Masters Thesis

The Information- and Communication Technology
at Agder University College

Mobile learning
Technological challenges on multi-channel e-learning services

by

Knut Ola Topland

siving.hia.no

Grimstad, spring 2002

Abstract

The focus of this thesis is the challenge of developing multi-channel e-learning services on Internet in which the content is located at the same node. Multi-channel is here defined as the technology which is using a framework on how the content should be distributed such as web, wap, phone and fax. The term multi-channel e-learning service means an electronic learning service which is capable to provide the same content to a number of different devices using more than one media type such as text, sound, picture, video and data sent in both directions. The thesis concentrates on the technological architectures, the current official standards, mobile end-user devices, and how this technology can improve the learning processes.

Chapter one gives some basis and definitions in the range of the subject. Second chapter describes relevant e-learning theory. A describing and discussing of the multi-channel approach is in chapter three. Chapter four defines and discusses technological architectures and standards. A prototype which is based on central parts of the learning theory and technology architecture is presented in chapter five. In the final chapter a discussion is done.

Result from this thesis work shows that it might be reasonable to assume that multi-channel e-learning services can be developed by using current technological standards. By using the control content standards that negotiate device capabilities to the content server there is reasonable to conclude that methods can be made to control the quality of the distributed content. Hence, prototype tests have shown that older devices or browsers does not support content control standards which can give problems to realize multi-channel services that uses these content control methods. Content control can also help content editors to improve the learning outcome and utilize the service to increase the user's learning process.

Preface

This Thesis is part of the Master of Science (MSc) degree in Information and Communication Technology (ICT) at Agder University College. It is the closure on the education that leads to the Master of Science degree.

I would like to use this opportunity to give a special thanks to my main advisor Lars Line at Agder University College in Grimstad, Norway, and second advisor Kristian Folkman at Norwegian Trade Consul in San Francisco.

Thanks to Mark Butler at HP, Marcelo Milrad and Juan Perez at Framkom, and Eilif Trondsen at SRI for useful assistance thoroughly the thesis period.

Grimstad, May 2002

_____

Knut Ola Topland

Table of contents

Figures and Tables

1 Basis for this thesis

1.1 Introduction

E-learning is learning supported by computer technology. Often it is expressed as distance learning by using Internet technology as the carrier medium and a type of software as the learning medium. E-learning changed radically in the end of the 90s and there has been a major increase in both e-learning companies and e-learning departments in the biggest traditional Internet companies such as Oracle, Microsoft, and Cisco. Simultaneously there has been a great expansion of dissimilar mobile devices and many people are using these devices at work or in their spare-time. The e-learning industry has faced that instant learning services might have a great commercial potential and, therefore, has a strong focus on how e-learning services best can support them. Services such as instant learning involve the multi-channel approach because a user, who needs information there and then, should be able to use his/her device to receive. This involves a lot of technical challenges which are described more thoroughly in chapter 4.

The main challenge which I want to give an overview of in this part is the standardization approach. After the mobile Internet market increased, there has been an evolution of different e-learning architectures. Commercial companies as well as universities and organizations have faced the need to use standardized technologies in order to support most devices or other e-learning services.

The technologies that have been important to standardize include:
1. content specification, logic, distribution, and presentation
2. standardize interfaces between the e-learning services in order to support other e-learning services
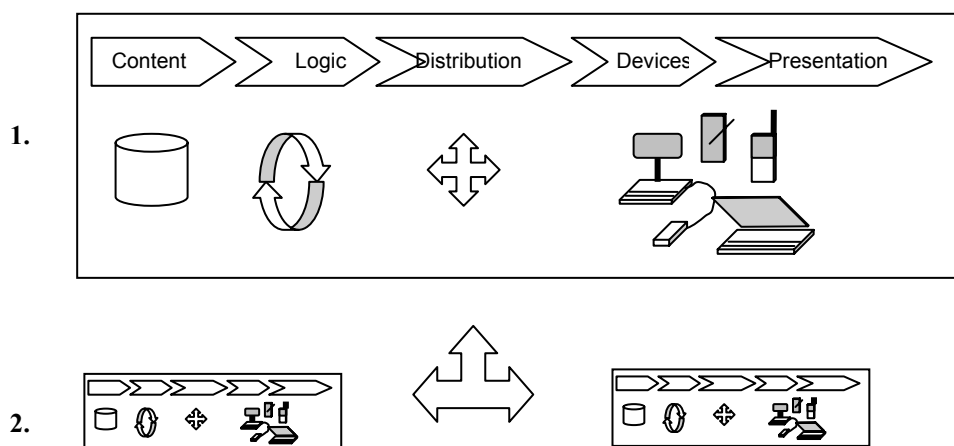
Figure 1 E-learning standardization fields

This thesis is not describing all standardized technologies, but is focusing on how content can be distributed with a high certainty that the information can be presented to most devices, is understandable to the user, and improve the learning process to the user.

E-learning multi-channel services might utilize and improve existing pedagogical methods and learning theories. Relevant theories to my approach are described in chapter 2. By using theories it makes a foundation how technology can be developed and used. The last chapter discusses improvements and challenges how technology can be utilized in order to improve the learner processes.

1.2 The assignment

Mobile learning, Technological challenges on multi-channel e-learning services.

As a result of the big expansion of Internet most learning institutions supply learning based services through the Internet. Currently we are facing a new evolution with the new generation of mobile network. This will result in more complex types of joint distribution channels and multiplication of terminals and this provide a wide array of new opportunities, challenges, and arenas in the field of e-learning. Today's basis requirement on "Just in time learning" (JiT) and "Learning on Demand" (LoD) can be easier realized with multi-channel e-learning services. An example is the craftsman who can be consulted by an expert-system on the spot. New and creative learning based services will appear and new services are going to supplement and have to be integrated with existing learning channels such as classrooms and networks. The challenge is to find out how one can effectively and with high quality produce multi-channel e-learning services.

The thesis will focus on challenges of developing of multi-channel service development in which all of the content is located at the same node. This paper will give an overview of multi-channel services, with emphasis on the use of these services in the field of learning. Products, technology, applications, and research of current interest will be presented. Also an application of multi-channel services and its technology will be researched. Furthermore this researcher is going to develop a prototype which will illustrate the challenges or weaknesses in multi-channel services.

1.3 Thesis focus and definition of terms

The focus on this thesis is how e-learning services can provide high-quality learning by using multi-channel services adjusted to an unknown number of end-user devices.

High quality learning content is broad term, but I will focus on the channel's ability to provide an understandable message as the creator of the content meant. A channel refers to the technologies that carry and present content. Multi-channel services include architectures which are capable to carry the same content to different channels. And learning comes into being when the content is understood by the user.

The content can consist of one or more media types such as text, sound, music, graphics, video, or data and a presentation specification. A major problem occurs when learning consists of one or more media types that the device cannot display. This may give incomprehensible meaning of the content. Therefore, I will define the different technologies in the content-storage, the intervening architecture, the devices, and the presentation.

1.4 Limitations

In this thesis I have done these limitations:
- I have a focus on multi-channel e-learning theories and technologies for the adult learners, especially in a work-situation.
- I am describing both wireless (mobile) terminals and wired (stationary) computers, but have the emphasis on the mobile terminals.
- Moreover I limit the network- or server-architectures, software, or other technical aspects to technologies that supports different devices.
- I limit the specification of devices to the commonly used devices
- Through the chapter of technology, I am describing the different technology architectures, but goes in depth on the Internet Client architecture.
- I am limiting to some markup-languages in order to do a more deeper description
- I am not discussing interaction-technologies

1.5 Methodology

This thesis was mainly based on studies of relevant books, online papers, and technical standardizations. Most e-learning theories were taken from books and online papers. The main sources of information in the chapter of technology originated from technical reports and online papers.

Participating at the E-learning Forum at SRI (Stanford Research Institute) and REN (Research E-learning Network) conference have informed me of what the e-learning industries are currently doing. Business meetings with mobile Internet companies in Silicon Valley have taught me about current technological innovations. E-mail correspondence and a paper from a researcher at the HP (Hewlett-Packard) have been helpful in the chapter of technology. Also researchers from Framkom have been useful to my prototype, especially the choice of architecture.

The prototype was developed by using standard computers and software. The final product was installed on a server at Agder University College.

2 Theory

2.1 Introduction

When researching and developing e-learning services, there is need to have basic knowledge of learning theory. Theory can be helpful to support one's technological assumptions and choices. This chapter expounds upon relevant learning and communication theories within the multi-channel e-learning services approach. Each sub-chapter includes discussions focused on mobile learning situations and summaries of the theoretical relevance for technical developing.

2.2 Matching media, content, channel with the user

In order to make communication or learning situation meaningful, sender and receiver need a mutual set of symbols. According to Gavriel Salomon [1] Symbol Systems consist of rules that make users able to make meaningful interpretations of content. Content can consist of languages, text, video, sound, colors, figures, and so on. The sender's and the receiver's (the user's) interpretation of the content has to correlate to make meaning.
The underlying symbol system has to be shared by both sender and receiver, and the medium must be capable of delivering it. Apart from the interpretation of the message more technical aspects might come into consideration as well. A slow connection in mobile situations can hinder the delivery and result in a diffuse video or slow synchronization between different media types.

Salomon explains that the medium comprises of four features; the technology, the content, the situation of use, and the use of the symbol system.

1. **The technology**. The technology refers to the distribution of the content, its availability and its accessibility, and the methods used to present it.
2. **The content**. Content is the information that consists of one or more media types. It is dependent on all the other four features.
3. **The situation of use** refers to the use of the medium. The learner may for instance expect another type of content in mobile situations than at a workplace.
4. **The symbol system** refers to the chosen set of symbols used to form the content. It depends on that the technology is capable to deliver and present the content.

These four features can be helpful when introducing new mediums or making new e-learning solutions. When the WAP technology was introduced on the mobile phones, the main problem was that it was a medium without any content. Additional explanations of failure are the price, lack of earnings potential to content suppliers, and a huge difference from its inherited medium. If we compare WAP with the I-MODE system in Japan, I-MODE was a success because it had content from the start accompanied by an earning model for the content supplier and much content caused by the HTML compatibility.

I want to focus on Salomon's third feature on how the same media works in different social contexts and situations. There is difference in outcome by using same media type in different media. A media type is the format of the content such as text, image, video, or sound. There is for instance an unequal need for multimedia applications on a computer than on a mobile phone. The situation, technology, and the user's financial expenses indicate that there is no

need for the user to watch half an hour of video on a mobile phone. Often there is a bigger request for short instructional clips that show exactly what the person needs to know. A shorter instruction followed by practice can make an increase in the quality of the learning process and a bigger motivation to use such services in other situations.

As an example of third feature, work process support seems to be a likely field of applying mobile learning solutions. The need corresponds with the strength of the media: short, just in time, performance improvement oriented content via handheld devices.

Salomon's fourth feature is focused on what mental processes required to digest different types of content. For instance, watching a video needs less mental processing than reading a book. Salomon also shows that there is less need for mental processing if the subject is well-known and represented by shared familiar symbol system for both sender and receiver. This points to the importance to use well known symbol systems if the content can be difficult to interpret and is meant to be distributed to an unknown number of users who have different skills and underlying symbol systems as a basis for the interpretations. Using audio visuals might even ease the learning process. The different mix of media types also can give a different outcome on the users learning effect. Salomon uses the example that text and sound are harder to perceive than by using text and image or sound and image. Furthermore, he says the user is using less mental processes to understand a message by using several media types than fewer media types. For instance, using text, image, and sound to instruct how to feed a lion, is better than using only sound.

The matching of a shared symbol system with contextual awareness of the user and user profiles is almost impossible. The number of individualized deliveries gets almost to infinity. A more practical approach is to map and limit the type of symbols used in communication with specific groups of users and adapt these to those of the receiver. The next step would logically be to utilize user profiles for delivering content in the most suitable media formats, making use of as many media formats to support the processing of the message on the user side. Finally, the underlying notion of how context influence learning should be taken into consideration when designing any type of learning program. I will not elaborate on specific principles of contextual design, but stress the need to carefully examine and take into consideration different aspects of both the cognition and influence of context on cognition when designing any type of learning content.

If it is feasible, the best way to make good e-learning solutions may be when you know the learner, the situation, and the browser facilities of the device. Then it is possible to specialize the content in such way that it uses a well-known symbol system that fits the user's skills.

Relevance for technical developing:

    - Define the four media features; technology, content, situation, and symbol systems, in the learning service in order to improve the mental processes.
    - Map and limit the number of symbols used in communication in order to serve most skills and styles of learning.
    - Consider how the context can influence learning.

2.3 Constructivism

When people are in an active learning process they transform information, construct hypothesis, and make decisions. This is described in the constructivism learning theory mostly based on research by Jerome Bruner [2], Jean Piaget [3] and Lev Vygotsky [4]. I find it most relevant to focus on the link between this theory and practice that Jonassen has proposed by systematization the constructivism theories. Hence I want to focus on the andragogy theory that is focusing on adult learning.

There are two major stands of constructivism: social constructivism and cognitive constructivism. Each of them has its own emphasis, but they share several common characteristics. These characteristics are [5]:

1. All knowledge is constructed through a process of reflective abstraction.
2. A learner's cognitive structures facilitate the learning process.
3. An individual's cognitive structures are constantly developing.
4. Constructivist learning has to have constructivist learning methods and pedagogy to back it up.

Constructivist principles used in learning situations have an influence for the learning environment, the roles of the teacher, and the learner. Jonassen [6] illustrate how knowledge construction can be facilitated by the following design principles:

1. Constructivist learning environments provide multiple representations of reality.
2. These representations represent that complexity of the real world.
3. Knowledge construction is emphasized over knowledge reproduction.
4. Authentic tasks are emphasized in meaningful context.
5. Real world settings or case-based learning is provided.
6. Thoughtful reflection on experience is encouraged.
7. Enable context- and content- dependent knowledge construction.
8. Supports collaboration and social negotiation among learners.

Traditional education the teacher is seen as the transmitter of knowledge. In the constructivism the teacher is a 'guide on the side' which gives opportunities for the learners to expand their knowledge. Because of all learners do not have the same background knowledge or experiences, the teacher should focus on the missing connections for the learners.

In constructivism it focuses on active learners who are encouraged to be independent thinkers and problem-solvers which give them opportunities to hypothesize, to analyze, to interpret, and to predict.

In mobile-learning the teacher can support the learner live (for example E-mail, Instant Messenger, newsgroup, phone, video-application, or shared-application software) or as a virtual support-system (for example computer-based help-system, simulation-software). Example of use can be e-learning services that support instant assistance when it is needed just in time, or knowledge management services where useful experiences can be acquire the system in order to support other learners (workers) when they obtain the same problem. This kind of learning touches the andragogy area which is described in the next chapter.

> Relevance for technical developing:
> - Base the e-learning service to focus on problem-solving
> - Implement a support-system; live or virtually, which can guide or help the learner in the learning or problem-solving situation.

2.3.1 Andragogy

Andragogy is an alternative term of pedagogy which involves adult learning and the need of life-wide learning of a person. This includes institutionalized forms of learning, self-directed and partly-intentional or non-intentional forms of learning.

Jost Reischmann [7], University of Bamberg describes this by his model in [Figure 2]:
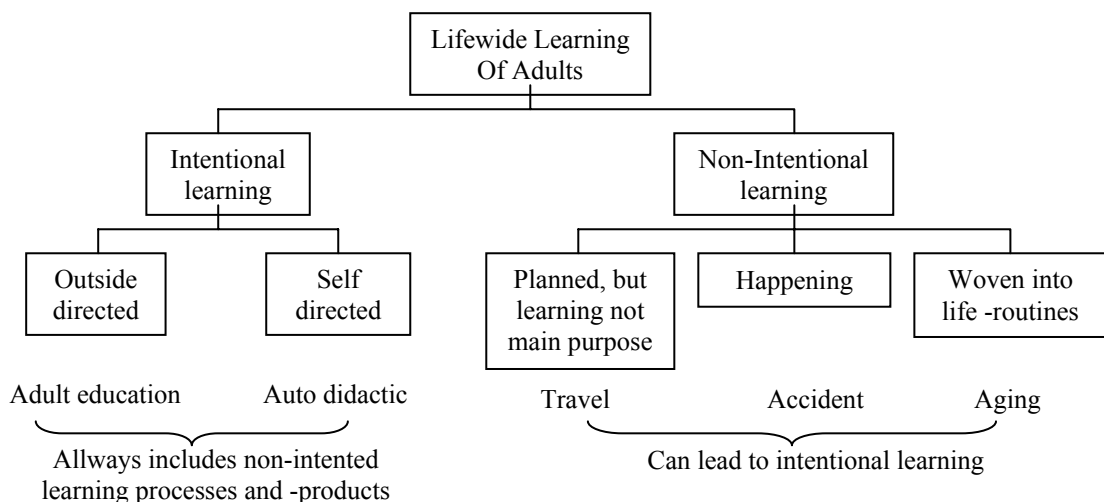
Figure 2 Structural scheme of adult education and adult learning

Andragogy emphasizes especially the differentiation between the field of practice and the scholarly approach. In practical terms, it means that instruction among adults needs to focus more on the process and less on the content being taught. The instructors should have a role as facilitator or resource rather than lecturer or grader. Examples of instruction can be simulations, role-plays, case-studies, task oriented problem solving, and self-evaluations. This improves the performance and supports the work-process.

