ESPRIT Project No. 25 338

Work package J

Pilot application 2

Requirements

ID:	DJ2 V-2.0	Date:	01.05.98
Author(s):	L. Amsaleg, M. Banâtre, M. Billot, P. Couderc, V. Issarny, A-M. Kermarrec, M. Le Nouy, J-P. Routeau	Status:	
Reviewer(s):	D. Donaldson (APM)	Distribution:	

Change History

Document Code	Change Description	Author	Date
DJ2 V-1.0	First version of document.	INRIA & TCM	30.Mar.98
DJ2 V-2.0	2nd version. Includes feedback from partners.	INRIA & TCM	01.May.98

1 Introduction

Newspapers across the globe are turning their sights on producing electronic versions of their daily and weekly editions. With home penetration of PC's forecast to expand by as much as 23% worldwide in the next four years, with more than 200 million people accessing the Internet by the end of the decade, newspapers cannot afford to ignore these alternative paths to information delivery. Newspapers are looking to the Internet as a means of attracting new readers, securing customer satisfaction among current clients and increasing revenue with new services. Publishers also hope that their entrance into Internet may help papers struggling with an aging and declining readership. The Newspaper Association of America (NAA) predicts that on-line editions, audio text and video text will supplement, not replace, the printed newspaper. As of April 23, 1998, the Editor & Publisher Interactive Online Newspaper Database has 2,859 online newspaper entries (see http://www.mediainfo.com). There are about 499 online newspapers in Europe, 1,749 in the US and about 213 in Canada.

Designing and implementing an electronic newspaper service is a difficult problem. On the one hand, a successful on-line newspaper has to provide valuable additional services that can not be achieved easily with the traditional paper-based newspapers. Such services can be a fast and easy access to the archives, high quality for pictures (that can be in color, as opposed to the typical black-and-white pictures of papers), support for multimedia data, instant updates and news flashes, gateways to other services like ticket booking, etc. On the other hand, providing these additional services impacts the architecture of the electronic newspaper application and raises new problems in the design of fundamental low-level features. For example, all layers of an on-line paper application have to be designed such that they enforce strong Quality of Service requirements (responsiveness, availability, scalability), which includes caching, information retrieval and multimedia access problems.

As demonstrated by this paper, the FollowMe infrastructure offers good support for building a complex, large-scale, on-line electronic newspaper application. This document is a requirement analysis that describes the overall architecture of the second pilot application (namely ETEL++) that is constructed on top of the FollowMe framework. Section 2 presents an overview of this pilot application. Section 3 details ETEL++ and presents an object model that describes its main components. Section 4 presents several use cases that describe the flows of data within ETEL++ along with concrete examples showing interactions between users and the pilot application.

2 Overview of the Pilot Application 2: ETEL++

The purpose of this section is to provide an overview of the second pilot application. We first present a short summary of some related work in the area of electronic press publishing. We then present ETEL, which is the electronic newspaper developed by TCM, INRIA and O2. The second pilot application, ETEL++, is derived from ETEL. ETEL is therefore the background of the pilot application 2. We then present the fundamental differences between ETEL and ETEL++, and motivate the need for the FollowMe infrastructure. We conclude this section by a summary of the technical achievements of the pilot application.

2.1 A Short Survey on Electronic Press Publishing

The traditional, paper-based, version of a newspaper is somehow similar to its electronic counterpart in the sense that both offer to a reader selected information presented and organized in a specific way. There are, however, many difference between the electronic and paper versions, and highlighting some of them is the purpose of this section.

A first difference is the opportunity for easy navigation between articles. This well-known feature corresponds to the notion of navigation via hyperlinks in hypertext documents. Another difference is the layout of electronic articles, which usually differs from the one used for the paper edition. First, the size of the screen is likely to be much smaller than the size of a typical sheet of paper. In addition, paper and electronic versions have different physical orientations: papers are usually in a portrait mode; screen are landscape-based, and the pixel definition of screens is poor with respect to the quality of today's professional printers.

In general, the layout of an electronic newspaper is based on the notion of blocks, each block being composed of other nested blocks, titles, headers, pictures, sections, paragraphs, etc. Two basic approaches for the layouts are commonly used today, and they differ by the way blocks are generated. The first approach creates blocks that simply depend on the actual size of the electronic page as well as the volume of information kept within each block. The resulting layout divides the page horizontally (see http://crayon.net/using/sample, http://www.telegraph.co.uk, http://www.welt.de). The second approach creates blocks depending on their relevance and also

on the relationship they have one another. Two blocks that contain tightly related articles are likely to form a new block by itself, while two articles that share only little are likely to be apart. The layout of the blocks within the electronic page reflects somehow this notion of relevance: related blocks are juxtaposed (see http://www.washingtonpost.com, http://www.liberation.fr and ETEL as described below).

It is fairly complex to create a layout that nicely displays the content of a newspaper. The service provider has to support multiple end-devices with different resolutions, sizes, graphical capabilities, etc. Furthermore, the layout *for each day* depends on the actual content of the information provided. The automatic computation of a consistent, meaningful, intuitive, nice and easy-to-use layout for the electronic version of a daily newspaper is still an issue – today, specific teams have to recompose electronic pages using their paper-based counterpart.

Another major strength of electronic newspapers is the possibility to support the creation of personalized newspapers by allowing customers to determine which sections of the news they want to receive on a daily or even hourly basis. In addition, electronic newspapers typically supplement their paper-based counterpart in that they offer multimedia articles, access to archives, search by content, push of data such as news flashes, etc.

There are today basically three different types of electronic newspapers: (i) electronic newspapers built from their paper counterpart as a base, (ii) electronic newspapers built from a collage of various articles available online and (iii) electronic newspapers built from scratch. While the first category encompasses many major news providers (see ETEL as described below, http://www.washingtonpost.com, http://www.lemonde.fr, http://www.nytimes.com, http://www.liberation.fr, http://www.welt.de), the second and third categories deserve special interest.

An excellent example of the second type of electronic newspapers is CRAYON (see http://crayon.net). CRAYON repackages existing news offered on the Web for free, that is, CRAYON is pasting-up news found all over the net. It allows a user to customize his interests and save his choices. With CRAYON, the user has to choose which areas the personalized newspaper will cover. Typical areas are US News, Regional and Local News, World News, ... To each area corresponds a certain number of information sources, each accessible by a hyperlink embedded within the CRAYON page. The potential problem with this service is that it relies upon the availability of all the news sources on the Web for free. Another major problem is that no layout of data is enforced. Repackaged news are simply concatenated into a list itself divided into thematic sub-lists.

Examples of the last category (online papers created from scratch) are the Nando Times in the United States and “Le petit bouquet” (see http://www.le-petit-bouquet.com) in France. The Nando Times and “Le petit bouquet” have no existing printed product.

More information on other related work can be found in the FollowMe deliverable DJ1 – Survey on Electronic Newspapers.

2.2 ETEL as a Background for the Pilot Application 2

Ouest-France is the largest regional daily newspaper in France. Every single day, 40 different editions are printed (one edition is around 35 pages), 1,500 pictures, 2,500 articles (around 125,000 lines). More than 800,000 copies are sold every day. All the different editions share a common set of pages (e.g., national and world news, business, etc.). One edition, however, differs from the other because each contains a set of specific pages giving very detailed local news.

The creation of an online version of Ouest-France was initiated in October 1995. It is called ETEL. INRIA, Ouest-France, TCM and O2 develop ETEL. In a nutshell, INRIA is responsible for implementing the bulk of the service (personalization, data delivery and enforcing Quality of Service), Ouest-France is the data provider, TCM is responsible for solving the complex issues of automatic layout computation and user interfaces and O2 is responsible for providing the low-level object-oriented database storing all the articles and pictures. The ultimate goal is to offer to subscribers, around 1999, an online access to Ouest-France, with value-added features like personalization, multimedia, gateways to related services, access to archives, digest, etc.

Since subscribers will be charged for using ETEL, enforcing the quality of the provided service is a major concern, and has an impact on all the layers of the design. In this context, the quality of service has two meanings: (i) quality of the provided information and (ii) quality of access, that is, availability and fast response times [BIL96, BILC97]. For the quality of access, the design includes a profile-based predictive prefetching policy that tries to anticipate the user's requests based on his profile.

For the quality of content, the design of ETEL includes the following features:

- Coupled production of paper and electronic editions, i.e., production of the two versions from the same data. ETEL is one of the first electronic newspaper that is able to automatically build an electronic edition from the edition's logical data that are also used for the construction of the paper version.
- Customization of the electronic edition according to the user's particular interest, also known as the user's profile. Profile allows a user to compose its own personal edition with respect to his centers of interests. Graphical tools help the user to specify which thematic and geographical areas the personalized newspaper will cover.
- Presentation of the information that combines the advantages of both the paper version (e.g., layout) and the electronic support (e.g., interactivity), including the integrated view of newspaper-based information and links to various services. An extensive work has been done so far to preserve as much as possible the actual look of the paper-based newspaper once displayed on the screen.

A prototype of ETEL is now fully functional. It is possible to automatically generate the electronic version using the raw data provided by the journalists. ETEL supports multi-dimensional customization in profiles. It works in a client-server-based environment, that is, the main Ouest-France server manages a centralized database (there are several servers, however) and serves the requests issued by remote clients (the users). Users access ETEL through a

dedicated interface solely available for PCs running Windows'95 having Acrobat Reader (the delivered newspaper is in PDF format). The telecommunication infrastructure relies on an ISDN-based network. The current work focuses on subscriptions, collection of fees, access to archives and the integration of value-added services (travelling, advertisements, news digests, gateways to the Web, full support for multiple media, ...). An additional focus is the deployment of a full-scale system in order to support a large number of users and a significant load for ETEL's servers.

2.3 From ETEL to ETEL++

We present in this section ETEL++. ETEL++ can be viewed as a possible solution to some of the limitations that are inherent to ETEL. We therefore detail these limitations, and then how ETEL++ gets over. We finally demonstrate that the infrastructure provided by the FollowMe project is adequate for building ETEL++.

