o9, ntpp, se, none, o1).
```

Figure 5.12: Spatial relations after distance refining call

set of new refined relations, we concatenated the original document ID with the term 'toporefine' to differentiate the original and the refined specification. Figure (fig. 5.14) shows the new set of relations in the resulted specification. Not all relations are shown, only the refined relations are asserted in the system for better optimisation. Before calling the topology refining algorithm predicate we used the original document specification, as explained previously and according to algorithm (alg. 4) the topology refining predicate is used after calling the direction rotation predicate.

```
?- time(perceptTopologyRefine(reduce,d1,smartPhone,X)).
% 166 inferences, 0.000 CPU in 0.000 seconds (?% CPU, Infinite Lips)
X = d1toporefine .
```

Figure 5.13: Topology refining predicate call in PROLOG

```
spatialRelation(s1topo, o2, ec, e, far, o3).
spatialRelation(s2topo, o3, po, e, none, o4).
spatialRelation(s3topo, o2, ec, s, near, o5).
spatialRelation(s4topo, o3, ec, s, near, o6).
spatialRelation(s5topo, o4, ec, s, near, o6).
spatialRelation(s6topo, o5, ec, e, near, o6).
spatialRelation(s7topo, o6, tppi, s, near, o7).
spatialRelation(s8topo, o6, tppi, s, near, o8).
spatialRelation(s9topo, o7, ec, e, near, o8).
spatialRelation(s10topo, o5, ec, e, medium, o7).
spatialRelation(s11topo, o5, ec, e, near, o9).
spatialRelation(s12topo, o7, ec, s, near, o9).
spatialRelation(s13topo, o8, ec, s, near, o9).
```

Figure 5.14: Spatial relations refined after topology refining call

5.5 XMS in comparison with existing solutions

The most closely related work is that of Euzenat et al. [19], where the authors take a similar approach by designing a new intermediate document called MADEUS. They use this document as a canal between a high-level XML-based multimedia document (source document) and a low-level rendering format. Unlike XMS, MADEUS does not only describe the original document. Instead, it is a whole transition that contains both content and specification. The main inconvenience to this is the level of separation. By using MADEUS, there will be no separation between the content and the adaptation. Meanwhile, XMS takes the original document out of the equation. Also, the fact that MADEUS is not based on the higher level standardized multimedia XML-based document do not allow it to go further and become a generalized solution for multimedia adaptation in general.

Another related research is that of He et al. [13] where authors proposed a flexible content adaptation system using a rule-based approach called XADAPTOR. They develop an adaptation technique for structure object HTML table, the proposed adaptation system is based on the reorganization of table cells to adapt a content of an HTML document. This technique is more focused on only a specific part of HTML instead of a generalized solution for content adaptation.

The authors claim the solution is extensible by adding rules in the PROLOG based inference system. However unlike XMS, the separation between the content and adaptation is not addressed, thus limiting the applications of the techniques proposed by the authors.

The next table provide a comparison between XMS, MADEUS and XADAPTOR according to adaptation approaches factors discussed in section (sec: 4.1).

Table 5.1: Comparison between XMS, MADEUS and XADAPTOR

Factor	XMS	MADEUS	XADAPTOR
Abstraction	1-2	0	0
Workload	1-2	2	1
Separation	2	0	0
Flexibility	1-2	1	1
Author control	1-2	0	0

An extensive comparison between XMS and other existing solutions that tackle the problem of adaptation is presented in table (tab. 5.2), the compared solutions include the previously closely similar solutions to XMS (MADEUS, XADAPTOR) (see tab. 5.1).