M Knowles [8] proposed that there must be considered and addressed five issues in formal learning:
1. Letting learners know why something is important to learn
   - The need to know.

2. Showing learners how to direct themselves through information
   - The need to be self directing.
3. Relating the topic to the learner's experiences
   - Greater volume and quality of experience.
4. People will not learn until they are ready and motivated to learn
   - Readiness to learn.
5. A need to have a life centered, task centered, or problem centered orientation
   - Often this requires helping them overcome inhibitions, behaviors, and beliefs about learning.

The first and fifth opinions are about the need for a mutual understanding of a life-wide learning and why it is important to learn new things. Like the SMS (Short Message Service) changed the mobile communication culture among the youth, there is need to change the learning culture how mobile learning can be useful in the adults. The underlying model of how we learn is to be challenged to make changes in how to learn. According to Argyris & Schön [9], the ability to align the underlying model of how to learn (the media model of learning) to technical innovation is critical to succeed. They claim that learning needs to correlate with the organization culture in order to be productive. By this, learning systems cannot be static and must be in a mutual adjustment face. People are more frequently changing their jobs, and consequently there is need to learn the new employees quick in their new work. Examples of such jobs can be in the service industries where there often is need for different work situations and mobility. In contrast, people who have been working in same job for years needs to understand that no-one are finished their learning. The new working environment might be the factor that forces new ways of learning to take place.

The second and third opinions meet some of the advantages in mobile learning. Mobile technology can help people do training in between their work. If a worker gets a problem, he/she can use e-learning services to understand what to do and why he/she should do it that way. If he still have problems to solve the problem, he/she can ask facilitators such as a "teacher" (by phone), newsgroups (in the e-learning service), instant messaging (with other co-workers) or by shared applications (with an expert).

The fourth opinion about motivation and concentration of the learner gives challenges to the e-learning services. Traditionally, learning among workers is in classrooms often in another city or country. The e-learning situation is often on the workplace as an application or as fragmented learning in between the work-situation. Fragmented learning done in the work situation might improve the motivation or concentration due to its exploration and its problem-centered approach. On the other hand, if the e-learning service does not support or provide a teacher-role or guide, fragmented learning might give confusion to the learner.

When doing problem-centered learning, it accounts on the highest level in Bloom's taxonomy of educational objectives. These levels are described in the next chapter.

Relevance for technical developing:
- Add functionalities that motivate the learner to use the e-learning service
- Relate the topics to the learner's experiences
- Know that organizational structures are always changing and
take this into consideration when developing learning services.

2.4 Bloom's Taxonomy

In the learning taxonomy, Bloom and his colleagues classified the variety of educational objectives, the way people learn in. There are three types ordered from the simplest to the most complex; cognitive, affective, psychomotor. Cognitive is for mental skills, affective is for growths in feeling or emotional areas, psychomotor is manual or physical skills. All these levels are divided into new sublevels. I find it reasonable to focus on the cognitive aspect, which is most relevant to my thesis.

The cognitive aspect has six sublevels where the learner gradual adopts each sublevel:
1. Knowledge
2. Comprehension
3. Application
4. Analysis
5. Synthesis
6. Evaluation

The knowledge level is when the person can define terms. The level of comprehension is the level when the person can work on assigned problems and can exemplify what he did. At application level the person recognizes what methods to use and then use the methods to solve problems. At analysis level the person can explain why a solution process works. The level of synthesis is when the person can combine the part of a process in new and useful ways. And last, at evaluation level the person can create a variety of ways to solve a problem and then, based on established criteria, select the best solution.

In this framework, Bloom claimed that the learner must finish one level before utilizing the next one. This means that e-learning services should support all levels in order to support persons of all cognitive levels.

In the choice of test type, some might want to know which level the learner is on according to Blooms taxonomy. NETg [10] found out that they used a poor exam type. In the past they used a multiple-choices exam which correlates level 1 and 2. By using Bloom's taxonomy they adjusted the exam method to be problem-solving which draw a parallel to level 6.

By Blooms framework, mobile-learning might improve learning processes on every level by:
- Provide repetition-service on learned subjects (level 1)
- Provide tests on learned material (level 2)
- Give Just in Time (JiT) learning that the learner can use in a practical situation (level 3)
- Provide background-information so the user can evaluate the relevancy of data to the specific situation (level 4)
- Give tools that help the user to develop new documents or projects (level 5)
- Give the user different information on the same subject to let he/she evaluate/judge what information that is relevant (level 6)

Relevance for technical developing:
   - Use Blooms framework to utilize the learners' skills and learning processes.

2.5 Communication theory

In communication theory I want to focus on the elements that can influence the quality of the message through different kind of media. There are two major aspects that can affect the message; the physical delivery of the content such as network or phone lines and the comprehension of the message.

The physical communication theory is described by Shannon and Weaver [11]. He explains that all information is transported by a physical medium that he calls channel. This channel can be for instance air, phone-line, paper, or Internet. Every channel has its own limitations that influence the information and this can be seen as noise [Figure 3]. For instance when watching a video on a laptop through a mobile network, a vary bandwidth can be a noise source that can result in diffused video and reduced the comprehension of the message.
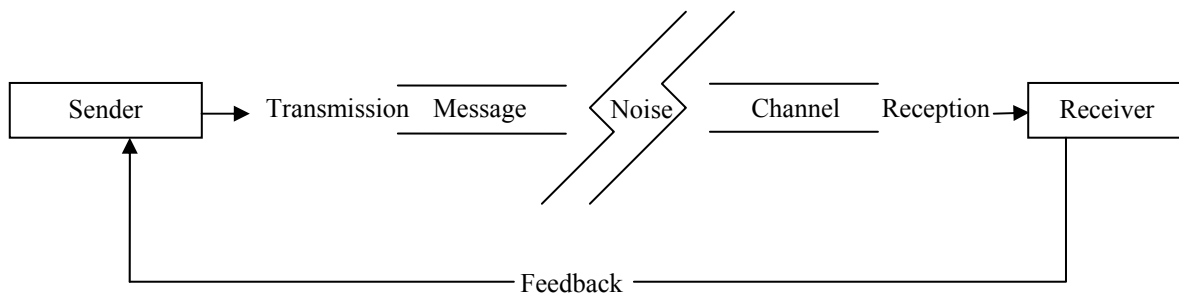
Figure 3 Shannon's communication model

The comprehension communication theory is proposed by McLuhan. He was a psychology researcher and made up some well discussed theories and statements. One important statement was from his book, "The medium is the message", where he meant the medium is affecting and changing the message, (described in Figure 4). He argued that content (1) is constant and the chosen medium or channel (2), which has its own technological facilities, must be involved to make the content into a message (3). The person that is interacting with the medium makes up his/her mind of the message's meaning due to his/her brain's ability to decode the message.

This means that constant information like knowledge placed on a central server can have different meaning after distributed to different media such as car-computer or handheld computer.
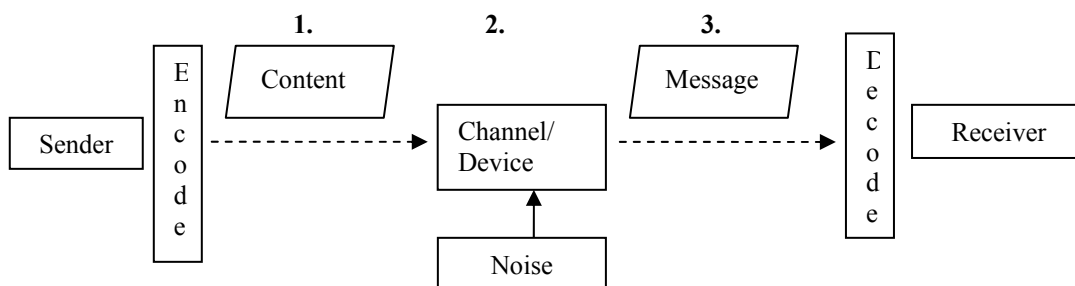
Figure 4 Communication from sender to receiver, McLuhan

Both theories states that the individual meaning of the message relies on the medium and its technological facilities. Most e-learning services are explorative, learning oriented, discursive and fragmented. This makes challenges to mobile learning services, due to its nature to be compact and concrete information.

> Relevance for technical developing:
> - Define how technology or the medium affects the message.
> - Define what amount and type of information is required to make learning happen.

2.6 Multi-channel and rich media integration

Multi-channel communication theorists claim that learning is effective when cues presented across channels are related or redundant [12] and together make a mutual meaning. By this, the individual information pieces sent through different channels is shaped into a message. Hsia [13] claims that multi-channel communication increases the dimensionality of the information, strengths the information sent through each channel, and gives a improvement of the total communication-quality. Redundant information pieces related to other pieces that have no individual meaning can lead to unintelligible message if one of them cannot be presented.

Present e-learning services can be far richer than the multiple channel theorists could imagine 30 years ago. Showing video does not need to be only by using audio and video together, but can be a joint of different media types or channels to strengthen the information. By using e-learning services meant for wired devices but adapted for mobile devices, it might give challenges if various cues are redundant and depends on each other. If the mobile device cannot present all cues, the information might be equivocal or meaningless. Then there should perhaps be a restrictive use of too many redundant cues.

The media richness theory has much in common with multi-channel communication theorists but goes deeper. It deepens a rational explanation for the choice of media. The media choice should be driven by the message's content in such way that the different use of media is supporting and together emphasize the message's meaning. Equal to the multi-channel communication theory, the equivocality of the message must be matched to the medium so that uncertainty in interpretation of the content is reduced. It is expected then that communication is most effective when the medium matches the content of the message.

Lengel and Daft [14] proposed a classification of lean and rich media and stated face-to-face as the richest communication medium to e-mail as the leanest communication medium. Hence Trevino, Daft, and Lengel [15] proposed that the degree of complexity, emotion, controversy, or ambiguity in a message affects which type of media that should be used. Richer media can capture the non-verbal aspects of communication and provide trust and learner motivation.

> Relevance for technical developing:
> - Define which information pieces are redundant and what the e-learning services should do if some of them cannot be presented on the learner's device.
> - Use rich media in order to provide trust and motivation.

3   Multi-channel e-learning services, description and discussion

Multi-channel e-learning services have two important dimensions that I want to discuss before defining the term multi-channel more thoroughly:

1. the pedagogical effect by receiving learning through different ways
2. the availability of the learning in different contexts

First, instruction by using different mediums is not a new thing. Instructors and pedagogic theorists have for a long time used different methods or channels to utilize the learning processes such as:

- lectures
- discussions
- collaborations
- simulations
- repetition
- practices
- multimedia such as video, music, and pictures

By using computer science, services can be utilized and adapted to improve the learning quality. For instance, discussions or chat sessions can be stored for later use or simulation can be improved by live animations.

Secondly, we are in a constantly learning process and we are learning in different contexts. Impressions and experiences change our realization of different thing. By talking, exploring, thinking, reading instructions, hearing, feeling, etc. we learn how to do new things and why it is so. Structured learning, such as instruction where the teacher-role guides the learner to easier understand is usually done in classrooms, at home, at work, on the move, or in a situation when one faces a problem. Traditionally instruction has been done as the points described above, by telephone, by telephone-video, manuals, books, instructor, etc.

E-learning can improve the availability to the learning-tools. Books, manuals, discussions, simulations can be utilized in more situations than they have been used traditionally. For instance, books or manuals can be changed with intelligent devices and the content can be instantly downloaded. This approach has been stressed by e-learning developers in recent years because of the great development of broadband networks and mobile networks and has resulted in many e-learning services and applications. The thought is to give people the opportunity to learn more instantly, Just in Time, when they face a problem, or when they just want to learn something about a specific topic.

Many challenges have appeared in solutions that support instant learning-services which this thesis stresses. Firstly, the system must support the learner's device. Secondly, due to many different devices, the system must be capable to deliver an understandable content to the user. Thirdly, the system must handle different network connections. A service that fulfills this can be called a multi-channel service. Multi-channel services are described in the next sub-chapter.

3.1 Multi-channel services

There is no general definition of e-learning multi-channel services. By basing on different thoughts, I have defined it as:

> Multi-channel service is an architecture that can carry content to a range of devices through specific interaction software.

Content is pieces of learning materials like a paragraph of text, a picture, sound, music, graphics, video, or data. A service is technology that delivers content to the user. Multi-channel service delivers content through several channels.

Device	Interaction software
Palm	Web-browser
mobile phone	Wap-browser
Home/Work computer	Shared/distributed application
mobile laptop	Instant messenger
	E-mail software
	Newsgroup-software

Table 1 Examples of devices and interaction software in channel services

The ideal technical goal for a multi-channel service would be to automatically deliver all content with an acceptable quality through all channels. Such quality is dependent on the device's physical characteristics and technological capabilities. Devices are very different from each other which make this approach almost impossible. Hence, the presentation and amount of the content are varying such as different format, layout, size, etc.

A realistic target would be to make the multi-channel services to automatically deliver content with a maximum attainable quality of content through all kinds of channels.

Supporting all types of channels in multi-channel services is unrealistic to the service developers, and makes challenges to obtain a high quality of the content to most devices. Consequently, there is reasonable to limit the number of supporting devices to the most potential. The range of supporting devices is dependent on the purpose of the learning service.

Because devices have different facilities, the quality of the content can differentiate between the devices. Such quality reduction can be that the content is not adapted to the physical and technological limitations of the device, there is lack of network capabilities, or the content is not adapted to the situation of use.

Quality reduction can prevent a coherent meaning of the content and reduce the motivation to the user. For example in mobile situations, users mostly want an instant access to the information. If the service cannot present it adapted to the device, he/she might not tolerate this quality loss and perhaps not continue to use it.

There are many companies that develop multi-channel e-learning services as Cisco, Oracle, and Microsoft. Developing multi-channel services might involve problems. Should the service support all potential devices on the market, or only support some selected in order to provide the best functionality and quality of the content? By supporting only a range of devices, this

can exclude many devices and users which might hamper it to be a "Just in Time" and "Learning on Demand" service.

There are many devices, functionalities, architectures, and standards that must be specified in order to fulfill multi-channel services to an unknown number of devices. This is characterized and described in the next chapters.

4   Technology

Today the e-learning development society is in many cases removing the 'e' from the 'e-learning'. That might be results from the evolving Internet technology and the following user- and company-demands. In the last ten years, e-learning services have developed from to be courses performed on computers in learning institutes or on CD-ROMs to more 'Learning On Demand' centered mobile learning services based on and performed in the worker's work-situations.

This chapter is defining and discussing the different technologies that are needed to supply content to such mobile e-learning services from a server. The focus is on multi-channel services that support different devices and the challenges this involve.

The main challenge is that Internet has had a great global expansion in number of connected devices, innovative software- and hardware-technologies, and many new non-technical users with different demands. Since this evolvement has mostly happened the last ten years, there have been many egoistical technical developments in order to make positions in the market. This has resulted in confusions to the content-providers on what protocols and standards that they must use to provide content to many devices.

In recent years most software- and hardware-companies have faced that the increasing number of different capable devices makes it hard to develop services. In e-learning it has been attempts to re-authorize content to the different groups of devices in order to support most of them, but it has resulted in an exclusion of the other devices – and of course the users that are using them. Most researchers and academic environments have since the start of the Internet focused on the mutual need to follow the official standards and the commercial companies have faced that it is gives a mutual value.

This chapter defines, describes, and discusses what exists of common used standards and how they can be used in multi-channel e-learning services to support different capable devices.

Next sub-chapter is defining channel characterization used in the term e-learning multi-channel service.