2.3.1 Fundamental Limitations of ETEL

ETEL is a very complex, professional, full-scale electronic press application. Its design and implementation have to deal with real-world constraints. Also extremely powerful, ETEL has to face some fundamental limitations, which are listed below. It is possible to get over these limitations. Doing so would require, however, a significant revision of ETEL's architecture. As illustrated in this section, this revision can be greatly improved (i.e., made simpler and more efficient) if the revised design rely on an infrastructure similar to the one provided by the FollowMe project. Before presenting why and how FollowMe could be used to enrich ETEL, we briefly describe the limitations mentioned above. These limitations are:

- *Centralized architecture, traditional client-server model, scalability.* While a centralized architecture makes the design and implementation simpler, it fails to cope with today‘s requirements for wide-area, largely distributed applications. In addition, it is well known that centralized applications do not scale well when the load dramatically increases. Coping with thousands of users calls for a better distribution of ETEL. Furthermore, this type of architecture is not balanced, that is, the potential processing power of today's powerful client workstations is ignored. Therefore, the maximum load its central data repository and the associated communication links can handle limits the overall performance of ETEL.

- *Request-Reply model of communication.* This model falls from the client-server-based architecture. It fails to cope with today's requirements for supporting both *pull* and *push* technologies. Requests-replies are pull, and its advantages are well known. Push technologies, however, provide opportunities for delivering data in a radically different way. Push-based data delivery does not require any (possibly automated) periodic pooling of the data providers in order to receive updates, as pull-based systems do. Pooling typically places a heavy burden on the overall system since it loads both the clients that pool and the servers that have to answer immediately. In addition, setting an appropriate frequency for pooling is difficult. Pooling too slowly decreases the accuracy

of data; pooling to often overloads the systems and is a waste of resources when nothing new has to be returned.

- *On-line access*. On-line access appears to be the most straightforward way for interacting with an electronic newspaper. Not surprisingly, ETEL offers this type of access to its users. The emergence of nomadic computing, however, invalidates this basic assumption. The increasing number of mobile users calls for means for accessing information in a disconnected way. Therefore, on-line, instant access is not anymore the dominant paradigm. It is necessary to provide alternative ways of accessing information.

- *Single output format*. ETEL targets PC's users having Windows'95, Acrobat Reader and an ISDN connection. It is however unlikely that all users of a very large user set will all use the hardware/software configuration targeted. Rather, users are likely to use many different types of terminals (powerful desktops, cheap PCs, portable computers, laptops, palmtops, etc.) each having specific characteristics (energy autonomy, multimedia capabilities, network connection type). Therefore, supporting only a single type of output (currently PDF) is a severe limitation for ETEL.

2.3.2 Getting Over these Limitations: ETEL++

The purpose of ETEL++ is to investigate new ways to get over some of the limitations that have been highlighted above. ETEL++ can be viewed as a new electronic newspaper application that sits along with ETEL. It is important to note that ETEL++ *is not* a new ETEL, that is, it must not be seen as a new release of ETEL providing additional functionalities. Rather, ETEL++ is designed to be an *avant-garde* prototype in which the most recent advances in research and the most up-to-date industrial technologies can be mixed together. It is likely that specific solutions developed for ETEL++ could be reused in other environment. From an abstract point of view, ETEL++ can be seen as a large-scale, widely distributed application managing huge amounts of data that have to be delivered to a very large set of mobile users.

ETEL++ is an attempt to get over the above limitations. It is important to remark that the features subsequently presented raise a lot of research issues; building ETEL++ requires solving these issues. ETEL++ gets over ETEL's limitations by providing:

- *Large-scale distribution, better scalability*. While ETEL is mostly centralized, ETEL++ will be mostly distributed. One major aspect of ETEL++'s architecture is that it will not contain any centralized point of control that can potentially be a severe bottleneck. It is necessary to state, however, that the core of the data provided by ETEL++ comes from Ouest-France, and therefore, this could be perceived as a bottleneck. ETEL++ will disseminate the data as soon as it is extracted from the Ouest-France central database. The destination of this disseminated data is not the final users, but intermediate sites that are ultimately accessed by users at newspaper browsing time. It is important to remark that this model in which data is spread over many places is fundamentally different from ETEL's model in which data remains at a central place until the newspaper browsing time.

  A fully distributed model, as the one chosen for ETEL++, is more suitable for coping with large increases in the overall size of the system, e.g., in the number of concurrent

users or in the global volume of data that is exchanged. Controlling the behavior of a fully distributed application, however, raises many difficult issues that have to be addressed. Having mobile code and objects is a possible approach for addressing these issues. Mobility in this case may help in keeping the implementation simple. Another advantage of having mobile code is that it becomes possible to use, if necessary, the processing power of user terminals.

- *Internet access*. ETEL relies on specific connections (ISDN). In contrast, ETEL++ will base its communication infrastructure on Internet. One salient aspect of this general-purpose communication network is the large variance in the response time when transferring data between any two machines. In this context, it is therefore hard to enforce strong Quality of Service requirements as it is for ETEL. Enforcing responsiveness, accessibility and availability in the context of Internet is a central research issue for large-scale applications.

- *Multi-terminal support*. In contrast to ETEL, ETEL++ will support multiple types of user terminals for accessing the electronic newspaper. Doing so raises many hard problems such as the conversions between one format to another (e.g., converting an ASCII-based article body to an equivalent article that could be delivered over a telephone). Allowing access to the service via a large set of terminals, each differing in their physical characteristics, increases the visibility of Ouest-France.

- *Agent-based interactions*. Agents are useful in breaking down the traditional client-server model on which ETEL is based. In addition, agents allow the discovery of services that may interest readers.

- *Pull- and Push-based data delivery*. In contrast to ETEL, ETEL++ will support these two modes of data delivery. This enables implementing several features that are based on push technology, and some have been presented in the list of limitations. In addition, push and pull can be used for helping in the deployment of ETEL++ itself, and not only for delivering data to end users. For example, the dissemination of data to intermediate sites can be achieved using either push or pull, or their combination. This flexibility offers opportunities for fine performance tuning.

- *On-line and Off-line access*. Coping with mobile users, unpredictable disconnections, and off-line paper browsing is one of the goals of ETEL++. Supporting these two modes increases the complexity of the interface that is offered to users, and raises interesting issues and tradeoffs.

- *Context-sensitive data*. ETEL offers tools for building personalized editions, that is, any given user can pick and choose which articles he wants to read and which he does not. Therefore, two users may have different customized editions if they have different centers of interests. The content of an edition for a given user, however, remains fixed unless the user changes its preferences. It is therefore impossible for a user to receive context-sensitive data, that is, data which varies depending on a particular context such as the physical location of the user. With context-sensitive data, a user can have access, via the newspaper, to the current list of movies that are played in the town from which he connects ETEL++. Obviously, this list differs if the user is in Paris or London. Accessing

weather forecasts is another straightforward application of context-sensitive data: instead of always including the weather of a fixed place, the electronic newspaper could display the weather that corresponds to the actual location of the user. Context-sensitive data can be used in several ways, and other ways can be quickly devised (sensitive to performance for example). This short paragraph only presented a context based on users geographical locations.

Finally, it is important to note that ETEL++, while providing additional features, will only support a subset of ETEL. ETEL is a very complex, full-scale, professional application. It includes for example complex information about the physical location of articles, as well as modules that are able to automatically produce a nice and easy-to-use layout. (It includes other features that are not listed here, however.)

ETEL++ will support simplified versions of the features that are not in the critical path of demonstrating how ETEL's limits can be broken. For example, the automatic generation of the electronic edition will not be part of ETEL++. In contrast to ETEL, ETEL++ will not include the complex algorithms aimed at preserving as much as possible the look of the paper-based newspaper once on the screen. A simplified version of profiling will be supported in ETEL++ (less dimensions than for ETEL). In contrast to ETEL, ETEL++ assumes that there can be only one picture per article, that the number of headings is bounded, that the length of any article always stays below a limit. Furthermore, ETEL maintains a large number of indexes referring to the content of the articles for the purpose of information retrieval. ETEL++ adopts a similar approach, also indexes are kept simple.

2.3.3 FollowMe as an Adequate Infrastructure

The infrastructure that is provided by the FollowMe project is adequate for building ETEL++. This infrastructure provides the basic building blocks that compose the low levels of the pilot application 2. By assembling these building blocks in an appropriate way, it is possible to keep the implementation of ETEL++ simple and highly modular.

All the modules of FollowMe are developed in Java. It is therefore straightforward to deploy ETEL++ on platforms with different hardware and/or operating systems (SunOS, Solaris, Windows'95). The choice of Java was crucial and helps in keeping the application simple.

At the lowest level, the infrastructure of FollowMe offers transparent mobility for the code and for objects. This therefore removes all the complexity of keeping track of object references, and allows programmers to focus solely on their application. In addition, transparent –although controlled— mobility helps in designing highly distributed applications, making easier data sharing across sites or data movement for bringing data close to their consumers. It also simplifies the deployment of a large-scale application by allowing code movements as well. Mobility will enable to deliver information depending on the location of the user. Also, the mobility of the information may help in enforcing the quality of access in the sense that moving the information between geographically distributed sites in order to follow the users (for example) becomes simple.

Mobile agent technology is also of a great help. It increases the autonomy of users and enhances information discovery and delivery. Agents are useful to gather and also to disseminate data.

ETEL++ needs agents, mainly for coping with context-sensitive data. Reusing the weather forecast example given above, mobile agent technology can be directly used to search for a service providing weather forecast information, this information being subsequently integrated in the newspaper that is delivered at a specific location. In addition, agents increase the level of autonomy of the various components of the system, and may help when the scale of the system will grow.

ETEL++ is also concerned with the multi-terminal support provided by the underlying FollowMe platform. Allowing access to the service via a large set of terminals, each differing in their physical characteristics increases the visibility of Ouest-France. In general, relying on a layer providing such support reduces the complexity of the application and allows coping easily with new devices in an extensible way. Otherwise, major revisions of ETEL++ would be required for extending the set of supported terminals. We would therefore loose in genericity.

2.4 Features of FollowMe Emphasized by ETEL++

In the context of FollowMe, ETEL++ will first demonstrate the need for mobile data and code. Without mobility, the design of ETEL++ would have been more complex and it would have been far less powerful. Mobility impacts all the layers of ETEL++. For example, enforcing the Quality of Service (preserving small response times) whatever are the user's locations requires ETEL++ to deploy itself across the sites that participate to the FollowMe Project. This translates into moving newspapers (or parts of) close to users (newspapers "follow" users) and moving computations to under-utilized (or to specific) machines for the purpose of global load-balancing. Transparent mobility, in this case, is of great help. It should be apparent that a major strength of ETEL++ is that it uses mobility and load-balancing *together*.