The following values are used in the previous table:

- N: stands for "No"
- Y: stands for "Yes"
- N/A: stands for "Not Applicable"
- U: stands for "User" and means 'it depends on the final user'
- A: stands for "Author" and means 'it depends on the author of the document'
- P: stands for "Predefined" and means 'predefined on design time of the approach'

Table 5.2: Comparison between XMS and other similar solutions

Solution	XMS	MADEUS [19]	XADAPTOR [13]	WebViews [42]	AHM [63]	MPEG 21 [17]	SMIL [11]	AHA [64]	Cuypers [66]	MM4U [65]
Adaptation of object	N	N	N	N	N	Y	N	N	N	N
Adaptation of composition	Y	Y	Y	Y	N	N	Y	Y	Y	Y
Alternative specification	A	N	N	U	Y	Y	Y	N	N	N
Spatial specification	Y	Y	N	N	N	N	Y	N	Y	Y
Temporal specification	Y	Y	N	N	N	N	Y	N	Y	Y
Hypermedia specification	N	Y	N	N	Y	N	Y	Y	Y	N
Separation	Y	N	N/A	N	N	N/A	N	N/A	N	N
Transformation rules	Y	Y	Y	N	N	N	N	Y	N	N
Flexible document	A	N	N	N	N	N	N	N	Y	Y
Replacement	N	Y	N	Y	Y	Y	Y	N	Y	Y
For existing documents	Y	N	Y	Y	N	N	N	Y	N	N
Level of specification	A	P	P	P	P	P	P	P	P	P
Level of flexibility	A	P	P	P	P	P	P	P	P	P

The legend of the comparison table is on the previous page and the following criteria and questions that an adaptation solution should address are used:

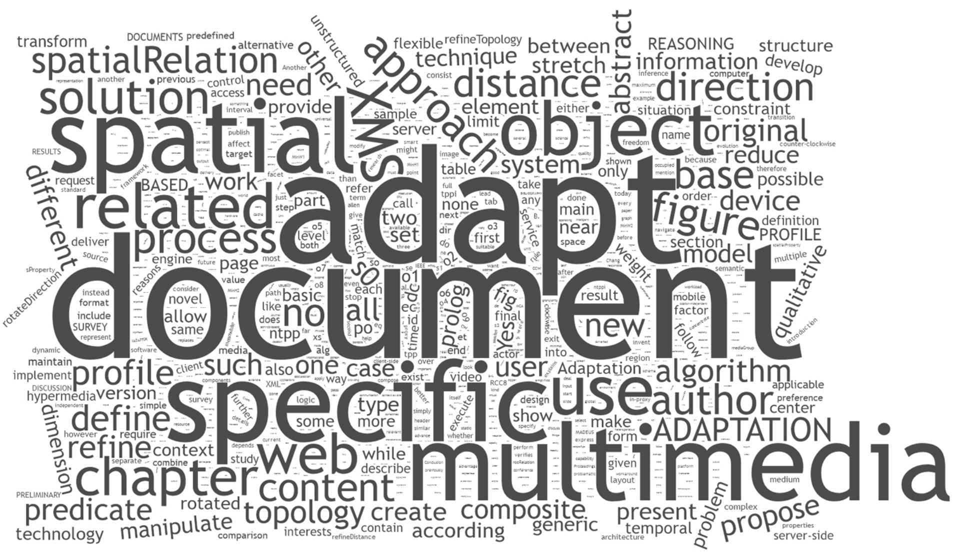
- Adaptation of object: this is a primary criteria in developing a solution for adaptation, the authors of the solution choose whether adapting single media elements is of concern for the solution under development or not.

- Adaptation of composition: same as adapting objects, the authors of an adaptation solution should choose if the solution require adapting the whole multimedia composition or simply media elements.
- Alternative specification: this criteria defines if the solution proposes a way to define some kind of alternatives whether to a particular object within the composition, or for the entire document. The 'author', 'user' values defines the case where the alternatives are defined by the author or the user.
- Spatial specification: this defines whether the solution proposes a way to specify the spatial dimension of the document.
- Temporal specification: this defines whether the solution proposes a way to specify the temporal dimension of the document.
- Hypermedia specification: this defines whether the solution proposes a way to specify the hypermedia dimension of the document.
- Separation: this criteria defines where applicable, if there is a separation between the content and the specification.
- Transformation rules: defines whether the solution proposes a way to define or has predefined rules for transforming either objects or the whole composition.