4.1   Channel characterization

Today there are a number of devices or terminals on the market due to a great innovation among the developers and the increasing need for mobile devices. These can be categorized into four groups:
   1. Stationary computer
   2. Laptop
   3. Palm/Handheld
   4. Mobile phone

Even though there is a great technological innovation in all groups, the least standardized group is the third and fourth group, mobile devices. This may give consequences in e-learning services that are meant to be used on different mobile devices and this topic will be discussed in the next chapters.

Because of the mobile market is quite new, there are no market leader that involves many unequal devices in usage, design, software, and hardware. The strength to this is the increasing improvement of the technology, but the disadvantage is the outcome of perhaps too many different technologies. Fortunately, many developers are using same operating systems and open source applications that decrease this confusion.

4.1.1 Operating system

There are two types of operating system architectures, native, and intermediate. Native operating systems run directly on the processor and the intermediate are running on top of other operating systems which means that applications is not dependent on the underlying operating system.

The most used operating systems are:
1. Native
   a. Windows
   b. Linux
   c. Macintosh
   d. Palm
   e. WinCE
   f. Symbian
2. Intermediate
   a. Java

These operating systems can handle some or more technology facilities which are specified in [Table 2] and [Table 3].

Input	Output	Processor/Memory/Power	Network capabilities
Keyboard	Screen	Multimedia processor	IR to mobile phone
Mouse	Sound	RAM	Bluetooth
Touch screen	Paper output	ROM	Modem 2400-128000 kbps
Voice input	Braille (system of	Hard-drive	WLAN
Keypad	writing for the blind)	Battery	Ethernet
Joystick	Vibrator	Power	
Digitizer			
Sensors			

Table 2 Hardware

Media types	Client application
Video	Content Markup
Picture	Client side scripting
Sound/Voice	Applets
Text	Styling
Data	

Table 3 Software

4.1.2 Input

Input technology is something that affects the device with some type of signals. These signals can for instance be characters typed on a keyboard, voice, a temperature reader, etc. Mostly these signals are initialized by person to bring in words or control signals to an application. What input technology the device handles often depends on the situation and the device's functionalities. Some situations only needs touch screen, such as workers in the service industries. Some devices need functionalities such as voice input or special keypads to help people with handicaps.

Most e-learning services only need two main input technologies: navigation and text-input. Some devices have combined these input technologies like touch-screen and text recognizing functionalities provided with Palm. The challenge to input technologies is in situations where input from a person is problematic. For instance, workers in gas- and oil-industries or people driving cars might need specialized input technologies. By start using such input technologies might be unfamiliar for some people which can prevent them from using it.

4.1.3 Output

Output technology is the way the devices communicate with the surroundings. This technology can for instance be printers, screens, vibrator, sound-system, etc. In e-learning the output capabilities often have a relation with the content and its media types. Even though there are many variations of media types, most content are presented through a screen or through speakers.

There are many characteristics of screens that can limit the possibility to display the different media types. Among these are:
1. Color depth: Photo quality, 256 color, grey-scale, a few colors, one color
2. Display: high/low resolution, pixel per inch (ppi), screen size
3. Speed of the display; video-capable
4. Backlight

The most sound output capabilities are described below:
1. One tone, dual tone, wave capable
2. MIDI-synthesizer capable
3. Volume/Pan
4. Mono/Stereo

4.1.4 Processor/Memory/Power

The processor, the power, and the memory are important design factors to mobile devices. They affect the device's price and size. The stronger processor on the device, the more energy it consume, and more warmth it emits. By this, more devices are using poorer processor power than stationary computers that results in a lesser ability to process much information, calculations, heavy graphics or video.

On mobile devices, the power is based on batteries. Even though there is a great innovation in this technology, the length of the battery life is poor dependent on if the device uses strong processor, color-display, or some kind of network connection. The average length of the battery-life when constantly working on laptops is 2-4 hours, Palm-sized from 8 hours up to several days, and mobile phones around one-day of use.

The devices' memory can be RAM (random access), ROM (Read Only Memory) or storage-mediums such as hard-drives. RAM and ROM are widely used on mobile-devices which does not consume that much power like storage-mediums. The consequence by using RAM and ROM is the limited capability to store much information compared to storage-mediums.

The devices' range of use depends on the different properties of the chosen device. If the battery capacity does not satisfy the requirement, the user might use it less, and the device gives more mess than value.

4.1.5 Network capabilities

Network technology makes it possible to receive or send data between other computers. The network capabilities can be categorized by:

- its data transmission technology such as the Token Ring and TCP/IP
- whether it carries voice, data, or both kinds of signals
- who can use the network (public or private)
- the usual nature of its connections (dial-up or switched, dedicated or non-switched, or virtual connections)
- the types of physical links (IEEE 802.x, optical fiber, coaxial cable, and Unshielded Twisted Pair, GPRS, GSM)..

Source: searchnetworking.com

Devices such as stationary computers, palm pilots, and cellular phones are capable to use one or more of these physical links:

Standard	Maximum bandwidth
TP	100 Mbps
IEEE 802.x	11 Mbps
UMTS	2 Mbps
EDGE	384 kbps
GPRS	115 kbps
ISDN	128 / 64 kbps
Analog modem	56 kbps
HSCSD	43,2 kbps
GSM	9600 kbps
Bluetooth	720 kbps
IRDA	4 Mbps

Table 4 Network technologies and their bandwidths

The table above describes the maximum of theoretical bandwidths to different physical links. In reality these are seldom optimum, especially in mobile network-connections.

The network-speed is not only dependent on the bandwidth, but also delays. A delay is the time used to transmit a signal from a client to a server and then back to the device. Such delays can happen in: satellite networks, networks with much noise where there is need to retransmit the data, routers with high traffic, or because of high server processing time of the data.

Delays and unstable bandwidth might not be satisfactory to services that need good network quality. Such services can be live interaction with graphics, sound, text, data-stream or video like in a conference where delays are not accepted.

Quality of Service (QoS) is the idea to solve these problems. The transmission rates, the error rates, and other characteristics can be measured, improved, and to guaranteed in order to offer an acceptable network quality to specific services. By using QoS, data packets transferred over Internet can be prioritized by importance that can prevent some of the delay factors.

Some networks supports roaming which means the ability to provide a virtual home environment (VHE) to the foreign network or provide a mutual payment solution between the different networks. Internet architecture offers (VHE) through mobile IP in IPv4 and IPv6. In order to support mutual payment solution between the different networks, the ISP needs to make an agreement between other ISP's.

Currently there are no services that support roaming between network layers such as IEEE 802.x, UMTS and modem. This kind of technology is a part of the fourth generation networks. If the e-learning service depends on roaming, this might give challenges.

4.1.6 Media types

Media types can be hard to characterize because they can convergent and appear as a combination. In this thesis I will focus on the main media types that are described in Table 3. By present or display media types, it depends on the input, output, processor/memory, network, and software capabilities of the device. Some devices can hinder the presentation, due to lack of color-display, stereo-sound, or video-rate that may reduce the message or the media quality.

A media type can be stored on the device, be streamed or downloaded through a network. Media types can be stored in different formats adapted for different purposes. Such purposes can be to utilize the media types for low bandwidths, small screens, or poor sound capabilities. Present, there are many different formats and many of them are not standardized. If developers or content users make use of such non standardized file formats, this can reduce compatibility to many devices. Hence, many non standardized file formats are broadly being used and become like a 'standard' such as the file types: GIF, AVI, WAV.

4.1.7 Client applications

There are three client application architectures that support different types of devices and operating systems. These architectures are thin client, proprietary client, and Internet client which are described in chapter 4.3.

The thin-client architecture (chapter 4.3.1) is designed for a downloaded graphical user-interface and the applications are processed on the server. This makes a less need of processor-power, but requires a more stable network connection.

The proprietary client (chapter 0) is designed for a local user interface and local application processing. This makes that calculations can be done on the client and result in a less need in stable network, but require more processor-power.

The Internet client (chapter 4.3.3) is designed for a local browsing tool where the content and presentation is downloaded on demand. This requires in most cases medium processor-power and medium stable-network.

4.2 Typical devices vs characterizing capabilities

It exist many devices with different capabilities. The table below describes the performance quality of device capabilities. The score is based on the average used devices.

		Stationary computer	Laptop	Palm/ Handheld	Mobile phone
Input	Keyboard/ pad	9	9	5	4
	Mouse/ Touch screen/ joystick	9	9	7	2
	Voice recognizing	9	9	4	4
	Text recognizing	5	5	9	1
Output	Screen	9	7	4	3
	Sound	9	8	4	2
Hardware	CPU, RAM/ROM, Storage	9	8	4	2
Network	Support of different networks	9	9	6	4
Media types and presentation	Video	9	9	4	2
	Picture	9	9	6	4
	Sound	9	9	4	2
	Text	9	9	9	6
	Data stream	9	9	4	2
OS	Native code	9	9	9	9
	Intermediate code	9	9	6	3
Markup	Supports different Markup-languages	9	9	5	3

0 - No support

1 - Limited support (character display, tone-signal)

3 - Support exist, but poor quality/effectively (simple graphics, poor resolution, simple keyboard)

6 - Satisfactory solution (high resolution and graphics, full-size keyboard)

9 - Good solution

Table 5 Devices' capabilities in different situations

This table shows us that the average quality of the device capabilities is decreasing the smaller and mobile the device is. The most important factor is that mobile phones have poor support to media types and output facilities, which means less support for rich-media learning services.

4.3 Types of client/server-architectures

This chapter describes relevant architectures in which the e-learning service or content can be distributed from a server to more clients through a network.
First, I want to base on the client architecture's perspective and then look at the underlying individual architectures.

There are three different architectures;
1. Thin client architecture
2. Proprietary client architecture
3. Internet client architecture

4.3.1 Thin clients

The "Thin client" architecture is a centrally-managed computer where the applications are executed on a server and the graphics are executed on a thin client computer. Currently, the major systems are Citrix MetaFrame [16] and Windows Terminal Services [17] that supports different devices, but it also exist an open-source system based on the Java virtual machine, VNC (Virtual Network Computing) [18].

One of the advantages of Thin Client Architecture is that the applications are centrally managed and processed. This means that the users do not need to think about software-installation and the system-administrator does not need to install new software or updates on every device. Consequently, the logic and the application processing are centralized and the devices might obtain a longer life. If the software installed claims more processing power, it only depends on the centralized server.

Since the device only processes the graphic, the requirement lies to the devices' network capabilities, graphical interface, and memory. The graphical elements in the above solutions are based on graphical objects. Graphical objects are described mathematically as shapes and lines, but pictures must be transferred as they are. This means that most graphical objects can be transferred with a reasonable quality over poor networks, such as mobile networks on 9600 kbps.

Even though the most systems support compression of transferred data, this might reduce support of rich multimedia such as video and sound in poor network. If the network does not provide the required QoS (Quality of Service), the presentation of multimedia can result in unintelligible information. None of the systems described above supports methods to handle this challenge.

The most systems support many devices and Operating Systems. Both Citrix- and VNC-clients support the most used PDA and mobile phones. Windows Terminal Services only supports Windows Operating Systems such as Windows 3.1/95/98/ME/XP/2000/CE.

4.3.2 Proprietary Clients

The Proprietary Client Architecture is a stand-alone client application developed to support a specific service. Examples of proprietary clients can be simulation-, database-, word-processing-, and course-applications. IBM Mindspan Solutions' Lotus LearningSpace 5.0 [19] is an example of an e-learning application that provides functionalities such as good student-

interface and real-time collaboration tool. LabView [20] is another example that provides different laboratory tools like simulations.

Strength to use this type of architecture is the possibility to have a near relation between the service and the device's hardware and operating system. This makes it easier to utilize the facilities on the device and make good functionalities. The weakness is the high development expenses because of the need to develop clients to every client platform.

Stand-alone applications give opportunities to have all user-interface and logic bundled with the application. By doing this, the transfer between the server and the client can be reduced which improves the application's response time. The disadvantage by placing logic on the device is the increased requirement of processor power.

4.3.3 Internet Clients

Most widespread Internet clients are web-, wap-, or I-mode-browsers. It includes browsers to the most terminals; PC, Laptop, Palm sized devices, mobile phones, WEB-TV, refrigerator, and Kiosks. Internet Explorer, Netscape, and Opera are the mayor browser developers.

One reason to the many browsers is that they are using standardized technology. Independent organizations composed of representatives from different companies make the standards. W3C is the organization that works out standards to the web technologies and WAP forum works out the standards to the wap technologies.

A challenge to the standardization organizations is the browser-developers are adding new non-standardized functionalities to their browsers. Content and logic developers have used these functionalities and then made content best fitted for some browsers. This does not only give challenges to the standardization group, but also the many people that cannot in worst case use the content because of their browser.

Hence many browsers, especially the mobile browsers, have different technological capabilities. Not all browsers are capable to present the media types as it was thought. This can frustrate the content developers that are using media types not supported by all browsers.

An advantage to Internet clients is that browsers often are pre-installed (especially mobile phones) or free to download.

In a comparison of these different client architectures, there is need to know the purpose of use. All have its strength, but in mobile e-learning, one of the architectures is pointing out. The Internet client provides a flexible technology in addition to a broad user group. Therefore, I want to focus on the Internet clients in the further work because of its well standardized technology and potential to deliver content to most different wired and wireless devices.

4.4 Internet client content adaptation architectures

As stated in previous chapters, browser and devices have different technological capabilities such as different support of markup-languages, which media types that are supported, the size of the display, color or sound capabilities.

Because of different device capabilities, an adaptation of the content and the presentation is needed before it can be presented to the user. This adaptation can be done on the server, on a proxy or on the client. Mark Butler [21] at HP [22] set up these in [Figure 5].
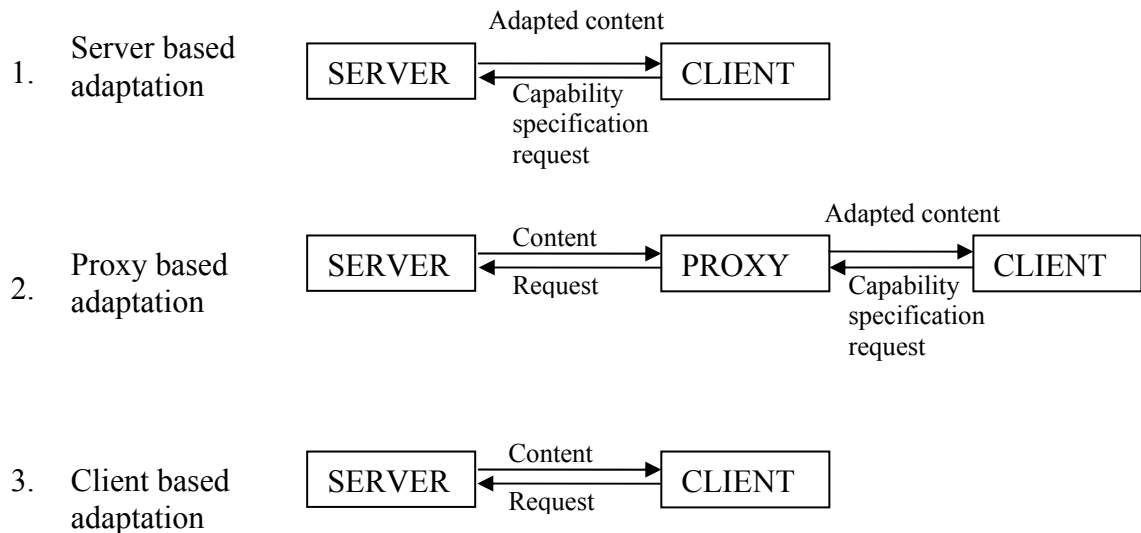
Figure 5 Content adaptation architectures, Mark Butler, HP

1. A *server-based* solution
   a. e.g. by using server-side programming-languages such as PHP, ASP, XSP to do the an content adaptation from XML to respective markup-languages
2. A *proxy-based* solution
   a. e.g. by using an AvantGO proxy or Web-clipping software to convert all html-documents on the web to a format that is usable to the browser
3. A *client-based* solution
   a. e.g. by using the client browser to process the XML/XSL or XHTML / CSS like what Internet Explorer is capable to do

4.4.1 Server-based content adaptation architecture

The server-based content adaptation solution is broadly used. The logic work is executed on the server that sends a fitted markup-language and its content to the device. There are both advantages and disadvantages doing it. The advantages are:
   - The content can be structured in e.g. XML (chapter 4.8.2) to simplify the transformation to other markup-languages
   - Calculations that need much processor power can be done on the server instead on the device.
   - Simplify connection with databases or applications installed on the server.
   - By content control negotiation with the device, the server can control the presentation layer and send content that the device is able to present.