Another goal is to demonstrate the need for multiple-terminal support. ETEL++ will focus on delivering the paper to at least two types of terminals: a powerful workstation running a Web browser, and a fax machine. Other types of terminals can be taken into account, time permitting.

In terms of FollowMe workpackages, the above description translates into demonstrating the advantages of building ETEL++ using what is provided by the workpackages B (*Mobile Object Workbench*), G (*Service Deployment*) and H (*User Access*).

While the above description is the primary focus of ETEL++ demonstration, the need for other features such as *Autonomous Agents* and *Service Interaction* will also be demonstrated, on a smaller scale, however.

Aside these demonstrative aspects of ETEL++, it is obvious that the entire infrastructure will be indirectly demonstrated. For example, ETEL++ needs features such as the Information Space and the Personal Profiles. The *Information Space* workpackage provides features that are used to store persistent versions of the newspapers, parts of newspapers, even raw data, or other control related information. *Personal Profiles* provides the means for knowing about the user, and in particular it provides a view on his agenda (also called diary), which is useful for determining where a user might be connected and therefore infer where the data has to be shipped.

2.5 Summary of the technical achievements for the Pilot Application 2

- A survey document (referred to as DJ1) was produced. It presents a short overview of the current work in electronic press services. It details what is specific to electronic press applications compared to traditional paper-based newspapers. It briefly presents the electronic newspaper developed by Ouest-France (ETEL), then moves to ETEL++ before concluding.
- The design of ETEL++ is well advanced. Three design sub-tasks deserve mention, however:
  1. Specific algorithms are designed for extracting the data that is stored in ETEL's database. That data is later transformed into a format that is more adequate for ETEL++. More details on data extraction are provided in this document, Section 4.1.1. This work required close cooperation between TCM and INRIA.
  2. A general structure for each ETEL++ document has been designed. This structure helps in fulfilling the requirements of *User Access* workpackage for supporting multiple terminals. This work required close cooperation between FAST and INRIA.
  3. INRIA is responsible for ETEL++ and the *Service Deployment* workpackage. Therefore, the design of a load-balancing policy tailored to meet ETEL++'s requirements has been initiated. More information on this topic can be found in this document, Section 4.2, and in other documents that have been produced for the Service Deployment workpackage (there are four such documents).
- The core of ETEL++ is designed. As of April 23, 1998, the document DJ3 is in preparation and will be available shortly. This document presents the main processing phases that compute personalized newspapers from the raw data produced by journalists at Ouest-France.
- The requirements for Agents and Multi-Terminal Access strongly relies on the software that is provided by other FollowMe partners. Agents and the discovery of services require *Autonomous Agents* and *Service Interaction* provided by UWE. Multi-terminal support requires *User Access* provided by FAST.
- Overall, the progresses of the Pilot Application 2 are in accordance with the schedule.

3 Detailed Description of ETEL++

Figure 1 provides a high-level view of ETEL++ and its relationship with its major input sources that are needed to produce a personalized edition for a user.

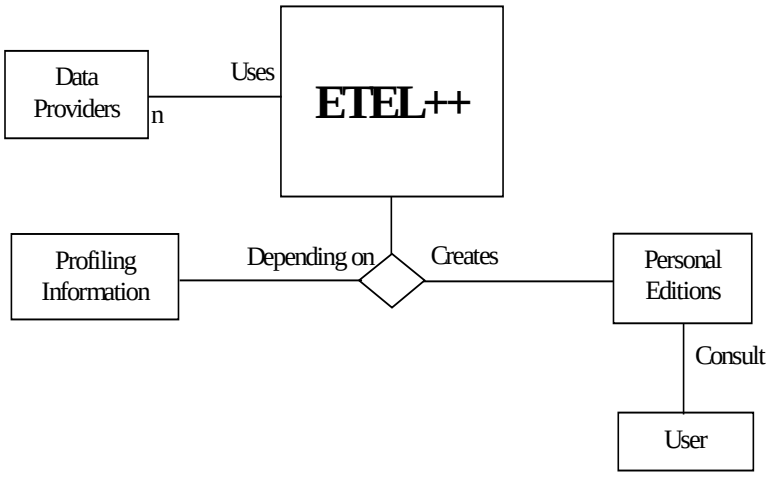

Figure 1: High-Level View of ETEL++.

To produce personalized editions, ETEL++ needs to consume data issued by 2 different sources of information, namely “Data Providers” and “Profiling Information”. Not surprisingly, data providers provide the actual content of the personalized newspapers. It includes for example the body of articles and the pictures that will be displayed along articles as illustrations. Ouest-France can be seen as the major data provider for ETEL++. Other data providers may be used. For example, ETEL++ may need the data provided by a German weather forecast service for satisfying the requests of a user connected in Munich. Data providers provide all the data (i.e., articles) that can potentially be in the users’ editions. A single edition, however, is likely to contain only a subset of these articles.

The other source of data, referred to as “Profiling Information” in the figure, is used by ETEL++ to determine the subset of possible articles that will ultimately compose the personal edition of

users. Profiling information may for example specify what are the preferences of a given user (e.g., Politics, Sport, etc.), or specify what is the type of terminal a user uses to consult his personal edition (a desktop computer, a fax machine, etc.). Depending on that information, the personalized edition may vary. In general, profiling information is used by ETEL++ to *control* the content of a personal edition for a given user. That is, it controls its contents, its structure and its appearance such that the resulting edition matches the user's needs. The purpose of this section is therefore to present the input of ETEL++, the typical mechanisms used to control the contents of an edition for a given user and finally to present its output.

3.1 The Input of ETEL++

As stated above, ETEL++ consumes 2 basic types of data: content-related data and control-related data. A more detail analysis shows that it is possible to subdivide each category. This is detailed in the following sections.

3.1.1 Content-Related Data

The first category of data provides to ETEL++ the informational material that is the starting point for building the electronic versions of the newspaper. From ETEL++'s point of view, content-related data can be seen as a collection of *articles*. An article is the minimum unit of information from which a newspaper can be built. Building a newspaper is essentially putting *some* articles together *in a certain way*. ETEL++ uses two major sources for obtaining articles (as described below): (i) external articles that are provided by data providers and (ii) articles generated by ETEL++ itself. These latter articles are transformations of external articles to cope with multiples media.

The structure of the articles may differ from one data provider to another. Conceptually, however, articles include the following information:

- Provider-related information (provider ID, authorship, …),
- Time-related information (article unique ID, date, time, period of validity, …),
- Title-related information (actual title, header, author, …),
- Descriptive information (list of keywords describing the content of the article, …),
- Body-related information (summary, text of the article, …),
- Associated information (embedded pictures, links to other pictures, ID of related articles, related keywords, …),
- Physical information (media, size, font, colormap, number of columns, layout, weight, regions to emphasize, …).

The above list far from being exhaustive and is intended to give a high-level view of the data ETEL++ consumes. As it is mentioned in the next section, defining a uniform framework for article description is part of our ongoing work with UWE.

3.1.1.1 Externally Provided Articles

ETEL++ gets the bulk of the articles from data providers. A data provider can be conceptually seen as a data source returning articles upon request or delivering articles periodically (or aperiodically). Data providers and ETEL++ have two ways to know about each other. First, a data provider can encode its description such that it can subsequently be discovered by ETEL++. This encoding can give (i) an abstract representation of the role of the data provider (i.e., what is the nature of the provided data, e.g., information about movies in Paris), (ii) a set of interfaces that can be used to interact with the provider and (iii) a representation of the output of the provider. In general, ETEL++ can search for, and possibly discover, services that deliver a particular nature of data. For example, ETEL++ can search for a weather forecast service providing the weather around Munich for a travelling user. Instead of ETEL++ searching for a service, it is possible for a service to make ETEL++ aware of its existence. Conceptually, ETEL++ is a service that consumes a lot of data. Therefore, ETEL++ can be discovered by other services.

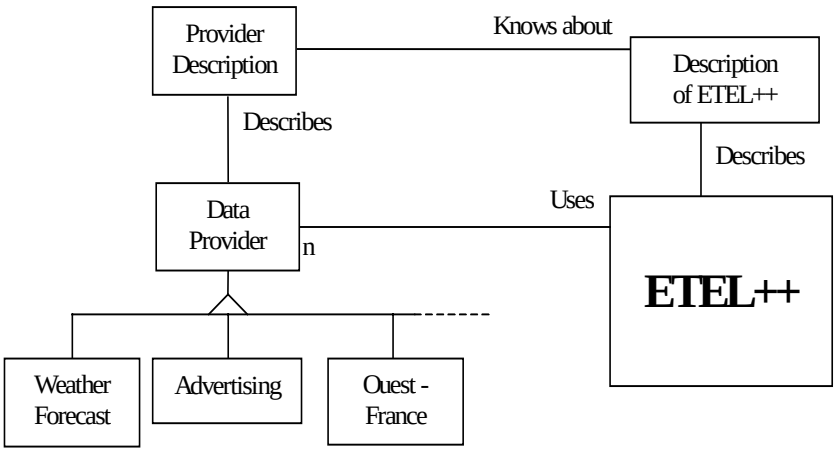

Figure 2: ETEL++ and Data Providers

Figure 2 shows the object model for ETEL++ and data providers. On this figure, both ETEL++ and the data providers are described such that they can interact as described above. These interactions can be based on agents for example. In general, the interactions between ETEL++ and data providers will benefit from two of the workpackages that are part of the FollowMe infrastructure: *Autonomous Agents* and *Service Interaction*. Precisely defining these interactions is part of the ongoing work between UWE and INRIA/TCM. Once ETEL++ has discovered (or has been made aware of) services of interests, data providers are used in order to obtain articles. There can be many different types of data providers, and Figure 2 shows only three of them. The third provider, namely Ouest-France, deserves special mention, however.

Conceptually, Ouest-France is just another data provider. From ETEL++'s point of view, Ouest-France has a well-defined role (it returns articles of various types like politics or sports) and it offers clearly identified interfaces for interaction. The pragmatic view, however, differs slightly. Ouest-France is the major data provider for ETEL++, and the internal of ETEL++ have been designed such that Ouest-France data fits well. In particular, the "intelligence" for interactions

between ETEL++ and Ouest-France as a service does not exist. Nevertheless, the conceptual framework for feeding ETEL++ with data is uniform, and does not give to Ouest-France a special status. This is particularly important for offering a simple description of the mechanisms that are actually used to extract the information from the data providers.