- Flexible document: defines whether the solution is based on a flexible document model, in the case of XMS 'author' means its up to the author to decide the level of flexibility of the specification.
- Replacement: defines whether the solution is based on a document that replaces the original one.

- For existing document: defines if the solution is used on existing documents, or is based on a specific new document defined as a part of the solution.
- Level of specification: the level of details included in the specification, XMS leaves this to the sole decision of the author of the document and the specification.
- Level of flexibility: the level of which the solution provide flexibility in the adaptation, in all the solutions the level is predefined except XMS where the flexibility is up to the author of the document.

In this chapter preliminary results of our manipulation paradigm and algorithms were provided. Due to the complexity of the adaptation approach, only a unique iteration of each algorithm was used to showcase the manipulation of the spatial dimension of the document. A further development of the spatial manipulation algorithms would show all potentials of the manipulation paradigm.

On another hand, combining other dimensions to the manipulation paradigm would lead to more interesting results in the adaptation. Other than that, the alternatives specification is not yet tested because of the simplicity of alternatives management algorithms. However, they can provide authors with more flexibility.



End

Chapter 5;

CHAPTER 6

CONCLUSION

Under the assumption that the most generic approach for adaptation is something possible, the aim of our work was to propose such an approach for profile based multimedia document adaptation. one that uses inference systems to modify a multimedia specification according to a given profile, we ended up by proposing a novel abstract document processing model. It is based on an abstract document called XMS that describes and accompany an original multimedia document, XMS is a replacement in the processing model to the original document and allows to generate a new adapted version using a set of manipulations to modify its spatial dimension.

We focused on the spatial dimension of a specification, a combination of spatial qualitative information were used to describe the spatial composition of the original multimedia document. Qualitative topologies, directions and distances were combined into a composite spatial relation to describe the relation between each two objects of a specification.

The key contributions of the approach based on XMS can be summed up as follows:

- The use of XMS as an abstract replacement of the original document in the processing model allowed us to achieve the goals set in the introduction, this also helped to maintain the separation between the original document and the adaptation system.

- The combination of multiple spatial qualitative information allows an accurate definition of the spatial layout; the proposed reasoning helps not only reduce the space occupied by the spatial specification but also extend it (or stretch) to fit a bigger output or simply a bigger percept.
- Three basic algorithms for manipulating distances, directions and topologies were proposed. The algorithms implemented in PROLOG forms a basis for the main algorithm of the adaptation, further work is needed and investigation about all the possibilities of possible manipulations that help achieve the goal by balancing limitations, authors intentions, and user preferences.
- The main algorithm assembles the basic algorithms, process, and test if the resulted specification matches a profile. In some cases, only basic distances refining may adapt the specification if the profile is for a situation not so much different than the original intended execution situation.
- The combined qualitative spatial information and the manipulation algorithms proposed in this thesis may have a broader applicability in other fields other than multimedia. Such fields may include and not limited to the following:
  - Visual tracking problems, further readings in [72][73][74].
  - Geography and geophysics problems, further readins in [2][70][71].
  - Pattern recognition problems, further readings in [75][76][77].

The reasoning in its current version does not exploit all the advantages provided in XMS, given that XMS also proposes ways to define limitations over the adaptation such as restrictions, alternatives, and also rule based alternatives.

Another limitation of the current version is it only processes the spatial dimension of the specification, a future work including temporal and even hypermedia dimensions is under study. A similar approach to spatial reasoning can be used on the temporal dimension using ALLEN interval logic rules [20][1][19].

One challenge is to provide a full framework for multimedia document adaptation, using both spatio-temporal and hypermedia reasoning combined adaptation based on the specification given by the author using XMS. Giving the author the full control on his document while allowing much of the flexibility while processing the specification through XMS.

Bibliography

[1] Alfonso Gerevini and Bernhard Nebel.
Qualitative spatio-temporal reasoning with rcc-8 and allen's interval calculus: Computational complexity.
In *Proceedings of the 15th European conference on artificial intelligence*, pages 312–316. IOS Press, 2002.