- Increases the chance that the content can be displayed on most browsers and devices.

There are weaknesses by using server-based content adaptation architecture, and the most important are:
- Not all browsers support content control negotiation which means the server must take assumptions on the browser's ability to present the content.
- Heavy server-side applications may slow down the server
- Often the content is made to utilize one browser's technological facilities application. This might reduce the use of other browsers' facilities.

4.4.2 Proxy-based content adaptation architecture

The proxy-based content adaptation architecture is quite new architecture that converts pre-defined types of content on the fly. Applications can be all from receiving e-mail as a SMS message or by converting HTML-pages to WAP-pages.

The advantages by using this architecture are:
- Make content available to new technologies. In the start of the WAP-epoch, it was lack of content. Proxies-based content adaptation architectures opened up the web-word.
- Lets the content creator focus on one type of presentation. People that are not using the target browser can make some use of the content, perhaps in a poorer quality.

The weaknesses are:
- It does not care about how the content or presentation layer is built. This might give challenges to different sized browsers. For instance broad images or tables might not fit very well on small mobile phones.

4.4.3 Client-based content adaptation architecture

The client-based content adaptation architecture is a solution where the client presents the content by using a style-sheet. This is not common on devices today, but might be used more in some years.

The client-based adaptation solution is a solution that is a good solution. The weaknesses are that there are almost none devices that support XML and XSL transformations yet due to a requirement on high processor power. In addition, it requires higher bandwidth since the content, the presentation and perhaps some logic needs to be in separate files.

If adaptation takes place on the server or the proxy, they all will need to know the identity of the clients or its capabilities to display different type of content and presentation. I want to focus on the server-based content adaptation technology, which I find more relevant to my prototype. This identification of device capabilities is described in the next sub-chapter.

4.5 Content control negotiation between the server and the device

In order to make an adaptation of the content to different devices, there is a need to know some about the device. I want to focus on standardized methods. Currently, there are two main methods to do this:

1. By using a unique identifier that the client device sends on every request to a server.
   a. HTTP request header field [23] (standardized by W3C [24]),
2. The device gives a capability specification to the server
   a. Composite Capabilities / Preferences Profile [25] (standardized by W3C)
   b. User Agent profile [26] (standardized by WAP Forum [27])

4.5.1 HTTP request header field and HTTP/1.1 Accept Header Fields

The HTTP request header field [23] is a unique identifier sent from browsers to servers. Web servers are using this information mostly for statically measurements, but also to provide content designed for different web-clients.

A user agent string that browsers are sending is a W3C standard and has the following syntax:

Browser / version (platform; security-level; OS-or-CPU description)

Figure 6 User-agent string

There have been attempts to extend the request header format to increase the use of it, but this has not occurred in a standardized way. For instance Netscape uses its own language specification in the request header and some proxies add proprietary information to the request header as well. Also the standard installation of the browser Opera identifies it selves as it was the Internet Explorer browser. Such wrong information limits the practical use of this string.

When developers are using the user-agent string in content adaptation, it can cause problems. Many web-pages are building upon a user-agent test to provide a page designed for the specific web-browser and to use different potentials in the browser. If the browser is unknown or are identifying in a wrong way, this can cause problems. Developers who build too poor user-tests can cause problems when a newer version of the same browser is provided or cause problems for other browsers by excluding them. Earlier the Internet Explorer identified it selves with Mozilla because web developers looked at the user-agent field to see if it was a Mozilla browser. That was the only way to see if it supported framed web pages. Clearly, if we do provide specific content for a device, we need to make sure the device or browser can override that if necessary. Ideally, devices should give more information about their capabilities and preferences.

Browsers that support HTTP/1.1 [28] are sending more information than the Request Header Field. When a client tries to retrieve a web page, it sends a request to the server. This information is known as the Accept Header. The HTTP/1.1 version is using four headers; Accept, Accept-Char-set, Accept-Encoding, and Accept-Language.

Examples of these fields are below:

Internet Explorer 5.0:

> Accept: image/gif, image/x-xbitmap, image/jpeg; mbx=5000, image/pjpeg, application/vnd.ms-powerpoint, application/vnd.ms-excel, application/msword, */*
> Accept-Language: en-gb
> Accept-Encoding: gzip, deflate

Figure 7 HTTP/1.1 Accept header, Internet Explorer 5.0

Netscape Navigator 4.73:

> Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*
> Accept-Encoding: gzip
> Accept-Language: fr; q=1.0, en; q=0.5
> Accept-Charset: iso-8859-1,*, utf-8

Figure 8 HTTP/1.1 Accept header, Netscape 4.73

4.5.2 CC/PP Composite Capability Preferences Profile

The Composite Capability Preferences Profile (CC/PP) [25] is created by W3C [24] and was an outgrowth of the W3C's Mobile Access Internet Group. The CC/PP's purpose is to provide an extensive method for communicating the capabilities of devices such as clients, proxies, gateways as well as a method to cache resources.

A CC/PP profile is generally constructed as a two level hierarchy: a profile has a number of components and each component has a number of attributes. A protocol for transmitting CC/PP profiles has been proposed but is based on an experimental variant of the HTTP protocol known as HTTP-ex [29]. Many existing servers do not support this protocol extension, which means that the developers have to adjust it to make it compatible in a way. HP has made a platform called DELI [30], which uses a protocol named W-HTTP proposed by UAProf. This has identical functionality to the CC/PP protocol based on HTTP-ex, but is compatible with HTTP/1.1.

A weakness to CC/PP is that it does not resolve two of the key problems concerning device independence. It does not provide a standard vocabulary for web clients to communicate their capabilities to servers and it does not describe the type of transformations and customizations that servers are expected to perform on behalf of devices based on their capabilities. The mandate for the CC/PP working group is not to see on these problems. These problems are beyond the scope of the CC/PP working group but must be addressed in order for the protocol to be of practical use.

This protocol is quite new and few browsers support it. Server-software should then make use of the HTTP request header field, HTTP/1.1 Accept Header Fields or the UAProf described in the next chapter.

4.5.3 UAProf

The UAProf [26] is a standard being developed by the WAP Forum and is proposed to be used to specify a communication standard between WAP-devices and servers. UAProf profile is a two level hierarchy composed of components and attributes. Unlike CC/PP, the UAProf specification also proposes a vocabulary - a specific set of components and attributes - to

describe the next generation of WAP phones. The specification also describes two protocols for transmitting the profile from the client to the server.

A strength to UAProf is that it defines five different categories of device capability: software, hardware, browser, network, and WAP. The weakness is that it does not resolve how servers and proxies should use the information provided by clients.

Broadly the CC/PP and the UAProf standard have much in common. The similarity is that they are both based on the W3C standard Resource Description Framework (RDF) [31] as well as CC/PP profile is designed in such way that it is compatible with the UAProf profile settings.

4.6 Content storage

Content is basically stored as files in a file-structure or as elements in a database-structure. It exist content management server solutions such as Microsoft Content Management Server [32] that provide file and database storage solution as well as a good user-interface, content indexing, search-routines, and content streaming facilities. I want to focus on the file-structure- and the database-technology.

4.6.1 File-structure

A file-structure can either be static without functionalities or be dynamic with functionalities such as indexing, and filter-capabilities. The advantage with file-structure is that it is not dependent on other storage-software than the operating system. By using file-structure, content can easily be stored on CD-ROMs, servers or other media.

4.6.2 Databases

A database is a collection of data that is organized so that its content can easily be accessed. The most common types of databases are relation-database and object-oriented database. A relation-database is a database from which the data items are organized as set of tables. The data can then be accessed or reassembled in many different ways without having to reorganize the database tables. An object-oriented database is data defined and stored in objects, classes, and subclasses and includes support for inheritance.

The advantages to database-systems are the scalability, the ability to reorganize, and response time on search requests.

4.7 Content authoring tools

There are two types of XML and XSLT authoring tools:
- code oriented
- presentation oriented

The code oriented XML and XSLT authoring tools provides a good environment for people familiar with programming and code-handling. Most tools handle DTD which help the developer to accomplish the XML structure and chosen tag-rules. The XSLT part of those tools might confuse because it does not provide authoring in a graphical and layout environment.

The presentation-oriented XML authoring tools provides a WYSIWYG interface in which can be familiar for most content developers. The difference from HTML tools is that dynamic content is not inserted in XSLT. Therefore, the editor needs to complete an XML source that he/she can base his/her XSLT work on.

There are many XML/XSLT authoring tools, which are presented in Appendix

Because of many standardized XML markup-languages there are also authoring tools made especially for them. Among the WYSIWYG authoring tools are

SMIL:
- GRiNS Editor (Mac, windows, SGI Irix 6, Sun) from Oratrix
  http://www.oratrix.com/GRiNS/
- MSlide Builder (Web-based)
  http://www.mslide.com/build/index.html
- RealProducer (windows) from RealNetworks
  http://www.realnetworks.com/products/media_creation.html
- G2 Objects for Dreamweaver (Windows, Macintosh) from RealNetworks, Macromedia
  http://www.macromedia.com

XHTML:
- eWebEditPro (windows) from Ektron Inc.
  http://www.ektron.com/

WML:
- Rasquares Wap Pro (windows) from Rasquares Information Systems Pte. Ltd.
  http://www.rasquares.com/

MMS:
- Adobe GoLive (windows) from Adobe
  http://www.adobe.com/products/golive/main.html

4.8 Content encapsulating and Markup Languages

The content in e-learning solutions based on web is based on Markup-Languages that specify how content should be presented. This is done by tagging the content elements. There are many markup languages specialized for different networks, situations, and functionalities such as:
- HTML [33] (Hypertext Markup Language)
- XML [34] (eXtended Markup Language)
- XHTML [33] (eXtensible Hypertext Markup Language)
- WML [35] (Wireless Markup Language)
- CHTML [36] (Compact Hypertext Markup Language)
- SMIL [37] (Synchronized Multimedia Integration Language)
- MMS [38] (Multimedia Messaging System)

The functionalities to these markup languages are described in the next sub-chapters.

4.8.1 HTML

HTML (Hypertext Markup Language) [33] is the most used markup-language today. It includes content handling, presentation, supports plug-ins, and supports client-side programming. Even though it was standardized back in 1990 with the thought that it could be

used on unequal devices and network, it has not worked out. In the ninetieth both browser developers and web page developers tried to get most out of the standardization as well as adding new parts in it which resulted in a markup-language best fitted for home computers.

Like XML, HTML specifies the content. Unlike XML it also specifies how the content should be presented. Due to many unequal screens and device capabilities, this made it hard to provide content to an unknown number of devices and there have been attempts to solve this. 2001 W3C recommended XHTML that was the inheritor to HTML and was meant to solve the most of HTML's problems.

4.8.2 XML

XML (eXtended Markup Language) [34] is a data definition language designed to structure data and describe information. The author of an XML document is able to create their own tags, and therefore, is not dependent on predefined tags like those in HTML. A DTD (Document Type Definition) is used to define the vocabulary and syntax of the user-defined tags.

The main advantages are:
- Encapsulate and specification of content and presentation information.
- Standardization to other XML dialects (W3C, SyncML, IMS Project)
- Interaction between computers (protocol) and applications (data)
- Markup Language conversion is done easily by XSLT (XML->WML/HTML)
- Scalability, easy to extend, upgrade or convert

Like the markup language HTML, it is not common to encapsulate binary content (for example media types) in the XML document. However this is possible and is described in the MMS XML-dialect in chapter 4.8.6. Instead the binary content is referenced as a link. A simple XML example is provided below:

```
<chapter>
      <title>using Powershot G1</title>
      <img>
            <src href="http://www.minicourse.com/img/powershotg1.jpg" />
            <alt>Powershot G1 camera</alt>
      </img>
</chapter>
```

Figure 9 XML content example

By encapsulating the links of the media types, the browser can use its own methods or installed plug-ins to display it. Due to different browser limitations it is important to provide alternative ways to receive the information. If the browser is not capable to display a picture, an alternative text should be displayed.

There are two aspects e-learning solutions should consider when using the XML:

1. *The need to use standardized or wide-spread vocabularies.*
   XML do not offer rules on how content should be tagged. Today, there are not many vocabularies to specify content and most companies make their own. Among the standardization groups and open vocabularies are:
   a. IEEE Learning Technology Standards Committee (LTSC) P1484.17 [39]
      This committee describes the packaging of learning content down to tag-elements.

   b. Docbook [40]
      This technical committee describes packaging of learning contents, especially best suited for technical books. This is not an official standard, but is widespread used and its first version was made in 1991.

   c. XHTML (XHTML is described in chapter 4.8.3)
      The W3C committee has developed this vocabulary to specify content meant for browsing tools.

2. *The need to specify content-parts and course-modules in order to reuse it in other e-learning services.*
   XML has a nature to be integrated with other solutions, it is important to make a specification how the content-parts or modules should be integrated. The most well-known committees in this area are:
   a. IMS (Instructional Management System) Global Learning Consortium [41]
      This committee is defining the technical standards for interoperability of applications and services in distributed learning and supporting the incorporation of IMS specifications into products and services worldwide

   b. IEEE Learning Technology Standards Committee (LTSC) P1484 [42]
      This committee is defining learning object metadata, student profiles, course sequencing, computer managed instruction, competency definitions, localization, and content packaging.

   c. Advanced Distributed Learning (ADL) Initiative [43]
      This committee has released the Shareable Courseware Object Reference Model (SCORM) which provides a foundation for how the Department of Defense (US) will use learning technologies to build, and operate in, the learning environment of the future.

   d. The Aviation Industry CBT (Computer-Based Training) Committee [44]
      The AICC committee develops guidelines for the aviation industry in the development, delivery, and evaluation of CBT and related training technologies

   e. The Dublin Core: Metadata for Electronic Resources [45]
      This committee describes a set of metadata element intended to facilitate discovery of electronic resources

4.8.3 XHTML

The markup-language XHTML (Extensible Hypertext Markup Language) [33] was recommended by W3C in 2001 is the inheritor of HTML 4. XHTML is converted HTML 4.0 with purpose to be a fully XML-vocabulary. Unlike HTML, XHTML can be extended by anyone that uses it. New elements and attributes can be defined and added to those that already exist.

The XHTML's advantages are extensibility and portability. Extensibility means that as new ideas for web communication and presentation appear, they can be implemented without having to wait for a next version of HTML and browser support. Specific sets of extensions for XHTML are planned for mathematical expressions, vector graphics, and multimedia applications. Portability means that Web pages can be made simpler than they were before in order to support small devices. This is important for mobile devices with low bandwidth, processor and memory. It also provides a separate of presentation from content, which means when a page is being displayed on a different device, it can either choose to use the default style sheet, use a style sheet specifically for that device, or use no style sheet at all. The disadvantage is that it currently only supports a few browsers.

4.8.4 WML

WML (Wireless Markup Language) [35] is an XML dialect markup-language specified by the WAP-forum to be used in poor network connections. Its bearer can be on top of standard data-link protocols such as GSM, code division multiple access and time division multiple access and provides a complete set of network communication programs comparable to and supportive of Internet's set of protocols. The disadvantage with is the lack of HTML support, which has resulted in little content to the different WML-browsers. Currently, the WML is in its second version with support for the XHTML Basic.

4.8.5 CHTML

CHTML (Compact HTML) [36] is a subset of standard Hypertext Markup Language (HTML) specified by the Japanese company Access Company Ltd adapted for use with small computing devices such as the I-mode cellular-phones and personal digital assistants (PDAs), and smart-phones. Because handheld devices have display constraints, and limited power, storage, and memory resources, Compact HTML is a stripped-down version of the standard and excludes support for the more demanding features of HTML pages, such as image maps, backgrounds, varieties of fonts, frames, style sheets, and JPEG images.
Due to Compact HTML is based on the HTML standard it will enable small devices to connect to the open Web, rather than merely a section of it. It is expected that, in the future, several different levels of Compact HTML will be developed to adapt to the requirements of different applications.