There are two such mechanisms. First, ETEL++ can explicitly request for specific articles by contacting a particular data provider. In general, ETEL++ issues the bulk of its requests overnight such that the newspaper is prepared while no users are likely to be connected (note that the real newspapers are typically composed and printed overnight as well). Second, data providers themselves can send articles to ETEL++ either periodically (like for weather forecast data) or aperiodically (like for news flashes).

In addition to the (trivial) goal of delivering *content* to users, one of ETEL++'s goal is to offer an access to this content from multiple types of terminals. It is however unlikely that any given article can be simply output on any type of terminal. Therefore, to support multiple terminal delivery, ETEL++ has to convert from one media to another the articles it has obtained from the providers. In a way, it is possible to view this conversion process as an internal generation of articles. This generation is detailed in the next section.

3.1.1.2 Internally-Generated Articles

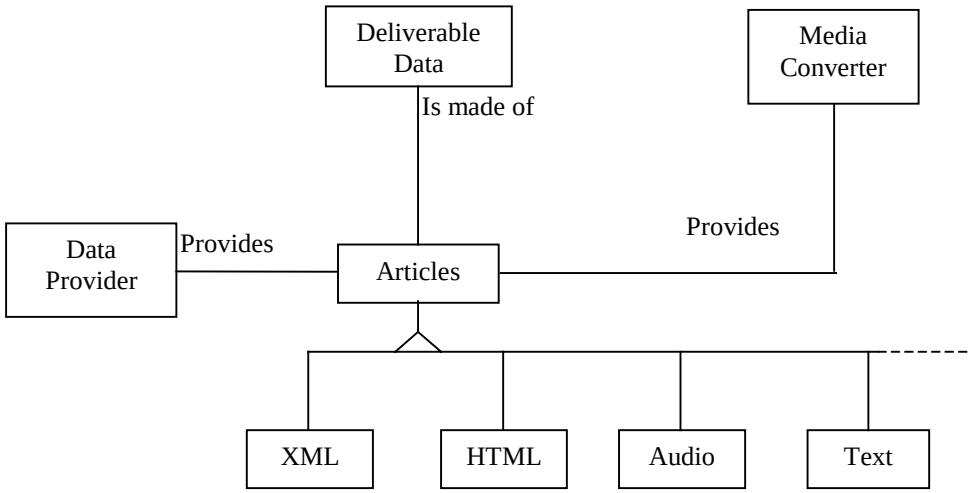

Figure 3: Converting Articles for Coping with Multiple User-Devices

To make possible the delivery of editions to users, ETEL++ has sometime to convert articles from one media to another, in order to fit the specific characteristics of a particular terminal from which users are connected. This means that many different physical representations of the same logical data exist. For example, ETEL++ needs to convert the ASCII-text of an article to an audio stream such that it can be delivered to the telephone of a particular user. The object model for this process is given Figure 3. At the center of this figure are Articles. Articles are encoded using different formats (the Figure presents only 4 such formats). Articles are obtained directly from data providers, and can be encoded using one or more formats. If necessary, that is, if ETEL++ needs to eventually deliver to a user an article in another format then the one(s) in which it has been obtained from the provider, then a conversion process is initiated.

Conceptually, this conversion process consumes an existing article in a given format and produces a set of articles encoded using different representations. For example, this process can convert an ASCII-based article to an HTML-based, a Postscript-based, a PDF-based, and to an audio-based article. Therefore, the set of delivered data, that is, all possible articles that can be delivered to users, is fed with external articles obtained from providers and with derived articles that are in some way equivalent to the external ones, differing, however, by their representation.

Converting an article from one representation to another is a complex process. Basically, it requires to either prune irrelevant information or to enrich the existing one with additional information. The pruning process is for example applied when the converter has to transform an article containing an ASCII-text and some pictures to an audio-based representation. In this case, the conversion process has to prune all the information attached to the pictures since this information is irrelevant to audio devices. In contrast, enrichment is necessary when for example the same original article is converted to HTML. In this case, the converter has to enrich the ASCII-text with tags specifying which part of the article is the title, which part is the main body, and where are the words to eventually emphasize.

The conversion process uses the facilities provided by the *User Access* workpackage. It should be apparent, however, that building a full-scale, general-purpose conversion process is a very complex open-problem tackled by major companies such as Sun Microsystems. Designing such a converter would not be reasonable in the context of the FollowMe project. Rather, the goal is to illustrate the potential for this type of conversion. The conversion process that will be embedded within ETEL++ will be simple: only a small set of devices will be supported, and no general algorithm such as a text-to-speech converter will be needed.

3.1.2 Control-Related Data

The preceding section presented the content that is ultimately delivered to users. This section focuses on the data that is used to control the actual content of a personalized edition. Figure 4 presents the object model for this control.

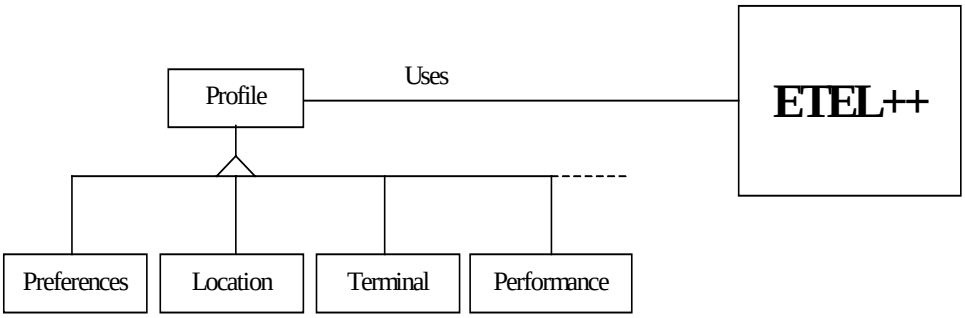

Figure 4: Profiles for Controlling the Content of a Personalized Newspaper

Basically, ETEL++ uses the information stored in Profiles for controlling the content of a personalized edition. A profile is a general data structure that can be broken-down in at least four specific structures. Each structure is detailed in the following subsections.

3.1.2.1 Personal Profiles

Personal profiles contain data that are personal to a user, like his name, his e-mail, his date of birth, his credit card number and so on. A personal profile also provides a view on the user's agenda, that is, it is possible to know for example the time at which the user is used to connect, or the physical location from which it will open the next connection. A more detailed description of Personal Profiles can be found in the documents describing the workpackage E.

Personal profiles are used by ETEL++ to (i) know the identity of users, (ii) know at what time a user is likely to connect and (iii) the next location of the user's connection. Knowing the identity of users is obviously needed since ETEL++ deals with *personalized* editions. Knowing at what time a user is likely to open his next connection is important to prepare his edition early enough such that the user does not have to wait for the newspaper-preparation time (he may wait for other reasons, however, as explained later). Finally, knowing where this connection is likely to take place (in Munich, in Paris?) is needed for ETEL++ to deliver the appropriate context-sensitive data as detailed Section 4.1.3 (see also Section 2.3.2).

3.1.2.2 User Preferences

ETEL++ stores in its internal data structures a profile for every single individual that has subscribed to the electronic version of the newspaper. This profile stores some of the data that is necessary for building the personalized edition for a given user, and in particular, it contains user preferences. Conceptually, this profile contains (this list is not exhaustive):

- Keywords corresponding to subjects that are potentially in "today's" edition. Example of these can be "sports", "politics", or "Internet".
- Keywords corresponding to an arbitrary geographical region of interest. For example, a user might specify that he want to have access to all articles about Rennes and all the villages around.
- Keywords that correspond to existing Ouest-France editions. For example, the keywords of that category can specify that a given user is interested in reading the editions corresponding to "Rennes" and "Brest".
- Keywords corresponding to subjects a reader does not want to see displayed.
- Keywords showing temporal interests. For example, the contents of the electronic version on Sundays might be different from the contents for weekdays.
- Etc.

The profile describing user preferences is built via repetitive phases of dialog between ETEL++ and the user. It is likely that ETEL++ will provide a page (or a set of pages) containing check-boxes that will ultimately reflect his interests. This process is illustrated Section 4.3.2.

In general, a user preference profile acts as a *selection operator* applied over the deliverable data. For a given user, ETEL++ selects the articles that are relevant to this user, and therefore will not include in his personal edition any unwanted article.

3.1.2.3 Terminal Profiles

The edition that is delivered to a user is impacted by the type of terminal that the user will use to "browse" the newspaper. The computations for creating an audio version of the newspaper are obviously not similar to the ones used when preparing a Netscape-based set of articles. It is thus necessary for ETEL++ to know what type of device users will use during their next-to-come connection. Conceptually, ETEL++ needs this knowledge for two reasons. First, ETEL++ needs this knowledge for estimating what will be the load, what are the media conversions that have to be performed and where the major streams of data have to go. This knowledge is subsequently used for deploying the service accordingly (see *Service Deployment* documents). Second, ETEL++ needs this knowledge to select the articles for a user that are compatible with the type of the terminal that is used. As user preferences, a terminal profile acts as a selection operator.

Knowing the type of terminal is possible via the tools provided by workpackage H, *User Access*. It is likely that a user-terminal profile will store a set of terminal descriptions, each description being structured as presented in the section 1.3 of document DH2. In a nutshell, this description may include the following information:

- Terminal type (phone, PC, ...),
- Direction of use (input, output, both),
- If it is Java enabled,
- The XY resolution of the screen, number of colors, lines,
- If it supports sound, stereo, pictures,
- Amount of available memory,
- Mips rate,
- If it has secondary storage,
- Etc.

3.1.2.4 Performance Profiles

Another type of profile that is required by ETEL++ is a profile type for describing the capabilities of the machines that are part of the FollowMe infrastructure. Having profiles describing the performance of the machines is necessary to opportunistically decide upon data and computation placement, such that the global response time for building and latter consulting an edition is minimized. Therefore, ETEL++ needs to have a detailed view on the performance of the infrastructure.