[2] Christopher B Jones, Ross Purves, Anne Ruas, Mark Sanderson, Monika Sester, M Van Kreveld, and Robert Weibel.
Spatial information retrieval and geographical ontologies an overview of the spirit project.
In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 387–388. ACM, 2002.

[3] Andrea Cavallaro.
Universal multimedia access.
In *Encyclopedia of Multimedia Technology and Networking*, pages 1001–1007. IGI Global, 2005.

[4] Yong Wang, Jae-Gon Kim, Shih-Fu Chang, and Hyung-Myung Kim.
Utility-based video adaptation for universal multimedia access (uma) and content-based utility function prediction for real-time video transcoding.
IEEE Transactions on Multimedia, 9(2):213–220, 2007.

[5] Anthony Vetro.
Mpeg-21 digital item adaptation: enabling universal multimedia access.

IEEE multimedia, 11(1):84–87, 2004.

[6] Habib Louafi, Stéphane Coulombe, and Umesh Chandra.
Robust qoe-aware prediction-based dynamic content adaptation framework applied to slides documents in mobile web conferencing.
Multimedia Tools and Applications, 74(18):7883–7920, 2015.

[7] Alti Adel, Sébastien Laborie, and Philippe Roose.
Semantic context-aware adaptation platform architecture.
Procedia Computer Science, 32:959–964, 2014.

[8] Dongsong Zhang, Anil Jangam, Lina Zhou, and Isil Yakut.
Context-aware multimedia content adaptation for mobile web.
International Journal of Networked and Distributed Computing, 3(1):1–10, 2015.

[9] N Saravanaselvam, K Rajangam, L Dhanam, and Anuradha Balasubramanian.
Mobile web browsing based on content preserving with reduced cost.
International Journal of Advanced Networking and Applications, 6(4): 2398, 2015.

[10] P Archana and Subhash S Kulkarni.
Meta-heuristic approach for resource optimization in mobile real time video traffic.
International Journal of Computer Applications, 110(14), 2015.

[11] Dick CA Bulterman and Lloyd W Rutledge.
SMIL 3.0: Flexible Multimedia for Web, Mobile Devices and Daisy Talking Books.
Springer Publishing Company, Incorporated, 2008.

[12] Nabil Layaïda.
Madeus: Système d'édition et de présentation de documents structurés multimédia.

PhD thesis, Université Joseph-Fourier-Grenoble I, 1997.

[13] J. He, T. Gao, W. Hao, I. l. Yen, and F. Bastani.
A flexible content adaptation system using a rule-based approach.
IEEE Transactions on Knowledge and Data Engineering, 19(1):127–140, Jan 2007.
ISSN 1041-4347.
doi: 10.1109/TKDE.2007.250590.

[14] Timothy J Ellis.
Multimedia enhanced educational products as a tool to promote critical thinking in adult students.
Journal of educational multimedia and hypermedia, 10(2):107–124, 2001.

[15] Qing Li, Jun Yang, and Yueting Zhuang.
Multimedia information retrieval at a crossroad.
In *Multimedia Technologies: Concepts, Methodologies, Tools, and Applications*, pages 242–249. IGI Global, 2008.

[16] Ian Burnett, Rik Van de Walle, Keith Hill, Jan Bormans, and Fernando Pereira.
Mpeg-21: goals and achievements.
IEEE MultiMedia, 10(4):60–70, 2003.

[17] Ian S Burnett, Fernando Pereira, Rik Van de Walle, and Rob Koenen.
The MPEG-21 book.
John Wiley & Sons, 2006.

[18] Chuck Musciano and Bill Kennedy.
HTML & XHTML: The Definitive Guide: The Definitive Guide.
" O'Reilly Media, Inc.", 2002.

[19] Jérôme Euzenat, Nabil Layaïda, and Victor Dias.
A semantic framework for multimedia document adaptation.

In *Proceedings of the 18th International Joint Conference on Artificial Intelligence IJCAI'2003*, pages 31–36, Acapulco, Mexico, August 2003. Morgan Kauffman.