4.8.6 MMS

Multimedia Messaging Service (MMS) [38] is standardized by the 3GPP (The 3rd Generation Partnership Project) [46] and the WAP Forum. It is meant to be the inheritor of the popular Short Message Service (SMS). MMS is based on the XML vocabulary SMIL that will later be described. With MMS one can make multimedia presentations that contain images, sounds, text and later also video clips. The presentations can be sent either between mobiles or between mobiles and fixed servers or PC's with e-mail capabilities. WAP is used on top of the bearers to facilitate efficient and robust transport of the messages. The multi-media used in a

MMS document can either be stored as a URI pointer [47] or be stored in the document by MIME (Multipurpose Internet Mail Extensions) specification [48].

4.8.7 SMIL

SMIL (Synchronized Multimedia Integration Language) [49] was developed by a group coordinated by the W3C and included representatives from CD-ROM, interactive television, Web, and audio/video streaming industries. The first public draft of SMIL was released in November 1997 and is now in its second version. It is based on XML and designed to control complex multimedia presentation.

SMIL allows Web site creators to be able to easily define and synchronize multimedia elements (video, sound, still images) for Web presentation and interaction. It provides a vocabulary that lets creators combine media types separately but coordinate their timing. Each media type is accessed with a unique Uniform Resource Locator (URL), which means that presentations can be made of objects arriving from more than one place and that objects can easily be reused in multiple presentations.
SMIL also lets the "producer" store a media object in multiple versions, each with a different bandwidth so that a lower-bandwidth version of a Web page can be sent to users who need it.

SMIL is supported by a range of applications and browsers: Real ONE, Internet Explorer, X-SMILE, Microsoft Media Player, Flash, and a selection of mobile phones have support for it.

4.9 Markup-language oriented Internet Client Applications

There exist a number of different browsers made for different needs. I focus on Internet browsers that are widely used that supports wired and wireless devices.

HTML browsers
Today the major browser is Internet Explorer, Netscape and Opera. They all support text, image, video, sound, animations, plug-ins, and client-side scripting and is used on both wired and wireless devices.
- Internet Explorer can be run on stationary computers and laptops that run Microsoft Windows or Macintosh, Windows CE, Pocket PC's, and Macintosh
- Netscape can be run on stationary computers and laptops that run Microsoft Windows or Macintosh, Linux/Unix which includes mobile devices
- Opera can be run on stationary computers and laptops that run Microsoft Windows or Macintosh, Linux/Unix, Symbian, OS/2, which includes mobile devices

SMIL browsers
There are some SMIL browsers, which are supported by different devices and Operating Systems. Among them are
- X-SMILE (java-based)
  http://www.x-smiles.org/
- SMIL Player (ARM, Pocket PC)
  http://www.inobject.com/
- MMS phones (supports SMIL Basic)

WML browsers

- KBrowser (multi-platform, J2ME-based)
  http://www.4thpass.com/
- WAP browser (Palm OS, EPOC, Win CE, OSE and REX. Ericsson, Hyndai, SK Teletech and LG)
  http://www.ausys.se

MMS browsers

- New mobile phones supports MMS

4.10 Server-side development technologies

In accordance to Figure 5 "content adaptation architectures", the server needs logic to distribute adapted content to different devices.

Historically, one of the first methods to activate logic to web-content was by using the web-application interface CGI (the Common Gateway Interface). Later, more general platforms have come, and many great products have embedded these new platform technologies such as IBM Websphere, and web-applications from Lotus and Oracle.

Hence more general open-source web-scripting languages such as JSP, Microsoft .NET, and PHP have been introduced to strengthen weaknesses to the older server-side technologies. Then more specialized web-scripting tools have been evolved in order to support for instance device-independence and system-independence.

I want to describe these technologies, but have the main focus on the Java technology because I have better knowledge of that technology.

First, I want to describe the interface technologies; CGI and Servlet technology, and then I describe the embedded scripting languages. Finally I want to describe scripting-languages and architectures, which are specialized in the device-independence approach.

4.10.1   CGI

One of the first and leading methods for creating web applications was CGI (the Common Gateway Interface) [50]. It is not a programming language, but provides an interface between the web-application and the web-server. The CGI-interface supports all programming languages that are installed on the specific server. The most used programming languages are Perl and C/C++. Because of security reasons, the web applications are default placed in the dedicated directory, "cgi-bin".

The strength to CGI is the standardized protocol, but the weakness is a poor process handling. CGI creates a separate process for each user request and terminates them after the data transfer to the client is completed. This makes CGI not suitable for applications with high traffic, which means a large number of client requests.

Another two weaknesses to CGI is the difficulty of separating presentation from logic and the handling of sessions. It is hard to separate presentation from logic because CGI code often contains embedded markup-languages such as HTML. The session handling is not supported, and the programmer has to use other ways to temporary store the user information. To solve this problem, developers often use cookies, hidden form fields, or storing the information in databases.

4.10.2   Servlets

Servlets is not a programming language, but the Sun Microsystems' [51] solution to improve the poor process handling in CGI. The Java Servlet Technology [52] is Java code run in a server application that answers the clients such as CGI are using HTTP.

The strength to a servlet is that it is loaded into memory first time it is initiated and is not reloaded on every new request. Servlets can then be called many times without having overheads of creating new processes as in the CGI environment. In addition the servlet technology is threaded and handles multiple requests simultaneously.

The weakness by using Java is a slow startup on java applications. When it is loaded to the memory, the application is going fast compared to other server-side programming languages. In addition, it does not solve the difficulty in separating the content from the presentation.

The strength to the servlet technology is that it is based on the well-known Java API [53] and that the language is used on both server-side and client-side, and that it is platform independent. This means that there is easy to develop applications that can be fully functionally on many computers as well as it gives broadly possibilities to exchange applications or libraries.

One of the most used servlet engines is Apache Tomcat [54]. It provides an easy interface to develop servlets or new web-application systems.

4.10.3   Embedded Scripting Languages

Embedded scripting languages were originally done to avoid the problem in CGI and Servlets that the application code often contained HTML. Embedded scripting languages are HTML with server-side code inside. There are a huge number of these embedded scripting languages that can be used in HTML, XML, and WML etc. By embedding the programming-code, it

made it possible to use common web developer-tools to build a web-site with a view to the design. Another advantage is that the code could easier be used on the whole web-site, not in dedicated directories, such as cgi-bin.

The disadvantage with code embedded on a page is that it is hard to share code or resources between pages. This problem has been fixed in the most new languages.

The most used embedded languages are Active Server Pages, PHP, Java Server Pages, and the new eXtensible Server Pages.

- ASP (Active Server Pages) and .NET [55] is developed by Microsoft [56], which supports many scripting languages. The VBScript and Jscript language is mostly used, because they are included as standard. On the Windows Platforms ASP is based on COM, Microsoft's Component Object Model, which means it can use COM components. By using COM components, it is easy to integrate ASP with other of Microsoft's applications.
- PHP (Hypertext Preprocessor) [57] is an open-source embedded scripting language based on C and Perl. The strength to PHP is that it has a big library of free components such as database, xml, e-mail support, and image transformation support.
- JSP (Java Server Pages) [58] is based on Java where the pages are transformed into Java Servlets and compiled which makes them very efficient but also means they generate rather confusing error messages. JSP can also use Java Beans that is objects like COM in ASP. This provides a better support for separating presentation from logic.
- XSP (eXtensible Server Pages) [59] is the new generation of markup languages that is developed by the Apache group [60] used in Cocoon [59]. XSP allows java code embedded in XML files. The advantage is that the included code uses XML syntax. This makes it easier to separate the content from the logic and presentation. The XSP is similar to JSP and has to be compiled into a Java Servlet.

4.10.4   Componentization

The main problem in some embedded programming languages such as JSP and ASP is the difficulty in separating logic from presentation which means problems when distributing content to different sized devices or devices with different technological capabilities. There have been attempts to solve this by making additions to the languages. The ASP technology has not reached this, but there are two Java technologies that propose this; custom tag libraries and JSP's with Java Beans:

- Java Custom Taglibs [61] are composed of a set of custom tags. These custom tags help separate presentation from implementation. Such custom tags are used by replacing common items of logic in JSP pages. The Custom Tag Library Extensions (CTLX) makes it possible to use the Java Taglibs in popular web publishing tools.
- Java Beans [62] is a Java class that has a number of properties. The class is made as a reusable component. This simplifies much develop-work since it can be used in different web-services.

4.10.5   XML Transformation

To transform XML to different output formats such as Markup-languages, it exist a number of solutions. The most used are:

- XSLT (eXtensible Stylesheet Language Transformations) [63] is an XML-vocabulary specified by W3C. Its purpose is to transform XML into other formats. XSLT is a stylesheet expressed in a XSL stylesheet. The advantage is the conversion, but

- SAX (Simple API for XML) [64] is a way of making XML data accessible in Java. It is an event based XML parser that processes the document sequentially. This is very memory efficient, as the parser does not need to keep a copy of the complete document in memory. SAX is useful because it may be simpler to specify certain types of transformations in a procedural language like Java than in a declarative language such as XSLT.

- DOM (Document Object Model) [65] is a way of accessing XML and HTML documents. A DOM parser uses a tree-based representation of an XML document and can interface to a number of languages such as Java or C++. Unlike SAX, it usually keeps the entire document in memory but it can support more complex transformations. DOM can be very expensive in terms of the memory required to hold the documents and hence the speed it can then process those documents.

4.10.6   XML/XSLT web-server architectures

Sun [51] has proposed three methods [66] for using JSP with XML / XSLT in web-server applications: single pipeline, multiple pipelines, and a combination of them.

In the single pipeline approach, the JSP pages are using XSLT stylesheets to generate the client-specific markup language from an XML source. Each type of client requires a different stylesheet.
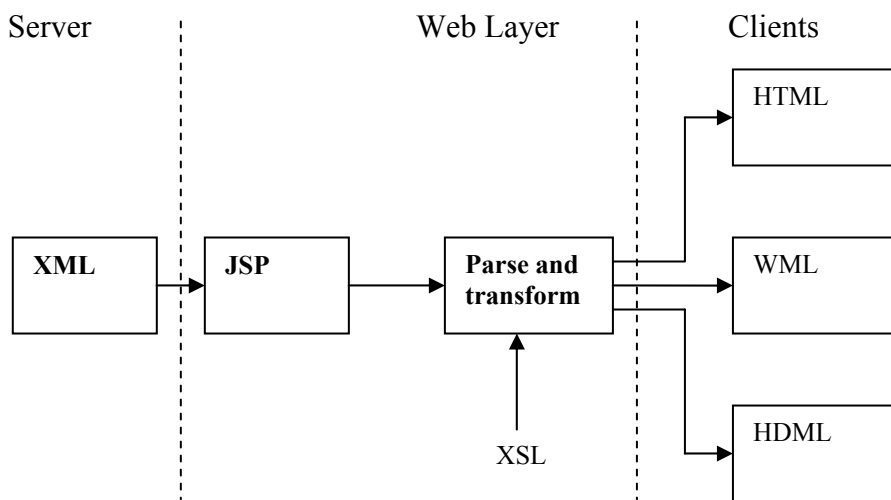

Figure 10 Single pipeline

The multiple pipeline approach uses a set of client-specific JSP pages to generate output. This might control device specific facilities.
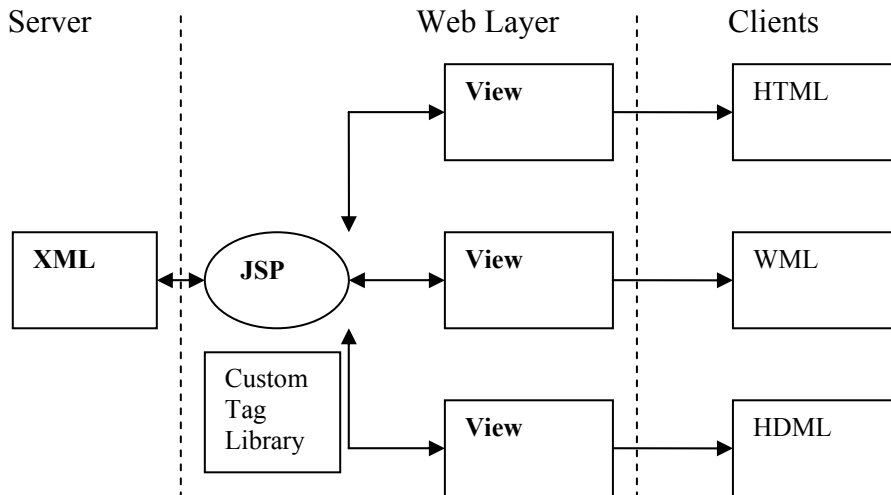
Figure 11 Multiple pipelines

The advantage to both architectures becomes a fact if the logic is developed per device and not per page.
A combination of both single and multiple pipeline approaches is supported which can help developing content to different markup-languages.

Among technologies that realize the architectures are Cocoon, AxKit, and DICE. Both try to support so the web designers can focus on web site design, content developers can work on a purely content basis, and the webmasters focus on their core competencies.

- Cocoon [67] a publishing framework for multiple target devices. It acts like a web server and provides a sophisticated method to compose the site-map and how the data transformation should be done to either the individual page or to groups of pages. It uses XSP, the Extensible Server Pages Language, as described above. Like other embedded scripting languages, XSP suffers as content and logic are placed in the same file. Cocoon users have suggested that stylesheets could be used to separate data from logic in XSP pages.
- Axkit [68] is an XML Application Server for Apache web-server [69] based on the programming language Perl [70]. It supports XML, XSLT, XSP, and XPathScript that is an XML transformation language based on Perl and XPath. AxKit makes the separation of content and presentation easily by using a pipelining technique that allows content to be converted to a presentable format in stages, allowing certain platforms to see data differently to others.
- DICE [71] (Device Independent Content Engine) is also similar to Cocoon and provides an easy way to adapt content by using the CC/PP and UAProf profiling. It is on top of a Java Servlet, and therefore, has access to the whole Java programming family.

The advantages by using XML and XSLT for device specific delivery are:
- Includes delivery to other output-data formats such as HTML, CHTML, TEXT, SMIL, PDF, and etc.
- Differentiation between the different devices capabilities to display content as it was meant to be displayed.

- Provide mechanisms that can reorganize, modify or split the content into sub-pages in order to fit content for small-displayed devices.

The disadvantages by using XML and XSLT for device specific delivery are:
- The XSLT is a quite new approach for web-developers, which might give unstructured solutions. It let the developers make their own set of tags and then write stylesheets to style XML for different devices.
- By using one stylesheet to all pages to every device might give limitations and freedom to the designer of the content.
- There are many authorizing tools, but there is lack of WYSIWYG (What You See Is What You Get) editors. This might confuse content oriented authors but give advantages to the developer.

5   Prototype: Multi-channel service

The prototype illustrates how content can be distributed through a multi-channel service. My interest was to investigate how far it was reasonable to deliver content to an unknown number of different devices with a measurable media type quality. I have focused the work on

1. Just-In-Time learning service meant for professional business users
2. Separating the content, the logic, and the presentation
3. Using technologies that support content control negotiation between the server and the displaying devices.
4. Let the content creator define technological requirements to each media type and specify which media types that are related.
5. Content-transformation to the most common markup-language oriented Internet client applications
6. Using standardized technologies

I made some assumptions and limitations:
- I am not focusing on the XML vocabulary used to store the content and the storage architecture.
- I am not focusing on design and layout of the web page, but the underlying technology
- The multi-channel service has support for browsers that handles one of the following markup-languages:
    - HTML
    - SMIL
    - WML
- I assume that the Quality of Service is ideal.

5.1   The Just-In-Time learning service and its construction

The prototype is a framework of a multi-channel service and includes one example. This example is an instruction-module on how to take good shoots with your automatically camera. By using a mobile phone, palm, laptop, or stationary computer the content can be received instantly. Such services touch the adult-learning theory described in chapter 2.3.1. This prototype is a limited example of a multi-channel service and is strongly limited. Interaction facilities are not supported except navigation and links. This version supports the media types: text, images, video, and sound.

The technologies used in this prototype that is described the next chapters are:
- *Content*
    - XML
- *Device profiling*
    - DELI which supports CC/PP, UAProf and HTTP header field
- *Transformation*
    - Plain XSLT technology. Using Cocoon to process the transformation
- *Serialization*
    - By using Cocoon's serialize technology.

The mode of the service is described in Figure 12 and starts by a request from the user. This request can be e.g. a click on a link "how to shoot portraits in sunlight". The device, that receives the request, sends then a request to the web-server. This request includes:

1. A client-identification (by HTTP header field) and also a client-capability-profile (if the browser supports either CC/PP- or UAProf profile)
2. The URI (pointer to the requesting page).