In general, ETEL++ needs to have access to performance monitors that return values that are subsequently used in its decision process for placing data, computations, and also in deciding when the computations and/or the transfer of data have to be initiated. Workpackage G, *Service Deployment*, offers general purpose monitoring tools aimed at helping ETEL++ in its decision process. More information can therefore be obtained from the documents describing this other

workpackage. In a nutshell, however, the performance information ETEL++ needs to place data and computation appropriately is, for a machine ETEL++ may use:

- Global amount of memory usable,
- Mips rate,
- Global amount of free disk space usable,
- Number of connections simultaneously openable,
- Bandwidth between any two machine,
- Physical location of the machine,
- OS type.

3.2 The Output of ETEL++

There are basically two ways to use ETEL++, therefore, it is possible to distinguish two different types of output. These two types are not targeted for being used by the same category of users. The first category of users are humans. These are the typical users of ETEL++. The other type of users are agents. In this case, ETEL++ can be viewed as a data provider itself. Conceptually, the data that is delivered by ETEL++ is analogous for both categories, although the output for humans is far more sophisticated. We detail hereafter the two types of output.

3.2.1 Output for Typical Users

Typical users, i.e., humans, receive from ETEL++ *personalized editions*. The edition of a user is said to be personalized because its contents, its structure and its appearance match the user's specifications. A personalized edition is made of a set of pages, each page being made of a set of articles. Editions, pages and articles can be modeled as described Figures 5, 6 and 7.

An edition (see Figure 5) is made of a set of pages, and of a set of meta-information. This meta-information is used to know to whom a given edition is built for, when it has to be delivered, and also to know some physical properties of an edition such as its overall volume. In general, meta-information is used by ETEL++ for taking appropriate decisions. For example, knowing the size of an edition is useful for estimating its transmission time and possibly using prefetching for enforcing Quality of Service requirements.

A page (see Figure 6) is made of a set of articles, and of a set of meta-information. The goal of that meta-information is, as for editions, to help ETEL++ in its decision taking processes. For example, keywords are meta-information attached to a page. In this case, keywords are used to determine the order in which pages are ultimately delivered, according to user preferences. Physical meta-information is also attached to pages, such as the size of a page, the number of articles it contains, etc.

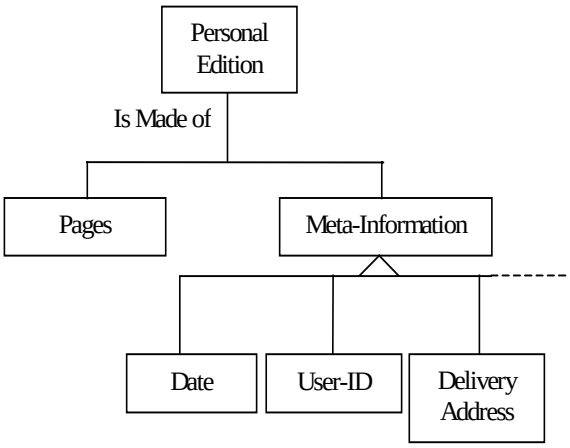

Figure 5: Model for a Personal Edition

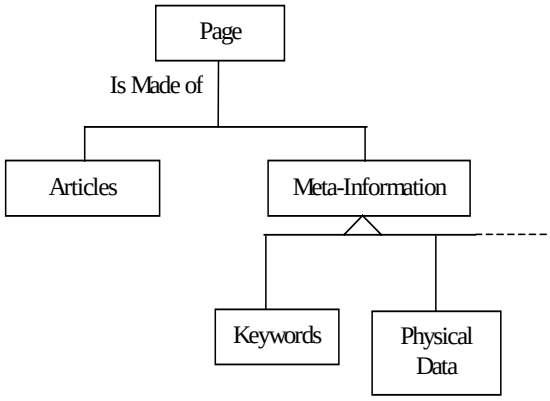

Figure 6: Model for Pages

Finally, Figure 7 describes the structure of articles. As editions and pages, an article is partly made of meta-information. In contrast to pages and editions, an article can not be broken-down into a smaller granule. Therefore, an article is the minimal unit of data that is managed by ETEL++ for the purpose of newspaper building. The information part of articles contains what we referred to as “contents” earlier in this document (see Section 3.1.1). The meta-information part stores what is used by ETEL++ for selecting appropriate articles that will ultimately be included in the delivered edition. Not surprisingly, this meta-information includes the fields that have been presented Section 3.1.1 and also include the required information for applying the user-preference and user-terminal selection filters, as described Section 3.1.2.

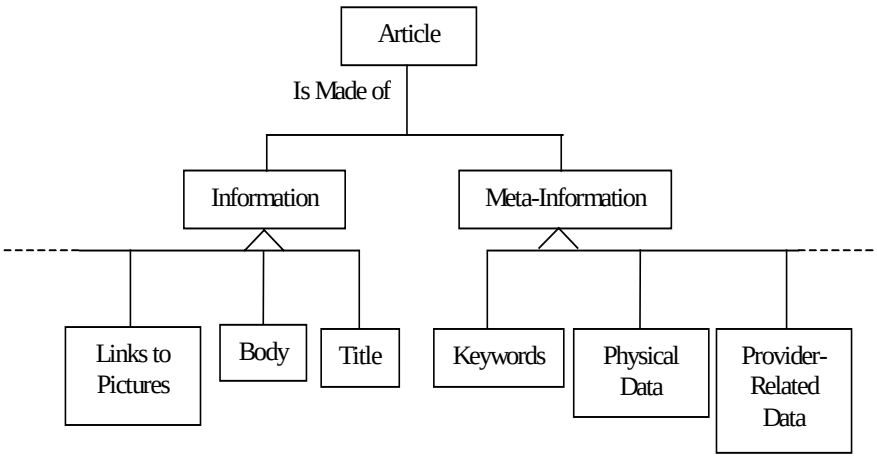

Figure 7: Model for Articles

3.2.2 ETEL++ as another Data Provider

The second type of output is not intended for typical users, but rather for agents. We explained Section 3.1.1.1 that Ouest-France is considered by ETEL++ as a regular data provider. It is therefore possible to consider ETEL++ itself also as another data provider. Therefore, ETEL++ will offer to agents for service interaction (i) a description of its role, (ii) several interfaces for accessing the data it manages and (iii) a description of the format output data conform to.

Turning ETEL++ into a data provider that could be queried by agents is not realistic within the context of the FollowMe project. We do not have a clever enough model (no one has, actually) nor sufficient manpower for achieving such a complex task. We will provide, however, a simple interface that can be queried by specific agents to demonstrate the potential of considering ETEL++ as a regular data provider. For example, given a keyword and a media type, it is possible to imagine ETEL++ returning an article.

The work that is needed to turn ETEL++ into a general-purpose data provider is still in its infancy. Work and cooperation with UWE is required in this case.

4 Use Cases

We present in this section some typical examples of the processing tasks that take place into ETEL++ for building a personalized edition. The purpose of these use cases is (i) to give a better understanding of ETEL++'s internal thanks to the examples and (ii) to give an overview of the flows of data that ETEL++ has to manage. In some cases, it is possible that design decisions emerge from the description – although a complete design document is in preparation. Use cases are divided in 3 major categories: (i) use cases illustrating how a personalized edition is generated, (ii) use cases illustrating how Quality of Service requirements are enforced and (iii) use cases illustrating typical interactions between ETEL++ and a user.

4.1 Generating a Personalized Edition

We illustrate here some typical processing tasks that are triggered for generating a personalized edition. For simplicity, we ignore any other data provider that is not Ouest-France. Therefore, the data that is used to produce the personalized editions comes only from one source. We first present the typical generation of a newspaper, then its delivery and finally how the generation of an edition is impacted by the actual location of the user.

4.1.1 From RAW Data to the Delivered Edition

Computing a personalized edition requires three major phases. The first phase is the extraction of information from the database at Ouest-France which stores raw articles directly written by the journalists.

The database managed by Ouest-France classifies articles according to two main criteria: the main theme of the article, and the geographical coverage. In a nutshell, Ouest-France builds a graph (not a tree) of links that associate articles together, this linking structure being adequate for the *ad-hoc* publishing tools and the heavy-weighted printing process. This database not only contains the information that is ultimately printed on the paper, but contains also a fairly large amount of data used during the paper-printing phase. That data refers to the physical location of articles and pictures on the paper sheets, and is used to precisely calibrate the printed edition and

to determine the constraints that might force a journalist to break-up an article into pieces that will be printed on different pages. That physical information is also used to determine the ultimate size of pictures, the size of the space between lines and between words for the purpose of justification.

ETEL++ does not directly access the Ouest-France database. Rather, it accesses the ETEL database (see Section 2.2 that describes ETEL as the background for ETEL++). The ETEL database is created by TCM. TCM converts the Ouest-France database into the ETEL database. The ETEL database is more appropriate for being the starting point of the electronic-press publishing process because it has a higher degree of structure than the Ouest-France database.

Figure 8 represents the hierarchy of data as it is represented inside the ETEL database managed by TCM. In general, the structure of a particular edition with ETEL can be depicted as a tree of information. The leaves are the actual articles and/or pictures, while the intermediate nodes of the tree structure the information into a hierarchy. A node might contain actual data, or might only refer to a set of information that is descendant of that node. Converting the Ouest-France database into the ETEL database is a very complex task. The data has to be converted into an object-based model, the graph has to be converted into a tree, and the complex location information for positioning the articles on the paper has to be transformed into PDF-related data for screen-based display.

Within the context of FollowMe, TCM plays another crucial role. TCM is responsible for developing together with INRIA the task that extracts data from the ETEL database in order to build the ETEL++ database. There are couple differences between those two databases that deserve attention:

- The data stored by ETEL++ is also a tree of information, as with ETEL. ETEL++ however, needs to store its data in a format that is more adequate for the various processing tasks required within the context of FollowMe. When data is converted from ETEL to ETEL++, it is transformed into an XML-based format that can be subsequently used by other work-packages such as *User Access*.

- ETEL++ contains only a subset of the ETEL database. ETEL is a full-scale, professional database while the purpose of ETEL++ is to illustrate how an electronic newspaper can benefit from mobility and multi-terminal accesses. Therefore, ETEL++ makes a number of simplifications regarding the structure of the actual newspaper. For example, ETEL++ assumes that there can be only one picture per article, that the number of headings is bounded, that the length of any article always stays below a limit, …

- ETEL stores complex, professional information about the physical location of articles. ETEL++ only maintains more simple layout information, although consistent with respect to the original one.