URL `https://hal.inria.fr/inria-00423411`.

[20] James F Allen.
Maintaining knowledge about temporal intervals.
Communications of the ACM, 26(11):832–843, 1983.

[21] Jonathan Steuer.
Defining virtual reality: Dimensions determining telepresence.
Journal of communication, 42(4):73–93, 1992.

[22] Howard Rheingold.
Virtual reality: exploring the brave new technologies.
Simon & Schuster Adult Publishing Group, 1991.

[23] C Burdea Grigore and P Coiffet.
Virtual reality technology.
London: Wiley-Interscience, 1994.

[24] Larry Masinter, Tim Berners-Lee, and Roy T Fielding.
Uniform resource identifier (uri): Generic syntax.
2005.

[25] Ian S Graham.
The HTML sourcebook.
John Wiley & Sons, Inc., 1995.

[26] Steve DeRose, Eve Maler, David Orchard, and Ben Trafford.
Xml linking language (xlink).
Working Draft WD-xlink-20000221, World Wide Web Consortium (W3C), 2000.

[27] Danny Ayers and Max Völkel.
Cool uris for the semantic web.

Woking Draft. W3C, 2008.

[28] James Clark, Steve DeRose, et al.
Xml path language (xpath) version 1.0, 1999.

[29] Anders Berglund, Scott Boag, Don Chamberlin, Mary F Fernández, Michael Kay, Jonathan Robie, and Jérôme Siméon.
Xml path language (xpath).
World Wide Web Consortium (W3C), 2003.

[30] Zhenhui Yuan, Gheorghita Ghinea, and Gabriel-Miro Muntean.
Beyond multimedia adaptation: Quality of experience-aware multi-sensorial media delivery.
IEEE Transactions on Multimedia, 17(1):104–117, 2015.

[31] Sébastien Laborie, Jérôme Euzenat, and Nabil Layaïda.
Semantic adaptation of multimedia documents.
Multimedia tools and applications, 55(3):379–398, 2011.

[32] Davy Van Deursen, Wim Van Lancker, Sarah De Bruyne, Wesley De Neve, Erik Mannens, and Rik Van de Walle.
Format-independent and metadata-driven media resource adaptation using semantic web technologies.
Multimedia Systems, 16(2):85–104, 2010.

[33] Davy Van Deursen, Wim Van Lancker, Wesley De Neve, Tom Paridaens, Erik Mannens, and Rik Van de Walle.
Ninsuna: a fully integrated platform for format-independent multimedia content adaptation and delivery using semantic web technologies.
Multimedia Tools and Applications, 46(2-3):371–398, 2010.

[34] Rong-Hong Jan, Ching-Peng Lin, and Maw-Sheng Chern.
An optimization model for web content adaptation.
Computer Networks, 50(7):953–965, 2006.

[35] Wai Yip Lum and Francis CM Lau.

User-centric content negotiation for effective adaptation service in mobile computing.

IEEE Transactions on Software Engineering, 29(12):1100–1111, 2003.

[36] Rakesh Mohan, John R. Smith, and Chung-Sheng Li.

Adapting multimedia internet content for universal access.

IEEE Transactions on Multimedia, 1(1):104–114, 1999.

[37] Brian Noble, Morgan Price, and Mahadev Satyanarayanan.

A programming interface for application-aware adaptation in mobile computing.

Computing Systems, 8(4):345–363, 1995.

[38] Yong Zhang, ShenSheng Zhang, and SongQiao Han.

A new methodology of qos evaluation and service selection for ubiquitous computing.

In *International Conference on Wireless Algorithms, Systems, and Applications*, pages 69–80. Springer, 2006.

[39] Richard Han, Pravin Bhagwat, Richard LaMaire, Todd Mummert, Veronique Perret, and Jim Rubas.

Dynamic adaptation in an image transcoding proxy for mobile web browsing.

IEEE Personal Communications, 5(6):8–17, 1998.