By this foundation, the web-server or the web-server extension process the request as it is programmed to do. My setup is described more thorough in chapter 5.3. This process includes transformation which can be XML/XSL transformation, adding logic to the pages such as XSP or JSP, and using device profiling to adapt the content to fit the device. After the transformation a serializing is done to add necessary header information to the content. This serializing includes both media types and markup-languages.
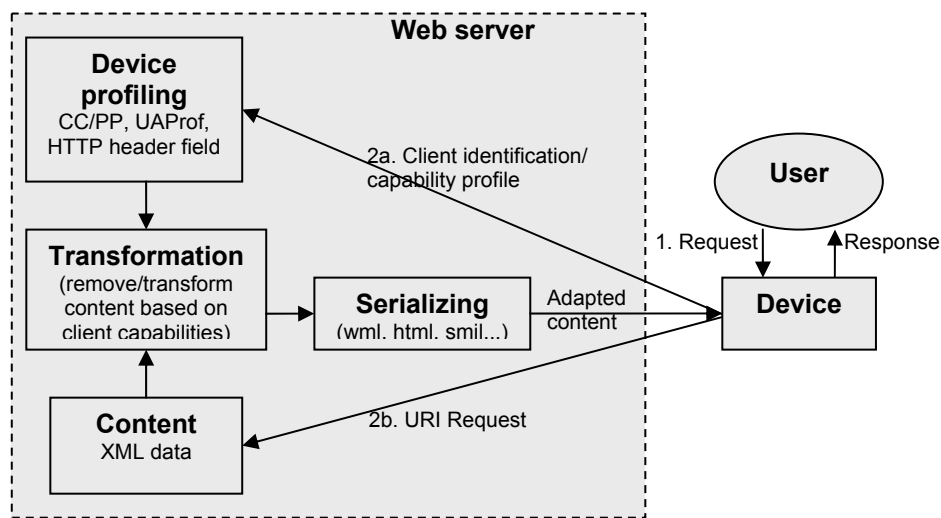
Figure 12 The mode of the system

The next chapters describe the web-server technologies more thorough.

5.2 Content

The instruction-module consists of the content-types: text, picture, sound, and video. All media types, except text, are stored as proprietary files. The XML-file includes all text, the order of media types and the structure of the course.

The structure or the XML-dialect is not based on a familiar existing markup-language. I used such dialect to not bind me to a specific browser or markup-language and my second reason was that I am not focusing on this.

The XML vocabulary used is based on a slide XML vocabulary used in a Cocoon-example but is modified. This vocabulary provides segmentation of content like chapters and sub-chapters. The root xml-element is <slides> which is includes title, overview and all slides (sub-pages) of the module. Content is tagged by <media> and the type of media is placed in its 'type' attribute. The minimal attribute requirements to the media-tag is

- Text
  <media type="text">Photos </media>
- Image
  <media type="jpg" src="image.jpg" alt="How a zoom function"/>
- Video
  <media type="mpeg" src="image.mpg" alt="How a zoom function"/>
- Sound
  <media type="au" src="zoom.au" alt="The click of a photo-camera"/>

Below is an example of two slides and the presentation of the first page in a web-browser.

```
<?xml version="1.0"?>

<slides>
  <title>Tips for Using a Point and Shoot Camera</title>

  <authors>
   <person>
    <name>Philip Greenspun</name>
    <email>philip@greenspunk.com</email>
    <url>http://philip.greenspun.com</url>
   </person>
  </authors>

  <overview>
In this mini-course you will learn how to use light, lens, film, exposure and the camera
<media type="jpg" src="venice-bridge-of-sighs-84.jpg" alt="Picture of Venice bridge. Dark and
rainy morning"/>
</overview>

  <section>
   <title>Sun: sunlight</title>

   <slide dur="1000">
    <title>Overcast Skylight</title>
    <media type="text">A high overcast is perfect for a lot of photography. A studio
photographer would think of this as "the mother of all softboxes". If you want to capture
architectural details, an overcast day lets you do it without shadows obscuring anything.
Overcast and/or rainy days are also the times to go into the forest and take pictures of
trees. The one bad thing that you can say about an overcast day is that a big white sky makes
a very bad photographic subject. Try to make sure that your photos have hardly any sky in
them.</media>

    <media type="text">Overcast skies are slightly more blue (7000 degrees Kelvin) than the
color temperature for which daylight film is designed (5500 K; a mixture of direct sun and
skylight). Officially, the Kodak Professional Photoguide will tell you to use an 81C warming
filter. I wouldn't bother unless you are photographing clothing for a catalog. [For
comparison, open shade from a clear blue sky is 11,000 or 12,000 degrees Kelvin and requires
an 85C filter.]
    </media>

    <media related="1" type="jpg" src="handbills-38.1.jpg" alt="Handbills.  Manhattan 1995" />
    <media related="1" type="text">If you wait long enough in New York City ... someone will
probably steal your camera. So maybe it is best to just shoot in whatever light you can find.
Here I used the fill flash on my point and shoot camera. </media>

    <media related="2" type="text">Below the town of Bomarzo, Italy (1.5 hours north of Rome).
This was the park of the 16th century Villa Orsini and is filled with grotesque sculptures.
Rollei 6008, Fuji Velvia, Zeiss 50mm lens, tripod. Probably f/22 and 1 second. Note that part
of the foreground is unsharp. There wasn't quite enough depth of field. Note that the
composition completely excludes the uninteresting overcast sky. </media>
```

```
    <media related="2" type="jpg" src="leaning-parco-small.jpg" alt="Leaning Parco"/>

  </slide>


  <slide dur="1000">
   <title>Twilight</title>
    <media type="text">There is no reason to put the camera away after the sun goes down. In
fact, you can usually get your best pictures then. You'll often need exposures of 30 seconds
or longer, however. Here are some photos from Chapter XV of Travels with Samantha as
examples...
    </media>

    <media type="jpg" src="ruby-after-sunset-37.1.jpg" align="center" alt="Picture of Ruby
Beach after sunset.  Olympic National Park (Washington State). Twilight"/>

    <media type="text">Arches National Park (Utah).</media>
    <media require="color" related="3" type="jpg" align="left" src="arch-frame-pre-sunset-
58.1.jpg" alt="Arches National Park (Utah)."/>
    <media related="3" type="text"> At left, before dark. </media>
    <media require="color" related="4" type="jpg" align="right" src="arch-frame-pre-sunset-
58.1.jpg" alt="Arches National Park (Utah)."/>
    <media related="4" type="text"> At right, after dark</media>

  </slide>
</slides>
```

Figure 13 Example of XML-content

Figure 14 Example of the web-presentation

5.3 Content adaptation techniques in Cocoon

Cocoon provides a perspicuous way to separate the content and the logic from the presentation. (Cocoon is thorough described in chapter 4.10.6 and 5.3.) It uses pipeline architectures described in chapter 4.10.6.

The Cocoon's basic mechanisms for processing XML content through pipelines are:

-   Dispatching based on **Matchers**.
    A Matcher is the start point of an xml pipeline. It attempts to match an URI with a specified pattern for dispatching the request to a specific processing pipeline.

- Generation of XML documents from files, logic, databases, objects or any combination through **Generators**

- Aggregation of XML documents through **Aggregators**.
  An aggregator is used to merge an XML structure (from for example an xml-document) into another XML structure

- Transformation (to another XML, objects or any combination) of XML documents through **Transformers**
  A transformer is used to map an input XML structure into another XML structure like by using XSLT to convert XML to WML.

- Rendering XML through **Serializers**
  A serializer is the end point of an xml pipeline
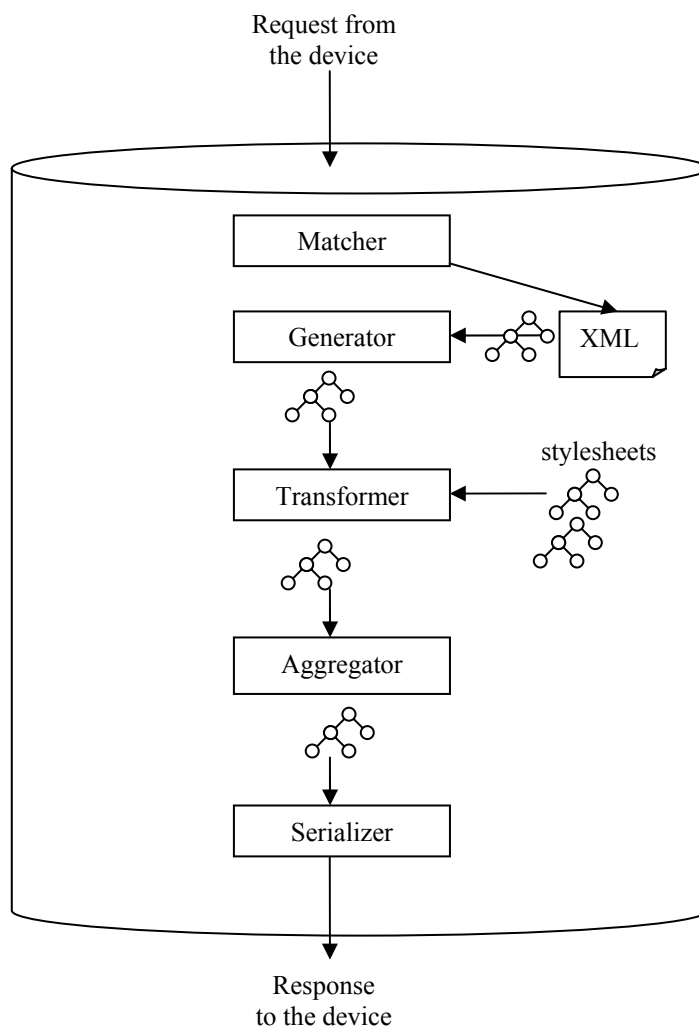
Figure 15 the Cocoon's basic mechanisms for processing XML content through pipelines

Cocoon pipelines are specified in an XML file named "sitemap.xmap". A Site-map can mount other sub site-maps, which gives scalability and portability. My multi-channel service used

one pipeline per device and all are specified in one single site-map. Beneath, the setup of the WML-pipeline is shown.

```
<map:match pattern="*.wml">
      <!-- get the content from an xml source -->
      <map:generate src="content/{1}.xml"/>

      <!-- extract one slide from the slide-collection-->
      <map:transform src="stylesheets/slides-select-slide.xsl" type="xslt">
            <map:parameter name="use-request-parameters" value="true"/>
      </map:transform>

      <!-- identify the device's capabilities and transform media types -->
      <map:transform src="stylesheets/slides-capabilities_framework.xsl" type="xslt">
            <map:parameter name="use-deli" value="true"/>
      </map:transform>

      <!-- remove media types that is dependent on other but cannot be presented due to
       limitations in the device -->

      <map:transform src="stylesheets/slides-relations framework.xsl" type="xslt"/>

      <!-- transform the XML code to WML code -->
      <map:transform src="stylesheets/slides wml.xsl" type="xslt"/>

      <!-- add wml content-type header -->
      <map:serialize type="wml"/>
</map:match>
```

Figure 16 The wml- pipeline and the configuration in Apache Cocoon

5.3.1 Transformers

Most of the content adaptation job is done through the transformers. My prototype is using four transformers that are serial-processed:

1. Split the big multi-slide XML-document down to a single slide that the user requested by providing the URI. This processing is done through XSLT.
2. Modify/remove media types that cannot be handled by the device. This is processed by using the XSLT and the DELI-framework.
3. Remove all media types that are unintelligible because they are related to other media types that are removed in the previous transformation.
4. Transform the adapted XML-document to the chosen markup-language by a XSL style sheet.

The second and third item in the list is where the content adaptation occurs. As mentioned, I have used DELI to provide the device's technological capabilities. DELI is a Java library that allows Java servlets to resolve HTTP requests containing CC/PP or UAProf information. HTTP requests header, CC/PP, and UAPROF are moreover described in chapter 4.5. By comply with devices that do not support CC/PP or UAProf DELI uses the HTTP Request Header Field and the HTTP/1.1 Accept Header Fields.

My prototype supports a range of UAProf devices and CC/PP devices and an overview of these is listed in Appendix.

I have utilized DELI to make a framework on how media types must be presented as regards to the device's technological capabilities. By this I have provided a set of attributes that the author can use:

- The relevance to other media elements
    i. Such as both a text-element and a picture-element must be shown to give any meaning.
        a. Indicated by the attribute "related"

- The minimum requirement to a media element
    i. Such as an image-element claims a color dependent device.
        a. Indicated by the attribute "require"

This framework supports the media-richness theory that is described in chapter 2.6. Using many media types in content there is a need to provide a mutual meaning. Therefore, I have proposed that if there is lack of a media type that relies on another media type, it should be removed.

The prototype is using attributes in the content to let the content composer specify which media types that depends on each other and the minimum technological requirement on a media type. For instance a text describing parts of a colorful image requires a color-capable device. The framework beneath is an approach to this functionality:

Media-type	Relation to media type	Optional requirement
Text	Image	Image, Color
Text	Video	Video, Color
Text	Sound	Wave-sound
Image	Text	Image, Color
Image	Image	Image, Color
Image	Video	Image, Video, Color
Image	Sound	Image, Wave-sound, Color
Video	Text	Video, Color
Video	Image	Video, Image, Color
Video	Video	Video, Color
Video	Sound	Video, Wave-sound, Color
Sound	Text	Wave-sound
Sound	Image	Color, Image, Color
Sound	Video	Color, Video, Color
Sound	Sound	Wave-sound

Table 6 Framework of media type requirements

If the device cannot present a media type it said it could, an alternative text is displayed.

5.4 Testing results from the prototype

My prototype consists of server software and browser content. The server software, Cocoon and DELI, was tested and installed on both Linux and Windows 2000 servers. The multi-channel service example was developed by using XML and XSL. The channels I tested were:

- Laptop using HTML capable browser
- Mobile phone using WML browser
- PDA supporting SMIL browser
- Laptop using SMIL

Due to my limited access to mobile network during my work on the thesis in the US, I have only tested the browsers and devices by using simulators on my Laptop.

The devices I tested were:

- Internet Explorer 6.0
- Netscape 4.8 and 6.0
- Opera 6.0
- Phone.com UP.SDK 4.01 (used in many Siemens mobile-phones)
- Ericsson R320, T68 simulators
- X-Smile (cross-platform SMIL browser)
- RealOne Player (A common used windows media player supporting SMIL)

My purpose to do this test was firstly to clarify if the DELI technology worked and secondly which devices that supported the CC/PP and the UAProf technology.

I did two tests. The first test was by echoing the device's technological capabilities back to the device to see what capabilities the different devices had. The second test was to see if a Just in Time service provided the content as it was thought.

By using the echoing test, I found that following devices supported CC/PP:

- Internet Explorer 6.0
- Netscape 4.08 and 6.0
- Opera 6.0

The echoing test gave different results to each browser and the Internet Explorer provided most capability information. Opera and Netscape provided least capability information. The browsers did not deliver plug-in capabilities, which I find weak. The capability information results are described in the appendix.

By investigating the UAProf capabilities, I found that mobile phones using UAProf provided much more information compared to web browsers using CC/PP. By using simulator I found that UP.SDK browser provides much capability information, which is described in the appendix.

I found that SMIL browsers did not support the CC/PP or UAProf. It surprised me that SMIL browsers had no support for SMIL because W3C has standardized the all of these standards. Hence, W3C also stresses the need to provide the device's capability to the server.

The second test I did was by testing a multi channel service on different devices. I find the results good except the SMIL results. Therefore, I had to turn off the DELI technology on such devices. The SMIL technology is quite new, and that might be the reason that SMIL is not supporting CC/PP or UAProf yet.

I found out that DELI received and delivered the device capabilities to Cocoon without problems on the devices that supported CC/PP or UAProf. Hence, DELI provided standard capability information to the devices that only supported HTTP header field.

5.5 Prototype improvements

My prototype has investigated in these technological challenges to multi channel services:
- How to deliver content pieces to different devices supporting different markup-languages.
- What must be done technologically if related content pieces cannot be displayed? Example: text describing a picture.

What must be done if media types that need special facilities cannot be achieved? Example: an image needs devices that support color presentation.

Due to my limited prototype, it has a great potential to be improved as a Just in Time service. These improvements include especially learning support methods, but also technical which are described in the next sub-chapters.