- ETEL maintains a large number of indexes referring to the content of the articles for the purpose of information retrieval. ETEL++ adopts a similar approach, also indexes are kept simple.

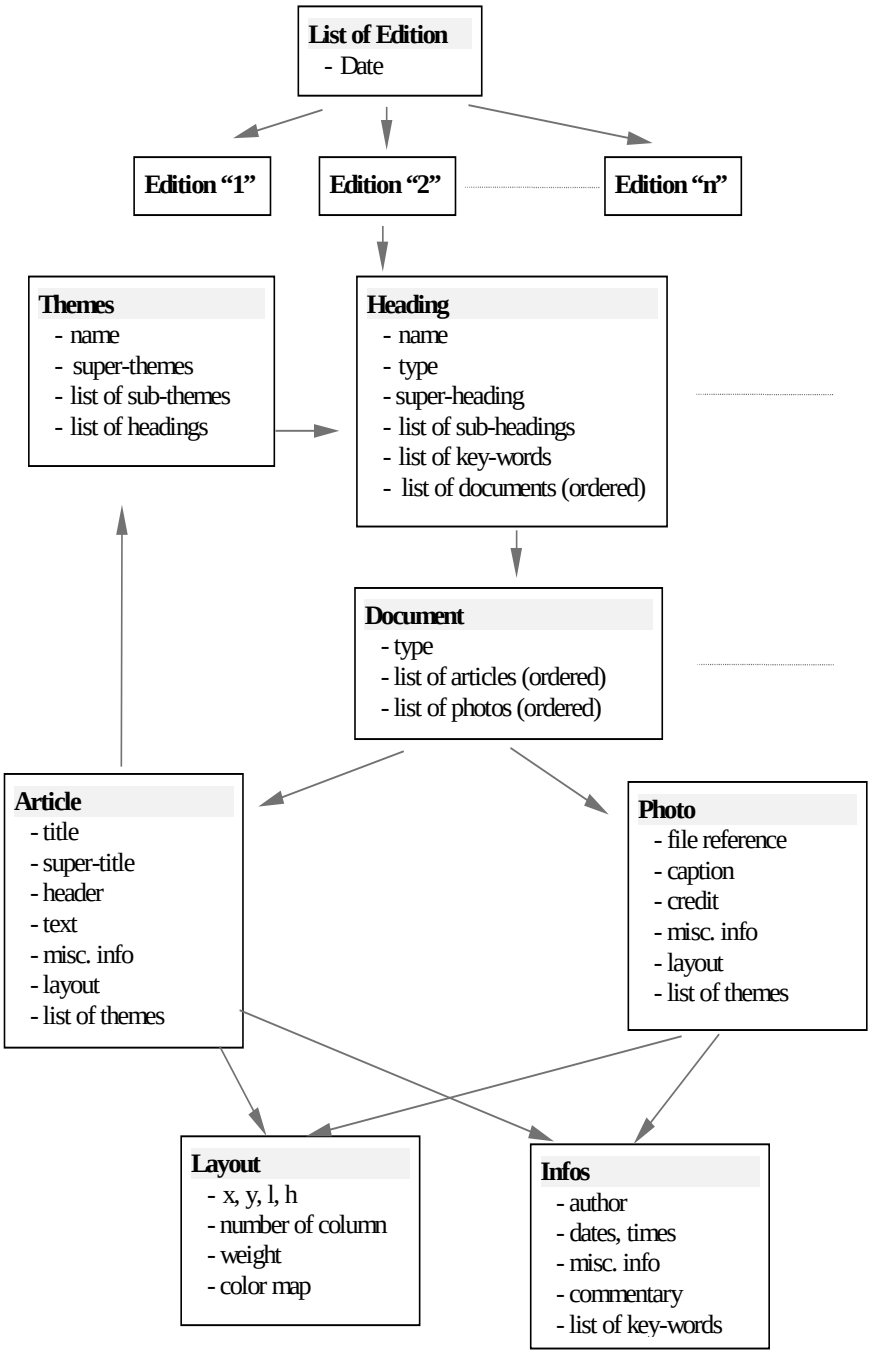

Figure 8: ETEL's Hierarchy of Data

This extraction of data is the first phase of the three major phases that are needed to create personalized newspapers, as illustrated by Figure 9. This first phase is ran every night, once the ETEL database has been created using the daily data generated by Ouest-France's journalists. On Figure 9, the ETEL database is represented as containing several articles (rectangles). The first major phase takes these articles, extracts the data they contain and then inserts them in the ETEL++ database. These ETEL++ articles are different from the ETEL's articles. They contain

additional information (meta-information, see Section 3.2) that is represented by the small black squares on top of each article. Note that ETEL++ builds an index (the triangle) over the inserted articles for the purpose of information retrieval.

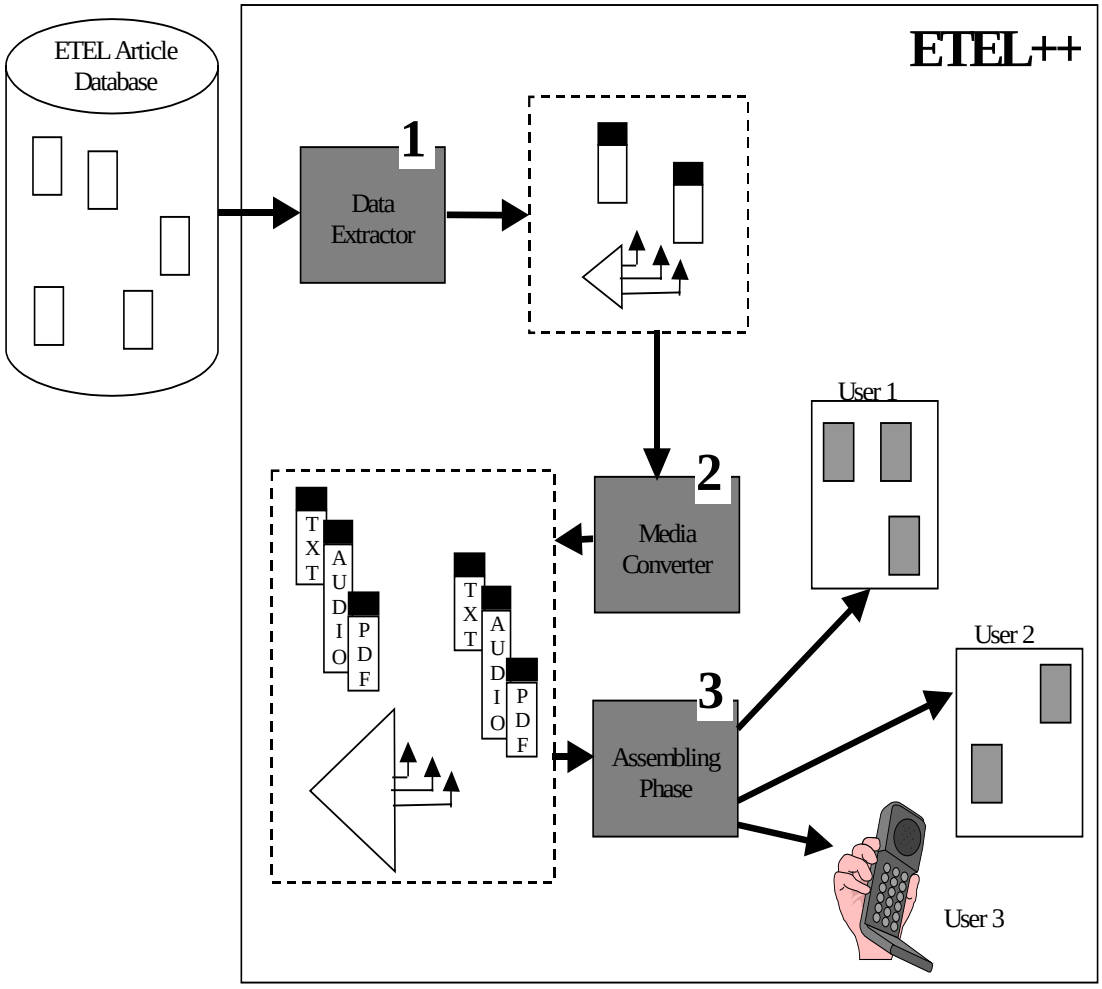

Figure 9: The 3 Major Phases for ETEL++

Once the data has been extracted, ETEL++ initiates its second major phase, namely the conversion of articles from one media to the other (see section 3.1.1.2). As it is apparent on the Figure, 3 representations of each article are generated (in this example): a text-based version, an audio-based version and a PDF-based version. Note that the index grows accordingly, since more data is stored inside ETEL++'s database. At the end of the phase 2, all publishable data has been generated (see Figure 3). Finally, the third major phase is triggered. Its goal is to select some relevant articles for a given user and then to assemble them such that they build the personalized edition of each user.

Phase 1 is triggered overnight and is ran on a machine that is physically located inside Ouest-France's buildings. Phases 2 and 3 are triggered at different times, depending on the way ETEL++ will deploy its service. In addition, these two phases can be run at many different physical locations, according to the service deployment hints. More information on when and

where Phases 2 and 3 are triggered can be found Section 4.2 of this paper and also in the *Service Deployment* documents.

4.1.2 Dealing with a Fax Machine and a Web-Browser

A powerful feature of ETEL++ is its ability to make the electronic newspaper accessible from multiple types of user terminals. We present in this use case the basic mechanisms that are used to prepare a personalized the edition for two particular types of terminals. We assume for this use case that ETEL++ knows that a particular user will be connected tomorrow morning, and that he will request his personalized edition to be delivered either on a desktop or on the fax machine in his office, if the desktop is used by another individual. Querying the Personal Profile of the user provides this knowledge. While this query returns possible terminal types from which our user can connect, it is necessary to interrogate the *User Access* workpackage to get an accurate description of those terminals (see Section 3.1.2.3). This is illustrated by Figure 10. It is necessary to know the type(s) of terminals a user is likely to use in order to retrieve from the database the relevant articles having the appropriate format. For this use case, we relevant articles that will ultimately be delivered are of two formats: plain text (for the fax machine) and HTML.