[40] Yonghyun Hwang, Jihong Kim, and Eunkyong Seo.

Structure-aware web transcoding for mobile devices.

IEEE Internet computing, 7(5):14–21, 2003.

[41] Fumihiko Kitayama, S Hitose, Goh Kondoh, and Kazushi Kuse.

Design of a framework for dynamic content adaptation to web-enabled terminals and enterprise applications.

In *Software Engineering Conference, 1999.(APSEC'99) Proceedings. Sixth Asia Pacific*, pages 72–79. IEEE, 1999.

[42] Juliana Freire, Bharat Kumar, and Daniel Lieuwen.
Webviews: accessing personalized web content and services.
In *Proceedings of the 10th international conference on World Wide Web*, pages 576–586. ACM, 2001.

[43] Quang Pham Hai, Sébastien Laborie, and Philippe Roose.
On-the-fly multimedia document adaptation architecture.
Procedia Computer Science, 10:1188–1193, 2012.

[44] Alti Adel, Sébastien Laborie, and Philippe Roose.
Automatic adaptation of multimedia documents.
Procedia Computer Science, 19:992–997, 2013.

[45] Vanessa El-Khoury, David Coquil, Nadia Bennani, and Lionel Brunie.
Personalized video adaptation framework (piaf): high-level semantic adaptation.
Multimedia tools and applications, 70(2):1099–1140, 2014.

[46] M Shamim Hossain, Mehedi Masud, Abdulhameed A Alelaiwi, and Abdullah Alghamdhi.
Aco-based media content adaptation for e-learning environments.
In *Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 2014 IEEE International Conference on*, pages 118–123. IEEE, 2014.

[47] Jemal H Abawajy and F Fudzee.
Dynamic path determination policy for distributed multimedia content adaptation.
Multimedia tools and applications, 74(19):8325–8341, 2015.

[48] Abdelkrim Abdelli and Nadjib Badache.
Context-aware adaptation of multimedia documents for consistent presentations.
Multimedia systems, 17(6):465–486, 2011.

[49] Soocheol Lee, Seungmin Rho, and Jong Hyuk Park.
Multimedia contents adaptation by modality conversion with user preference in wireless network.
Journal of Network and Computer Applications, 37:25–32, 2014.

[50] Velibor Adzic, Hari Kalva, and Borko Furht.
A survey of multimedia content adaptation for mobile devices.
Multimedia Tools and Applications, 51(1):379–396, 2011.

[51] Mariam Kimiaei Asadi and J-C Dufourd.
Context-aware semantic adaptation of multimedia presentations.
In *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 362–365. IEEE, 2005.

[52] Gregory D Abowd, Anind K Dey, Peter J Brown, Nigel Davies, Mark Smith, and Pete Steggles.
Towards a better understanding of context and context-awareness.
In *International Symposium on Handheld and Ubiquitous Computing*, pages 304–307. Springer, 1999.

[53] Tim Berners-Lee.
Axioms of web architecture: Metadata.
Retrieved April, 8:2006, 1997.

[54] Zhenhui Yuan, Gheorghita Ghinea, and Gabriel-Miro Muntean.
Beyond multimedia adaptation: Quality of experience-aware multi-sensorial media delivery.
IEEE Transactions on Multimedia, 17(1):104–117, 2015.

[55] Rakesh Mohan, John R. Smith, and Chung-Sheng Li.
Adapting multimedia internet content for universal access.
IEEE Transactions on Multimedia, 1(1):104–114, 1999.

[56] Zoltan Katai and Laszlo Toth.

Technologically and artistically enhanced multi-sensory computer-programming education.
Teaching and teacher education, 26(2):244–251, 2010.

[57] Franklin Reynolds.
Composite capability/preference profiles (cc/pp): A user side framework for content negotiation.
W3C note, 1999.

[58] Cédric Dromzée, Sébastien Laborie, and Philippe Roose.
A semantic generic profile for multimedia documents adaptation.