5.5.1 Improvements to support learning:

The most important improvements I find to support learning in my prototype are:

- Adding some type of repetition service on learned subjects. Learners can then improve the memorizing process.
- Adding test service on learned material to see what the learner knows.
- Adding an intelligent guidance tool which could quickly lead the learner to the most important subjects and matching his knowledge about the subject.
- Adding search capabilities to make the exploration faster through the learning material.

A discussion about what improvements that should be considered in my prototype and more general multi channel services are discussed in parts of chapter 6.1.

5.5.2 Improvements to the technical solution:

There are some important technological improvements I want to emphasize. Further work should consider a development on the server logic. This work could be to improve the intelligence that gives content editors a better control to the quality of the distributed learning. By quality I refer to how the service can prevent distributing content pieces without meaning.

Hence, the prototype has limitation on the structure of the content. A more well-known and standardized XML language should be considered. This language should also support content editing in WYSIWYG (What You See Is What You Get) environments.

At last the prototype could also have a further development on the interface between other e-learning services. Such interface could be the IMS Project described in chapter 4.8.2.

6 Multi-channel service discussion

In this thesis, I have explored both theoretical and technological aspects of creating learning environments and solutions suitable to assist in learning situations. The great challenge is to combine innovative technology with learning methodology that serves the goal: help users to learn better. In this part I want to base my discussions on the e-learning theory, the technological aspects, and the prototype work. A discussion on e-learning theory and the consequences this gives to multi-channel services are discussed in sub chapter one. The next two sub chapters are focusing more on technical challenges to the multi-channel approach.

6.1 E-learning implications for Multi-channel e-learning services

In this chapter I will be discussing e-learning implications for multi-channel e-learning services. To make a connection between the theory and technical multi-channel challenges, I want to use the prototype as a basis for discussion and relate it to the theory discussed in chapter 2.

Regarding the chapter of theory I outlined implications from theory on my project. These implications were highlighted in yellow boxes. Further, I will discuss each of these boxes and their relevance for the usability of my prototype. The discussion follows the topics more or less in a chronological order. I refer to the theory chapter for a more detailed discussion of the underlying theory.

6.1.1 Multi-channels, better learning?

Topics to consider:

- Define the four media features; technology, content, situation, and symbol systems, in the learning service in order to improve the mental processes.
- Map and limit the number of symbols used in communication in order to serve most skills and styles of learning.
- Consider how the context can influence learning.

My initial finding after developing the prototype is:

- The major problem is to adapt content to the learner, the specific learning styles, and the situation. My service is able to deliver multiple content via multiple channels, but not intelligent enough to cater to the specific need of the learner. The adaptation is focused on the functionality of delivery rather than use. Delivery of content is different from delivering the right content to the specific learning situation.

As discussed in the theory section, the success of any learning service is its ability to adapt to the specific learning situation and to make use of a set of symbols via multiple media channels that corresponds with the symbol system of the learner. This match is supposed to improve the learning process. The ability to adjust and adapt to the symbol system of the user, requires a pretty elaborated service to profile and match the user.

Within the limitation of this thesis, I have not been able to develop such intelligence into the service. What I have been able to achieve via the service, is technology that supports construction of learning material that makes use of different media types (such as text, sounds, images, and video) and to deliver it via multiple devices with methods to control the quality of content. It might be reasonable to conclude that this effort can improve construction of content and improve the ability to adapt content to the symbol system of the user.

From a technological perspective the CC/PP and the UAProf content negotiation technique between the device and the server specifies the channel's and the medium's capabilities. Devices that do not support this kind of negotiation to the server cannot utilize the prototype's intelligence. Such limitation prevents the ability to match the symbol system with the user and to improve the learning process.

6.1.2 Can technology support the user's problem in a specific situation?

Topics to consider:

- Base the e-learning service to focus on problem-solving
- Implement a support-system; live or virtually, which can guide or help the learner in the learning or problem-solving situation.
Mobile learning technology seems to be suitable to provide "just in time", "on demand based" knowledge to problem solving. My prototype could provide technology that enables users both to use knowledge in "live situations". The problem is to equip this type of learning services with more intelligence to recognize the situation in which problem solving takes place.

The immediate advantage of mobile based learning services seems to be clear:
- The ability to learn where you want, when you want.
The prototype exemplifies how this could be done by using multiple channels to improve on delivery of content adapting to different devices in different situations.

The prototype is meant to focus on problem-solving, especially when there is need to learn something instantly. On the other hand the prototype does not include any kind of guidance in the learning material and might cause problems for the user to navigate in the service. An underlying model of how to structure and arrange the learning content might improve this problem (for example a simple kind of curriculum or a more elaborated structure of content). An assistance-tool is important to multi-channel services to prevent any confusing to the learner and to make it easier to find the right information. This might compensate for the lack of "user intelligence" in the service and make use of the learner's skills and capabilities to navigate in the content.

6.1.3 Can technology support changes in the organization culture?

Topics to consider:

- Add functionalities that motivate the learner to use the e-learning service
- Relate the topics to the learner's experiences
- Know that organizational structures are always changing and
  take this into consideration when developing learning services.

The learning application has to be relevant to the learner, and flexible to the changing situation of the user. I focus on experienced learners, primarily professional users in business environments.
While the application in some sense meets the need of motivation of the learner (just in time, on demand) and relevance to the learner (bringing learning into real life situations), the difficult part is to adapt the application to changes in organization. This adaptation cannot be made by technology innovation, but a deeper understanding of how the specific organization learns and relates to use of technology in learning situations.

While developing the prototype, I related the information in the service to support a task-, and problem-centered learning orientation. The motivation of the learner often depends on the fact that the content is good, relevant, and that the technology provides access to the content in an intuitively way. This prototype does not provide many facilities, and multi-channel services should include tools such as interactivity, search, tests, forums, and etc. to improve the learner's motivation.

Multi-channel services should also consider being flexible in order to support changes to the learner or to the organization culture. Subjects, functionalities, and devices should be related closer to the learning style of the user and to the organization culture that support use of this type of technology to learn. In e-learning, technology is only a supporting tool to the user's learning process. I think e-learning services cannot be ideal to suit every type of organization culture or learner styles.

6.1.4 Are different learning methods improving different learning outcomes?

Topics to consider:

I have used Bloom's taxonomy to emphasis the point that different types of learning methods provide different types of learning outcomes.

Providing information could be done through the type of service I have described in this thesis. I am more uncertain if this type of application can provide learning at the higher levels in the learning taxonomy. The limited ability of interaction and collaboration in my prototype make it functional for information distribution (as a form of learning support).
To provide higher levels of learning in the taxonomy, the service should be integrated with other types of learning technology and more traditional ways of learning like group work, classroom education, and peer collaboration.

By using Blooms framework, the user's learning process can be improved. Multi-channel services can improve this by:
- Adding repetition-service on learned subjects (level 1)
- Adding tests on learned material (level 2)
- Providing Just in Time (JiT) learning that the learner can use in a practical situation (level 3)
- Providing background-information so the user can evaluate independently the relevancy of data to the specific situation (level 4)
- Be a tool that help the user to develop independently new things such as documents or projects (level 5)
- Providing the user different information on the same subject to let he/she independently evaluate/judge what information that is relevant (level 6)

6.1.5 Maintaining the media richness by using different capable devices?

There are still some major considerations left:

- Define how technology or the medium affects the message.
- Define what amount and type of information is required to make learning happen.

We know little about use of mobile technology in learning. The field is under constant research and investigation. What we do know is that use of content via mobile devices is radically different from stationary PC's. Some of the differences are:
- Content needs to be more targeted to the specific need of the user.
- The amount of content has to be more limited.
- Handheld devices provide lower quality output of media formats like video and sound.
There seems to be enough evidence to support the idea that the medium (mobile devices) might strongly influence the perceived message by the user. This also refers to the problem of sharing complex symbol systems making use of different media formats via this type of devices.

Multi-channel services must adapt the content to the device. My finding has pointed up that firstly, there is a need to provide a control mechanism that enables the editor to relate content pieces to other peaces. Secondly, such services also need mechanisms that evaluates whether the content can be displayed on the device at hand. Hence, technology should facilitate distribution of media rich content via multiple channels to different devices. I have not evaluated how the perception of content is influenced by the medium. This goes beyond the framework of this thesis, but I think it is a major topic to take into consideration relating to the usability of such services.

6.2 Device- and Network-capabilities

In this thesis work I have explained technological concepts on how content pieces build from media-types can be distributed through different channels to many devices. A channel is defined as an architecture that can carry content to a device through specific interaction software.

Following channels have been investigated in this discussion:
- Mobile phones using the WML markup-language
- Laptops using the HTML markup-language
- Palm pilots using the SMIL markup-language.

In my exploration to these channels, I have found out that the content control-technology is very important to adapt content to the device's capabilities. This technology is not broadly used by developers yet, and services will probably come soon. The reason why this technology is not broadly used might be that the content control-technology such as CC/PP and UAProf is quite new. Most new browsers are supporting it, but most users are using old browsers that do not support this technology which means they cannot utilize the strengths to the content adaptation technology.

If users are using old browsers, the service can use the HTTP Header Field (chapter 4.5.1) to identify the browser software and its version and assume what capabilities the device has. I have found that this method is not comparable to devices supporting CC/PP or UAProf. The HTTP Header Field provides limited information about the browser, which can be difficult to utilize in multi-channel services. Another weakness to this technology is that users can in some instances change these settings manually. The HTML browser Opera supports this manual control of this field.

Regarding browsers, the newer and most used HTML and WML browsers are using this technology. SMIL browsers are currently not supporting CC/PP and UAProf, which prevents the editor of the content to control these by server-side intelligence.

Multi-channel services should support most devices in order to fulfill the idea to provide content instantly on random devices. Due to challenges included to the content control technology, it would be reasonable to limit the number of supporting devices down to the most usual. By limiting them, the quality of content might easier be increased, because it includes fewer device capabilities to support.

A multi-channel service might face problems if the viewing device does not support the content negotiation technologies described in chapter 4.5. This might prevent the service to provide content that gives meaning for the end-user. Examples of these are:
- Layout-confusion
    a. Layout not fitted for the size of the display which reduces the user's ability to quick understand the message.
- Color-confusion
    a. Images that require color-display to give sense
- Sound-confusion
    a. Lack of sound-capabilities which results message without meaning
- Auto-scaling of pictures/video by the device
    a. Scaling by the device which is not controlled by the multi-channel service might result in picture that gives no meaning to the end user.

People tolerate that today's mobile devices are mostly based on text and simple graphics. Next generation mobile devices can display high quality images and video, and require a more stable network with higher bandwidth to display the other media types. It is reasonable to suppose that people want a better display- and sound-quality of the media types when these devices approach.

Different combinations of media types require more of the underlying networks. Such combinations will probably have diverse requirements to delay, bandwidth, and jitter for maximum performance. It is also a fact that the number of formats for each basic media type is growing. Each format is optimized to satisfy a specific need: transfer speed, storing size, or best quality. Naturally, these factors influence each other. Smaller file-size usually means poorer quality.

Live video is one of the most challenging media type. Today, wired devices can provide a reasonable quality but mobile devices provide vary and poor quality:

1. DoCoMo I-Mode [72] has some support for video, but has low frame-rate and poor color depth.
2. GSM/GPRS phones has no support for video
3. 3G devices is expected will display video with a reasonable quality
4. Palm connected with WLAN has a reasonable quality
5. PC/Laptop has broadcasting quality, but it depends on the network bandwidth

6.3 Technology standards

It is clear that technological standards are a significant factor in today's development of multi-channel services. Most developers are using standards in their developed services. Recent years the focus on content control mechanisms such as how to utilize the individual device's capabilities has been important because of the great increase in many devices and browsers.

It is a fact that the content control mechanisms CC/PP- and the UAProf –technologies are quite new and are only supported by the latest browsers and devices. The HTTP User Agent String technology has been used on most browsers and seems not to be removed in the near future. Multi-channel services that base on the CC/PP- and UAProf-technologies in order to provide device-oriented content and presentation must consider how to handle devices that do not support them. There should be no foundation to limit the service, such as limit the design- and the media types, in order to support all types of devices. This can result in poor services, reduced quality of learning, or reduced learner motivation.

Hence, XML and XSLT are broadly used and improve many aspects to multi-channel services. By using stylesheets and transformation, the content can be easily presented through many channels. The disadvantage is the need to customize stylesheets to every channel and in some instances in different parts of the learning. If the multi-channel is supposed to support many devices, the cost increases both on developing and additional development. Therefore, there should be an understanding what such development involves.
The advantages to use XML and XSLT are that DELI profiles, which mean CC/PP and UAProf standards, can easily be included. Hence, XML provides scalability and improvements to communicate c2c (between computers).

The weakness in using standards is a relatively slow standardization process. This slow process might give limitations to the developer such as to utilize facilities in new browsers or new devices. As a result many browser manufactures add special functionalities which reduce the compatibility approach. This prevents multi-channel services to display correctly on some devices.
Even though these challenges exist, standards have been shown to be one of the most important factors to succeed in multi-channel e-learning services. First, standards increase the possibility to make different multi-channel services compatible. Secondly, standards strengthen the ability for new developers to modify or add functionalities later. Finally, standards are not drastically changed or removed from the standardization committees without good reasons.

7 Conclusion

This thesis I have discussed how content can be distributed from a node (server) to an unknown number of devices with different technological capabilities. The focus has been how content can be distributed with a measurable quality to devices in order to improve the user's learning process. Different architectures and official standards of interest have been described and discussed and a prototype was developed to explore some methods to do it.

I have been concentrating on the Internet architecture's ability to support content distribution to different devices which includes these fields:
- The content specification
- The server logic
- The content control negotiation, which is distributing the device's technological capabilities to the server
- The distribution of the content
- The presentation of the content

The focus has been on the content control technology and how the server logic can adapt the content to these devices. Through my exploration, I have found that current standardized technology can provide acceptable supporting methods to distribute content to different devices in case the device can negotiate its technological capabilities to the server.

In my investigations I found that the following negotiation methods can be used: HTTP Header, CC/PP profile, or UAProf. CC/PP and UAProf gives more information about the device's capabilities than the HTTP Header. All these methods are based on official standards which give continuity and compatibility to the systems that use it. If the device cannot negotiate such capability information, the server has no basis to distribute an adapted content to the user. Results from the prototype tests I found that only the newest browsers supported CC/PP and UAProf. This reduced support might give poorer content quality to the old devices that are using today's multi-channel services.

The prototype is using Cocoon and DELI which support the above content control negotiation standards and shows how the quality of content can be controlled by the creator of the content. He or she can relate content pieces or set requirements to the device's technological capabilities. This technology can remove all content pieces that can't be presented on the user's device which can improve a little of the user's learning process.

Hence, discussions have shown that technology must provide many supporting tools in order to improve the user's learning process. The correlation between technology and learning involves a strong complexity such as:
- how to map the many learner styles
- how to adapt the content to the situation of use
- how the medium influence the perceived message by the user
- how the situation of use influence the message
- how to support the changes in organization cultures

Final, multi-channel e-learning services needs a strict use of the official standards and should provide methods to support the many users that still use old browsers. Multi-channel e-

learning services should also include different support e-tools in order to match the user's learning style and the situation of use. To map most of these additions, further work on this thesis should include a usability test of the prototype.