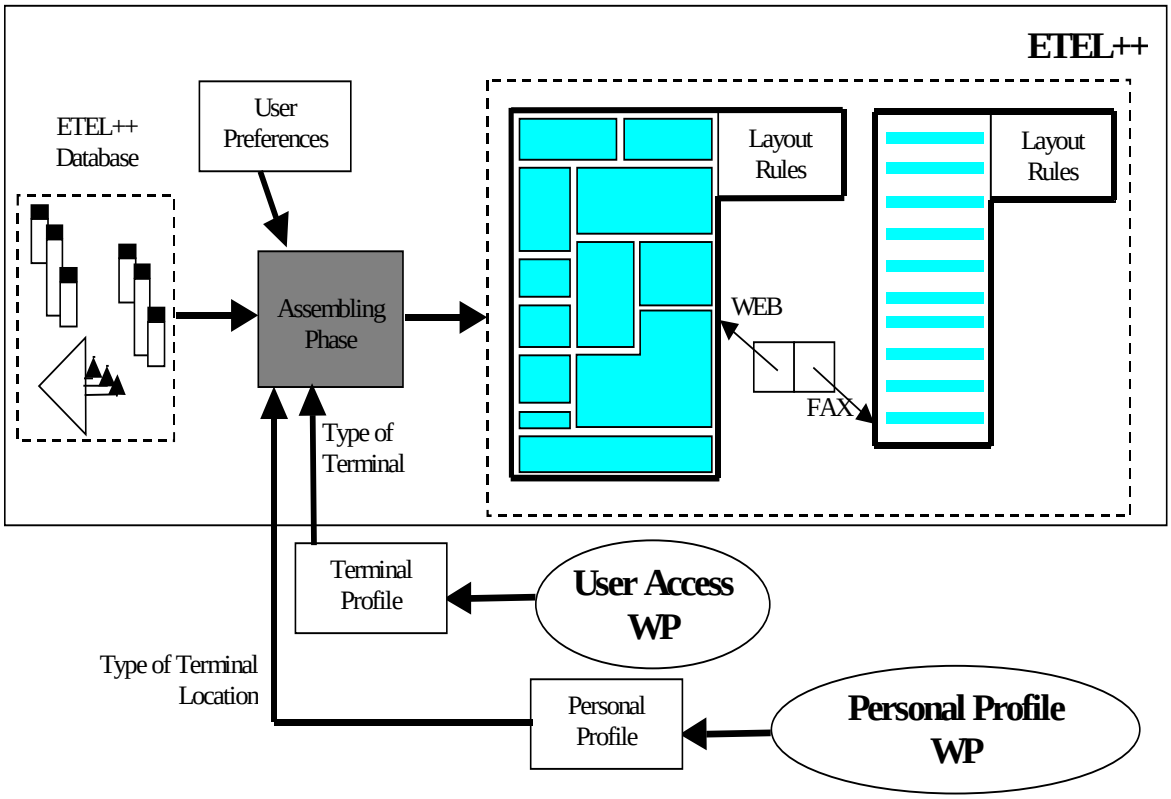

Figure 10: Preparing an Edition

Building an edition requires also the knowledge of the user preferences. Together, this knowledge enables the assembling phase to fetch the appropriate articles from the database and to assemble them such that they compose the personal editions. For this use case, there are *conceptually* two prepared editions that are very different: one for a Web-browser and one for a fax. Both are attached to our user and kept inside ETEL++, as illustrated by the Figure. (This description is at a conceptual level – the description of the preparation of editions *on a design level* would not be identical, however. For example, parts of different editions may be shared for efficiency reasons, and may not be *physically* kept inside ETEL++.)

Each edition is made of two types of data: content and layout. The content is the informational material, the layout is a set of rules that have to be applied and that impacts the actual look of the delivered edition. For simplicity, we chose to depict on Figure 10 the content formatted with respect to the associated layout rules. It is therefore rather intuitive to perceive the differences between the 2 prepared edition just by looking at the Figure. It is possible to deliver the editions once they are prepared. The delivery process is illustrated by Figure 11. Note that this delivery process assumes an on-line interaction between ETEL++ and our user. Off-line delivery is also possible, but beyond the scope of this use case.

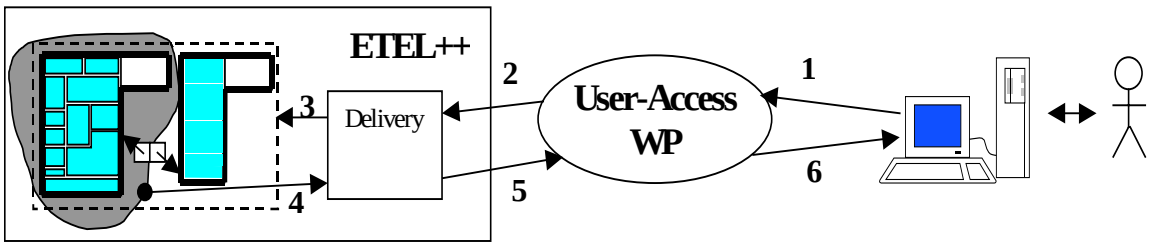

Figure 11: On-Line Delivery of a Prepared Edition

Six major steps are required to deliver his edition to our (on-line) user. These steps can be seen on Figure 11.

1. Connection time.
2. ETEL++ is made aware of the connection of the user. Together with this notification, ETEL++ receives a precise description of the actual terminal from which the user is connected. This description includes the physical characteristics of the terminal.
3. Depending of the actual type of terminal, the appropriate edition is selected.
4. The layout rules associated to the selected edition are checked against the description of the terminal received during step 2. During this step 4, ETEL++ computes an initial layout that matches the physical characteristics of the terminal. (Again, for simplicity, the Figures show editions that have already been laid-out.) For example, depending on the actual resolution of the end-user terminal, ETEL++ can compute a consistent layout.
5. During step 5, ETEL++ sends the edition back to the User Access workpackage. Conceptually, the information that is sent is made of (i) content, (ii) “style”

information providing a first cut of the layout elaborated during step 4 and (iii) additional rules to (possibly) cope with the final adaptation of the edition to the terminal. For example, these rules can trigger special transformations of the edition depending on the number of free color-cells that are available in the video-memory of the end-user terminal. If too little cells are available, then the rules may restrict the set of colors used in the edition. Although detailed information about terminals can be known, other information is hidden to reduce the overall complexity of coping with different terminals.

6. The final edition is eventually delivered.

Note this feature strongly rely on the software provided by FAST (workpackage *User Access)*.

Off-line delivery is not very different from the on-line delivery mechanisms that we presented here. With off-line delivery, ETEL++ can not rely on accurate information indicating the actual type of terminal the user is using (see step 1 above). Therefore, ETEL++ is likely to deliver to an off-line user several versions according to the possible inaccuracies of the agenda specifications. The last step (step 6 above) is also different since no user is connected. ETEL++ delivers the off-line editions to the Information Space that is attached to the user (see workpackage C). Therefore, when the user will eventually connect, he will found the delivered editions “waiting” in his information space, and ready for consultation.

4.1.3 Location-Dependent Information

Another interesting feature of ETEL++ is its ability to deliver context-sensitive data (see Section 2.3.2). We present here a use case that shows how ETEL++ delivers to a user weather forecast information that corresponds to the town from which the user actually connects. This use case is illustrated by Figure 12.

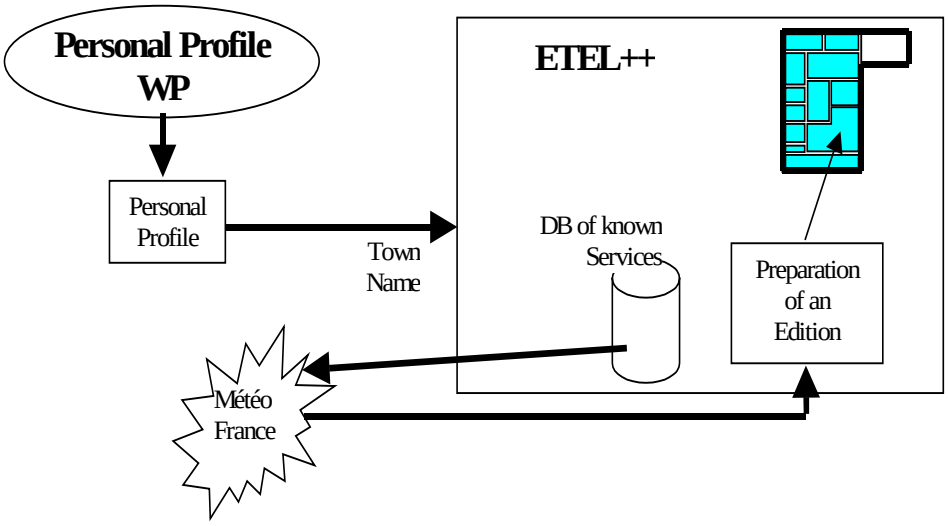

Figure 12: Location-Dependent Weather Forecast Data

ETEL++ knows it has to deal with context-sensitive data for a user because this is written in his user preference profile (not represented on the Figure). Therefore, during the preparation phase

of his edition, ETEL++ triggers a special task that is aimed at retrieving that type of data. In this example, this data is about weather forecast, and a rule says that the context in which this data must take its value is the physical location of the user when he will connect. This location can be known by interrogating the agenda, which is a subset of the Personal Profile attached to this user. In particular, it is possible to query for the town name in which the user said he will be connected on a certain day. ETEL++ therefore receives the name of a town.

This town name is used by ETEL++ to determine if it knows about a service that provides weather forecast data for the area corresponding to that town. ETEL++ uses the description of the data providers that have been presented Section 3.1.1.1. If no such service is known, then ETEL++ may send searching agents. If such a service is known (this is the assumption we make here), then ETEL++ requests the data to the data provider, and integrates it in the edition that is in preparation. Note this feature strongly rely on the software provided by UWE (workpackages *Autonomous Agents* and *Service Interaction*).

4.2 Enforcing Quality of Service Requirements

The goal of this section is to give a short overview of use cases illustrating the interactions between the workpackage G, *Service Deployment*, and ETEL++. More precise information can be found in the documents describing this workpackage. We briefly sketch here two policies that use the mechanisms provided by the Service Deployment workpackage. We first present how ETEL++ deploys its service to enforce Quality of Service requirement. We then present how a newspaper actually follows a user.