Intelligent Multimedia Technologies for Networking Applications: Techniques and Tools, pages 225–246, 2013.

[59] Patrik Osbakk and Nick S Ryan.
Context, cc/pp, and p3p.
UbiComp 2002 Adjunct Proceedings, pages 9–10, 2002.

[60] Albert Held, Sven Buchholz, and Alexander Schill.
Modeling of context information for pervasive computing applications.
Proceedings of SCI, pages 167–180, 2002.

[61] Cédric Kiss.
Composite capability/preference profiles (cc/pp): Structure and vocabularies 2.0.
W3C Working Draft, 30, 2007.

[62] Zhijun Lei and Nicolas D Georganas.
Context-based media adaptation in pervasive computing.
In *Electrical and Computer Engineering, 2001. Canadian Conference on*, volume 2, pages 913–918. IEEE, 2001.

[63] Lynda Hardman, Dick CA Bulterman, and Guido Van Rossum.
The amsterdam hypermedia model: adding time and context to the dexter model.

Communications of the ACM, 37(2):50–62, 1994.

[64] Paul De Bra, David Smits, and Natalia Stash.
The design of aha!
In *Proceedings of the seventeenth conference on Hypertext and hypermedia*, pages 133–134. ACM, 2006.

[65] Ansgar Scherp and Susanne Boll.
Mm4u: A framework for creating personalized multimedia content.
In *Multimedia Technologies: Concepts, Methodologies, Tools, and Applications*, pages 325–363. IGI Global, 2008.

[66] JPTM Geurts, Jacco R van Ossenbruggen, and Hazel Lynda Hardman.
Application-specific constraints for multimedia presentation generation.
Centrum voor Wiskunde en Informatica, 2001.

[67] Alfonso Gerevini and Jochen Renz.
Combining topological and size information for spatial reasoning.
Artificial Intelligence, 137(1-2):1–42, 2002.

[68] Andrew U Frank.
Qualitative spatial reasoning about distances and directions in geographic space.
Journal of Visual Languages & Computing, 3(4):343–371, 1992.

[69] Jayant Sharma and Douglas M Flewelling.
Inferences from combined knowledge about topology and directions.
In *International Symposium on Spatial Databases*, pages 279–291. Springer, 1995.

[70] Andrew U Frank.
Qualitative spatial reasoning: Cardinal directions as an example.
International Journal of Geographical Information Science, 10(3):269–290, 1996.

[71] Jochen Renz.

Qualitative spatial reasoning with topological information.
Springer-Verlag, 2002.

[72] Xiangyuan Lan, Andy J Ma, and Pong C Yuen.
Multi-cue visual tracking using robust feature-level fusion based on joint sparse representation.
In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1194–1201, 2014.

[73] Xiangyuan Lan, Shengping Zhang, and Pong C Yuen.
Robust joint discriminative feature learning for visual tracking.
In *IJCAI*, pages 3403–3410, 2016.

[74] Xiangyuan Lan, Shengping Zhang, Pong C Yuen, and Rama Chellappa.
Learning common and feature-specific patterns: a novel multiple-sparse-representation-based tracker.
IEEE Transactions on Image Processing, 27(4):2022–2037, 2018.

[75] David C Van Essen, Charles H Anderson, and Daniel J Felleman.
Information processing in the primate visual system: an integrated systems perspective.
Science, 255(5043):419–423, 1992.

[76] Keh-Shih Chuang, Hong-Long Tzeng, Sharon Chen, Jay Wu, and Tzong-Jer Chen.
Fuzzy c-means clustering with spatial information for image segmentation.
computerized medical imaging and graphics, 30(1):9–15, 2006.

[77] P Lasch and D Naumann.
Ft-ir microspectroscopic imaging of human carcinoma thin sections based on pattern recognition techniques.
Cellular and molecular biology (Noisy-le-Grand, France), 44(1):189–202, 1998.

الحَمدُ لِلّهِ

الَّذي بِنعمَتِهِ تَتِمُّ الصَالِحَات