References

1. Gavriel Salomon, *"Interaction of Media, Cognition and Learning"*, 1994
2. Jerome Bruner: http://snycorva.cortland.edu/~ANDERSMD/COG/bruner.html
3. Jean Piaget: http://www.koyote.com/personal/hutchk/piaget1.htm
4. Lev Vygotsky: http://education.indiana.edu/~cep/courses/p540/vygosc.html
5. Constructivism. University of Houston, College of Education, 1999
    http://www.coe.uh.edu/~ichen/ebook/ET-IT/constr.htm
6. Jonassen, D. *"Thinking technology. Educational Technology"*, 1994
7. Jost Reischmann model of lifewide learning,
    http://www.uni-bamberg.de/ppp/andragogik/andragogy/lifewide.jpg
8. Knowles,M. *"The Adult Learner: A Neglected Species"*,
    (3rd Ed.). Houston, TX: Gulf Publishing, 1984
9. Chris Argyris, Donald A. Schön, *"Organizational Learning II, Theory, Method, and Practice"*, 1996
10. NETg http://www.netg.com/
11. Shannon, Claude E. & Warren Weaver, *"A Mathematical Model of Communication"*,
    Urbana, IL, University of Illinois Press, 1949
12. Hsia, H. J., *"The information capacity of modality and channel performance."*,
    AV Communication Review, 19 (1), 51-75, 1971
13. Hsia, H. J. *"The information capacity of modality and channel performance."*,
    AV Communication Review, 1971
14. R. Daft, R. Lengel, *"Information Richness. Research in Organisational Behaviour"*, 1984
15. Trevino, R. Lengel, R Daft. *"Media Symbolism, Media Richness and Media Choice."*,
    Communication Research, 1987
16. Citrix MetaFrame, http://www.citrix.com/products/metaframe/
17. Windows Terminal Services,
    http://www.microsoft.com/windows2000/technologies/terminal/
18. VNC (Virtual network Computing), http://www.uk.research.att.com/vnc/
19. Lotus LearningSpace http://www.lotus.com/products/learnspace.nsf/wdocs/homepage
20. National Instruments, LabView http://www.ni.com/products/
21. Butler, Mark H., *"Current Technologies for Device Independence"*, 2001
    http://www.hpl.hp.com/techreports/2001/HPL-2001-83.pdf
22. HP, Hewlet-Packard, http://www.hp.com
23. HTTP request header field / http/1.1 Accept Header Fields,
    http://www.mozilla.org/build/user-agent-strings.html
24. World Wide Web Consortium (W3C) http://www.w3.org
25. CC/PP Composite Capabilities / Preferences Profile http://www.w3.org/Mobile/CCPP/
26. User Agent Profile http://www.wapforum.org/what/technical.htm
27. WAP Forum http://www.wapforum.org
28. HTTP/1.1 http://www.w3.org/Protocols/
29. HTTP-ex http://www.w3.org/Protocols/HTTP/ietf-http-ext/
30. HP DELI http://delicon.sourceforge.net
31. Resource Description Framework http://www.w3.org/RDF/
32. Microsoft Content Management Server http://www.microsoft.com/cmserver/
33. HTML http://www.w3.org/MarkUp/
34. XML http://www.w3.org/XML/

35. WML http://www.wapforum.org/what/technical.htm
36. CHTML http://www.w3.org/TR/1998/NOTE-compactHTML-19980209/
37. SMIL http://www.w3.org/AudioVideo/
38. MMS http://www.3gpp.org
39. IEEE Learning Technology Standards Committee (LTSC) P1484.17
    http://ltsc.ieee.org/wg17/index.html
40. Docbook, http://www.docbook.org/
41. IMS-project http://www.imsproject.org/
42. IEEE Learning Technology Standards Committee http://ltsc.ieee.org/wg1/
43. Advanced Distributed Learning (ADL) Initiative http://www.adlnet.org/
44. The Aviation Industry CBT Commitee http://www.aicc.org/
45. The Dublin Core Metadata for Electronic Resources http://dublincore.org/
46. 3GPP http://www.3gpp.org
47. URI, http://www.w3.org/Addressing/
48. MIME specification http://www.ietf.org/rfc/rfc1341.txt?number=1341
49. SMIL http://www.w3.org/AudioVideo/
50. The Common Gateway Interface http://hoohoo.ncsa.uiuc.edu/cgi/
51. Sun Microsystems http://www.sun.com/
52. Java Servlet technology http://java.sun.com/products/servlet/
53. Java API http://java.sun.com/
54. Apache Tomcat http://jakarta.apache.org/tomcat/
55. Microsoft .NET
    http://msdn.microsoft.com/library/default.asp?url=/library/en-
    us/dnmapnet/html/mapintronet.asp
56. Microsoft http://www.microsoft.com
57. PHP: Hypertext Preprocessor http://www.php.net
58. JSP: Java Server Pages http://java.sun.com/products/jsp/
59. XSP: eXtensible Server Pages http://xml.apache.org/cocoon/
60. Apache Group http://www.apache.org/
61. Java Taglibs http://jakarta.apache.org/taglibs/
62. Java Beans http://java.sun.com/products/javabeans/
63. XSLT http://www.w3.org/TR/xslt
64. SAX http://www.saxproject.org/
65. DOM http://www.w3.org/DOM/
66. Sun: Single/Multiple pipelines http://java.sun.com/products/jsp/html/JSPXML.html
67. Cocoon http://xml.apache.org/cocoon/
68. Axkit http://axkit.org/
69. Apache web-server http://www.apache.org
70. Perl http://www.perl.com/
71. DICE http://dice.ccpp.info/
72. DoCOMo I-Mode http://www.nttdocomo.com/

Abbreviations

3GPP	3rd Generation Partnership Project
ADL	Advanced Distributed Learning
AICC	Aviation Industry CBT (Computer-Based Training) Committee
API	Application Program Interface
ASP	Active Server Pages
CBT	Computer-Based Training
CC/PP	Composite Capabilities / Preferences Protocol
CGI	Common Gateway Interface
CHTML	Compact Hypertext Markup Language
COM	Component Object Model
CSS	Cascade Style Sheet
CTLX	Custom Tag Library Extensions
DOM	Document Object Model
DTD	Data Type Definition
GPRS	General Packet Radio System
GSM	Global System for Mobile communication
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineers
I-MODE	Internet-Mode
IMS	Instructional Management System
IP	Internet Protocol
ISP	Internet Service Provider
J2ME	Java 2 Micro Edition
JiT	Just in Time
JSP	Java Server Pages
LoD	Learning on Demand
LTSC	Learning Technology Standards Committee
MIME	Multipurpose Internet Mail Extensions
MMS	Multimedia Messaging System
OS	Operating System
PC	Personal Computer
PDA	Personal Data Assistant
PDF	Portable Document Format
PHP	PHP Hypertext Pre-processor
QoS	Quality of Service
RAM	Random Access Memory
RDF	Resource Description Framework
ROM	Read Only Memory
SAX	Simple API for XML
SCORM	Shareable Courseware Object Reference Model
SMIL	Synchronized Multimedia Integration Language
SMS	Short Message Service
SyncML	Synchronized Markup Language
TCP	Transport Control Protocol
UAProf	User Agent Profile

UMTS	Universal Mobile Telecommunications System
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VHE	Virtual Home Environment
VNC	Virtual Network Computing
WAP	Wireless Application Protocol
WML	Wireless Markup Language
WYSIWYG	What You See Is What You Get
XHTML	eXtensible Hyper Text Markup Language
XML	eXtensible Markup Language
XSL	eXtensible Stylesheet Language
XSLT	eXtensible Style Sheet Language Transformer
XSP	eXtensible Server Pages

A Appendix

A.1.1 XSLT authoring tools

Source PerfectXML.com

Name	Environment
eXcelon Stylus Studio An integrated development environment for creating, validating, and debugging XSL stylesheets, and XML-to-XML mappings.	Microsoft Windows
IBM XSL Editor The IBM XSL Editor application allows a user to import, create, and save Extensible Stylesheet Language (XSL) style sheets and Extensible Markup Language (XML) source documents.	All Java Platforms
Induslogic XSLWiz Save time by automatically generating XSLT to transform XML documents with our drag-and-drop tool.	Microsoft Windows
iXSLT A 100% W3C compliant XSLT (eXtensible	Microsoft Windows

Stylesheet Language Transformations) processor. Includes XML Style Wizard.

Komodo Komodo is ActiveState's cross-platform, multi-language Integrated Development Environment (IDE).	Microsoft Windows, Linux
MarrowSoft Xselerator	Windows
Visual XSLT A full-featured integrated development environment that includes all the features of a modern IDE.	Microsoft Windows
Whitehill <xsl>Composer Engineered to generate WYSIWYG XSL, replacing the need for manual hand coding.	Microsoft Windows
XML Cooktop® A development environment for authoring, editing, and testing XSLT style sheets, XML documents, DTDs, and XPATHs	Microsoft Windows
XML Origin Editor A powerful Unicode aware XML editing solution with substantial XSLT authoring and debugging support.	Microsoft Windows
XML Spy XML Spy now	Microsoft Windows

includes full XSL
capabilities that
range from editing
through
transformation to
viewing the output
and thus allows for
a complete round-
trip work-style.

xslide Emacs Major Mode for XSL Stylesheets	Emacs
XSpLit Splits HTML files into an XML DTD and XSL stylesheet.	Microsoft Windows

A.1.2 XML authoring tools

Name	Environment
<oXygen/> Java-based XML editor with support for XML, XSL, TXT and DTD documents.	Java
EditML Pro v2.0 A windows based editor for creating well-formed XML documents.	Microsoft Windows
ElfData XML Editor 1.13 for OSX/Classic Extensive XML Building tool that validates via DTDs. XML Editor is a clean, fast, flexible and powerful XML Editor. Many users claim that it is easier to use and more thorough than the PC XML editors.	OSX/Classic Mac
Emilé The Extensible Markup Editor for Mac	Macintosh

eWebEditPro+XML WYSIWYG, XHTML/HTML editor from Ektron	Microsoft Windows
Komodo Komodo is ActiveState's cross-platform, multi-language Integrated Development Environment (IDE).	Windows, Linux
Microsoft XML Notepad A simple prototyping application for HTML authors and developers that enables the rapid building and editing of small sets of XML-based data.	Microsoft Windows
Peter's XML editor Peter's XML Editor, version 1.1 is now available for FREE download!	
Stilo WebWriter An industrial strength XML Editing tool, providing all of the functionality expected of leading XML Editors.	Microsoft Windows
UltraXML™ 2.0 High-end WYSIWYG XML publishing system with real time ActiveXSL, Visual DTD editing.	Microsoft Windows
Visual XML A tool that enables you to create and modify DTD and XML documents.	Java
Visual XML Writer A Software Product from eInternet Studios	Microsoft Windows
WorX Product Suite XML Authoring Made Easy	Microsoft Windows

Xeena A visual XML editor, is a generic Java application from the IBM Haifa Research Laboratory for editing valid XML documents derived from any valid DTD.	Windows 95, Windows NT, Windows 98, Unix, MacOS
XMetaL XMetaL's word-processor like interface makes it easy for anyone to create XML content, or convert documents from other formats, including Microsoft Word and Microsoft Excel.	Microsoft Windows
XML Cooktop® A development environment for authoring, editing, and testing XSLT style sheets, XML documents, DTDs, and XPATHs	Microsoft Windows
XML Instance V2.1 XML Instance™ is a breakthrough product for schema-driven data editing that allows for the creation, editing, and management of data-oriented XML documents, messages, and configuration files.	Java (Windows 95/98/NT/2000, Unix, Linux, MacOS)
XML Origin Editor A powerful Unicode aware XML editing solution with substantial XSLT authoring and debugging support.	Microsoft Windows
XML Pro An advanced XML Editor with an intuitive interface that allows XML experts and novices alike to create valid, well-formed XML documents.	Microsoft Windows, Solaris or Linux

XML Spy
The first true Integrated Development Environment for the eXtensible Markup Language that includes all major aspects of XML in one powerful and easy-to-use product.

Microsoft Windows

XMLMate
XML editor with out-of-the-box auto conversion to XML for different file formats and data sources.

Microsoft Windows

XMLwriter
A powerful XML editor for Windows

Microsoft Windows

A.2 Support for UAProf devices:

UP.SDK

ColorCapable: No

TextInputCapable: Yes
ImageCapable: Yes
CcppAccept:application/vnd.wap.wmlc, application/vnd.wap.wbxml,
application/vnd.wap.wmlscriptc, application/vnd.wap.multipart.mixed,
application/vnd.wap.multipart.form-data,text/vnd.wap.wml,
text/vnd.wap.wmlscript, image/vnd.wap.wbmp, image/gif,
CcppAccept-Charset: US-ASCII,ISO-8859-1,UTF-8,ISO-10646-UCS-2,
CcppAccept-Encoding: base64,
OutputCharSet: ISO-8859-1,US-ASCII,UTF-8,ISO-10646-UCS-2,
InputCharSet: ISO-8859-1,
WmlScriptVersion: 1.2.1,
WmlVersion: 2.0,
SecuritySupport: WTLS-1
TablesCapable: Yes
WapDeviceClass: C
WapVersion: 2.0
WmlDeckSize: 3000
WmlScriptLibraries: Lang,Float,String,URL,WMLBrowser,Dialogs,
WtaiLibraries:
WTA.Public.makeCall,WTA.Public.sendDTMF,WTA.Public.addPBEntry,
BluetoothProfile: headset,
ImageCapable: Yes
Keyboard: PhoneKeypad
Model: R999
NumberOfSoftKeys: 0
ScreenSize: 121x87
ScreenSizeChar: 15x6
StandardFontProportional: Yes
SoundOutputCapable: Yes
TextInputCapable: Yes
AcceptDownloadableSoftware: No
FramesCapable: No
BitsPerPixel: 2
Push-Accept: application/wml+xml,text/html,
Push-Accept-Encoding: base64,quoted-printable,
aPush-Accept-AppID:
Push-MsgSize: 1400

Alcatel

Alcatel One Touch 301
Alcatel One Touch 311
Alcatel One Touch 501
Alcatel One Touch 511
Alcatel One Touch 512

Mitsubishi

Trium Eclipse

Motorola

Motorola A008
Motorola P7389
Motorola T191
Motorola T192
Motorola T2288
Motorola T280
Motorola Ti250
Motorola Ti260
Motorola V2288
Motorola V50
Motorola V66

NEC

NEC 21i

Nokia

Nokia 3330
Nokia 3360
Nokia 3410
Nokia 3510
Nokia 3590
Nokia 5210
Nokia 5510
Nokia 6210
Nokia 6250
Nokia 6310
Nokia 6310i
Nokia 6340
Nokia 6510
Nokia 6590
Nokia 7110
Nokia 7110 (4.94 and above)
Nokia 7210
Nokia 7650
Nokia 8310
Nokia 8390
Nokia 8910
Nokia 9110i
Nokia 9210 WML

Siemens

Siemens C35i
Siemens S35
Siemens S45

Sony-Ericsson

Ericsson A2628
Ericsson P800
Ericsson R320
Ericsson R380
Ericsson R520

Ericsson R600
Ericsson T20
Ericsson T20e
Ericsson T39
Ericsson T65
Ericsson T68R1
Sony CMD-Z5

A.3 Support for CC/PP profile devices

Opera

      DELI'S output of CC/PP capabilities:
          ColorCapable: Yes
          TextInputCapable: Yes
          ImageCapable: Yes
          CcppAccept: image/gif, image/jpeg, text/html,
          BrowserName: Opera
          TablesCapable: Yes
          ImageCapable: Yes
          Keyboard: Qwerty
          StandardFontProportional: Yes
          SoundOutputCapable: Yes
          TextInputCapable: Yes
          AcceptDownloadableSoftware: No
          Vendor: Opera
          FramesCapable: Yes

Microsoft Internet Explorer

      DELI's output of CC/PP capabilities:
          ColorCapable: Yes
          TextInputCapable: Yes
          ImageCapable: Yes
          CcppAccept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/vnd.ms-powerpoint, application/vnd.ms-excel, application/msword, text/html,
          CcppAccept-Charset: US-ASCII, ISO-8859-1, UTF-8, ISO-10646-UCS-2,
          CcppAccept-Encoding: base64,
          BrowserName: Microsoft
          TablesCapable: Yes
          ImageCapable: Yes
          Keyboard: Qwerty
          StandardFontProportional: Yes
          SoundOutputCapable: Yes
          TextInputCapable: Yes
          AcceptDownloadableSoftware: No
          Vendor: Microsoft
          FramesCapable: Yes

Mozilla (Netscape included)

      DELI's output of CC/PP capabilities
          ColorCapable: Yes
          TextInputCapable: Yes
          ImageCapable: Yes
          CcppAccept: image/gif, image/jpeg, text/html,
          BrowserName: Mozilla
          TablesCapable: Yes
          ImageCapable: Yes
          Keyboard: Qwerty

StandardFontProportional: Yes
SoundOutputCapable: Yes
TextInputCapable: Yes
AcceptDownloadableSoftware: No
Vendor: Netscape
FramesCapable: Yes

Lynx (text-based)

Amalya (W3C test browser)
DELI's output of CC/PP capabilities
ColorCapable: Yes
TextInputCapable: Yes
ImageCapable: Yes
CcppAccept: image/gif, image/jpeg, text/html,
BrowserName: Amaya
TablesCapable: Yes
ImageCapable: Yes
Keyboard: Qwerty
StandardFontProportional: Yes
SoundOutputCapable: Yes
TextInputCapable: Yes
AcceptDownloadableSoftware: No
Vendor: W3CFramesCapable: Yes