4.2.1 Deployment of ETEL++

Conceptually, the architecture on top of which ETEL++ is built is made of three entities that are represented Figure 13.

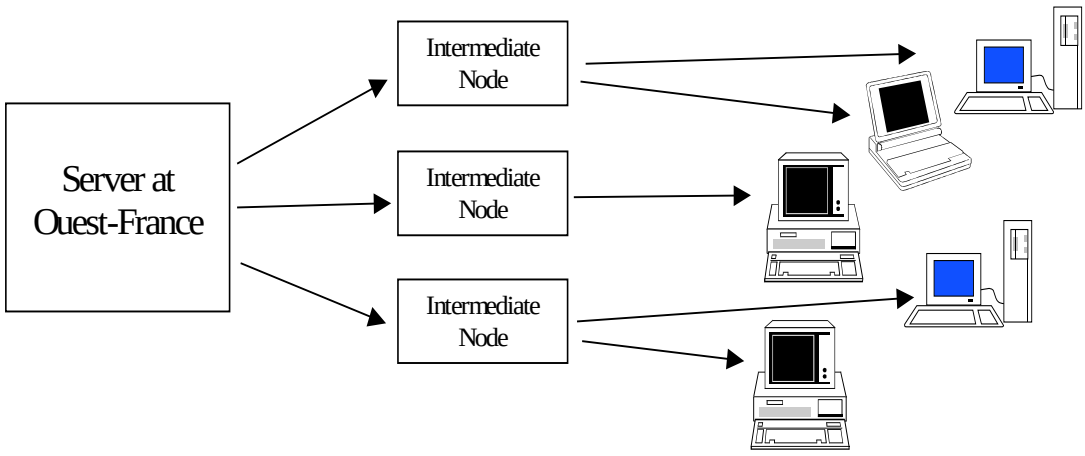

Figure 13: Server, Nodes and Terminals

These three entities are:

1. A server at Ouest-France. This server is the main data provider for ETEL++. This server is connected to several intermediate nodes as described next.
2. Intermediate nodes. Intermediate nodes are FollowMe-enabled machines. Each of these nodes is likely to have secondary storage and computing power available for performing FollowMe related tasks. To an intermediate node is potentially connected a set of user terminals as described next.
3. User terminals. User terminals can for example be personal computers. Users consult their personalized edition on user terminals. A user terminal *may* have secondary storage and computing power available for performing FollowMe related tasks.

As mentioned earlier in this document, the general goal of ETEL++ is to consume Ouest-France data, to transform that data and to produce personalized editions delivered to end-users. Three possible locations for data transformation can therefore be considered, given the entities depicted by the Figure 13. According to Figure 9, ETEL++ needs three phases for transforming the data. Obviously, the first phase (data extraction from the ETEL database to create the ETEL++ database) has to take place as close as possible to ETEL. Therefore, this first phase is executed by the entity called Ouest-France server. The two other phases (Phase 2: Media conversion and Phase 3: Assembling Phase) do not require to be executed inside this same server. It is possible to assign these two phases to one or more of the possible three locations of the Figure 13. For example, the overall cost of Phase 2 and 3 may be decreased if they are run on intermediate nodes. In this case, the server at Ouest-France would have to send elementary data to the nodes on which the edition preparation takes place. In this case, the degree of parallelism in the elaboration of editions is augmented.

Mobility makes easy for ETEL++ to migrate code, data or both to any machine.

4.2.2 Following a User

The agenda that is part of the Personal Profile gives the next location from which a user is likely to connect. Once this location is known, it is possible to determine what intermediate node is close to this place. Therefore, ETEL++ can use the features of mobility to send to this intermediate node the personalized edition of a user. Sending the edition is typically performed in advance, some time before the user actually connects (again, this knowledge comes from the agenda). Consequently, the response time for consulting his edition is kept to a minimum since data is always moved as close as possible to the user.

4.3 Typical User Interactions

We present here examples of interactions between a user and ETEL++. The descriptions stay at a logical level, that is, the actual design implementation may differ. In particular, the physical locations of data, and the physical flows of information going back and forth between a user and ETEL++ are likely to differ from the description here.

4.3.1 Basic Access from a Web-Browser

Users will typically access ETEL++ from a web-browser. A personalized edition will therefore look like a set of web pages with hyperlinks. The hyperlinks are used to navigate in the edition, that is, to jump from one article to the other, to click on a keyword and see displayed the corresponding article, etc. The web pages that will contain the articles are likely to look like the one depicted Figure 14. (This is an actual sample of a real screen dump. The final version of ETEL++ may provide another presentation of the pages within a web-browser.)

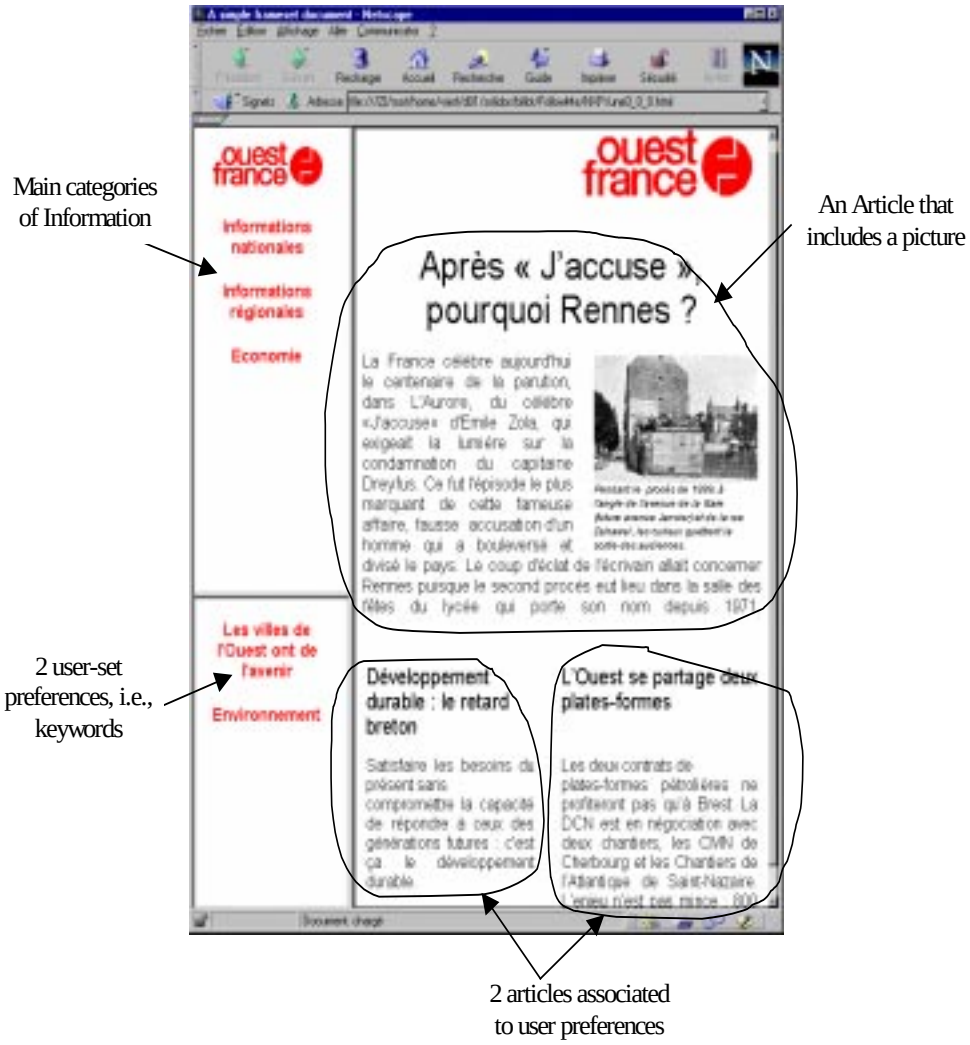

Figure 14: An Example of ETEL++ in a Web-Browser

We see on this Figure 3 major features of ETEL++: (i) 3 articles, and one of them contains a picture, (ii) 3 main categories of information (in this example “regional info”, “national info” and “economy”) and (iii) 2 keywords that have been set by the user and that specify his preferences (here a geographical coverage keyword and a thematic one). The categories of information and the keywords (the two frames on the left) are hyperlinks, and a click displays in the main frame the corresponding article(s). In the main frame, a click on the picture enlarges it.

4.3.2 Setting User Preferences

There are basically two ways for ETEL++ to know the preferences of its users. First, ETEL++ provides a set of web pages containing keywords associated to check-boxes. These keywords enable a user to explicitly state his personal interests (see Section 3.1.2.2). The other mechanism that ETEL++ uses is based on a learning process. ETEL++ can implicit discovery the user's interests by tracking his habits.

5 Conclusion

The world of electronic newspapers is evolving rapidly. It is predicted that electronic newspapers will become a source of big money. Premium services, like personalized newspapers, access to archives or multimedia support are an area of incredible growth opportunity for publishers and advertisers. Advertising revenues from the Internet and on-line services totalled about 55 million dollars world-wide in 1996 according the Jupiter Communications. Analysts predict it will rise to 343 million dollars this year and 5 billion dollars a year by the end of the decade.

In general, electronic newspapers are good applications when one need to show the performance of a large scale distributed system. The number of users to cope with, the volume of data to manipulate, the frequency of updates, the multiple media to support, the response times to enforce, all those constraints are highly demanding, and offering a good Quality of Service is a real challenge. FollowMe and ETEL++ are part of the game.

6 References

BIL96

Michel Banâtre, Valérie Issarny, and Frédéric Leleu. Etel : a newspaper-based distributed information system. In *Proceedings of the 1996 ACM SIGOPS European Workshop*, Connemara, Ireland, December 1996.

BILC97

Michel Banâtre, Valérie Issarny, Frédéric Leleu, and Boris Charpiot. Providing quality of service over the Web : A Newspaper-based Approach. In *Proceedings of the 6th International World Wide Web Conference*, April 1997, Santa Clara, USA.

Web Sites cited in this Document:

http://www.mediainfo.com

http://www.telegraph.co.uk

http://www.welt.de

http://www.washingtonpost.com

http://www.liberation.fr

http://www.lemonde.fr

http://www.nytimes.com

http://crayon.net

http://crayon.net/using/sample

http://www.le-petit-bouquet.com