Telephone Interface for Avalanche Warnings based on Information Server for Adaptable Content Delivery

Beat Signer, Moira C. Norrie, Peter Geissbuehler, and Daniel Heiniger

Dept. of Computer Science, ETH Zurich, CH-8092 Zurich, Switzerland
{signer,norrie}@inf.ethz.ch

Abstract. We present a system that provides an advanced telephone service for the dissemination of both national and regional avalanche forecasts in the Swiss Alps. The service enables members of the public and also mountain guides to access forecast information while travelling in mountain areas and, particularly, to be notified when entering regions of high risk. By telephone access, we include both voice and WAP-based access as well as combinations of both. The service is achieved through the integration of a special forecast content delivery database, including geographical data for location-dependent delivery, into the overall avalanche information system architecture. This database was implemented using the XIMA framework for adaptable content delivery, which is based on an XML server for the OMS Java data management system and XSLT presentation templates. The speech interface was implemented using VoiceXML.

1 Introduction

We are rapidly moving towards an information society in which communities of users will demand access to all forms of both personal and shared information from their workplaces, homes and on the move. Our research is focussing on providing technologies and methodologies to support the development of global information spaces that offer access to shared information from various forms of client devices and employing different modes of interaction. Specifically, we have exploited object-oriented, XML and web technologies to achieve general and flexible information management and content delivery solutions that can adapt to a constantly evolving world of new technologies and requirements.

In this paper, we describe a specific application system that has been developed based on our general information server framework. This enables us to explain the main concepts of the framework and show how it can be extended to deal with particular application requirements in terms of client devices and context-dependent delivery. The application consists of an advanced telephone service for the dissemination of avalanche warnings in the Swiss Alps and was developed in cooperation with the Swiss Federal Institute for Snow and Avalanche Research (SLF) in Davos. The service enables members of the public and also mountain guides to access forecast information while travelling in mountain areas and, particularly, to be notified when entering regions of high risk. This requires that the content delivered is dependent on both the geographical position of the user and the capabilities of the telephone used e.g. voice only or WAP-enabled. Further, it requires a mix of pull and push technologies to ensure that users can, not only enquire about avalanche risks, but also be sent warnings automatically when they venture into a high-risk area.

In the design of our framework, two key factors were critical. First, the separation of content from presentation and, second, the central role of a semantic information model. The first of these is essential in providing universal access from various client devices. The second factor ensures that we focus initially on defining the application semantics, rather than on document content. The resulting XIMA framework is based on OMS Java, a data management framework with a rich information abstraction layer [7, 15].

The need for adaptable content delivery has been widely recognised and it is very much an active area of research and development. However, work in this area can be divided into two main streams — web information systems and mobile computing. The web information systems community has tended to focus on the server side and the use of databases for the dynamic generation of web documents. This includes the document-based approaches of web content management systems such as Vignette [19], Obtree C4 [10] or Interwoven [5] and the information-based approaches of web database systems such as WebML [3]. Although some work has been done on support for different client devices, these systems tend to be optimised for HTML output generation, which makes it difficult to easily support new output channels such as VoiceXML. Within the mobile computing community, the focus has tended to be on the client devices and dynamic environments rather than on information management issues. As a result, information management issues dealt with tend to be those of context information rather than application information [13, 14].

Our aim is to integrate support for adaptable content delivery within the data management system. The database should not be simply regarded as a static repository of application data, but rather as an active, central component that manages all kinds of data and metadata about the application, its users and their environments. This approach ensures better scalability and flexibility in terms of system development and evolution and avoids the necessity of middleware portals that have been proposed as a means of supporting different client devices [12].

We begin in Sect. 2 with a description of the avalanche services currently offered by SLF and the motivation and requirements for extending these services. In Sect. 3, we describe how the existing architecture of the avalanche information system was extended with a content delivery database to provide for geographical- and device-dependent content delivery. Section 4 presents the general information management framework, XIMA, used to implement the content delivery database. The voice and WAP interfaces are described in Sect. 5 and 6, respectively. Concluding remarks are given in Sect. 7.

2 Avalanche Forecast System

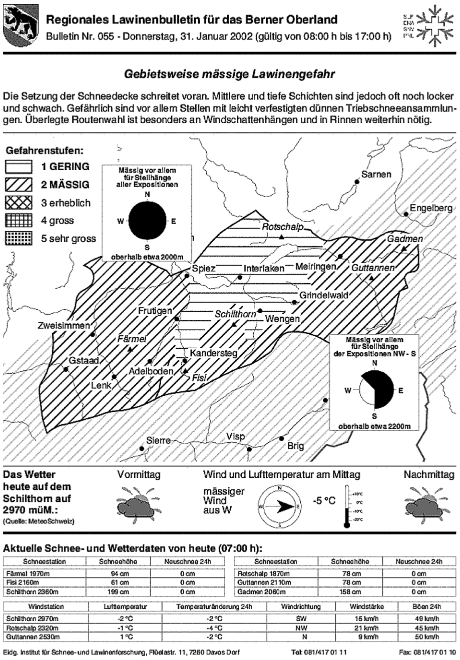
The Swiss Federal Institute for Snow and Avalanche Research (SLF) provides a range of information services in support of both the Swiss tourist industry and also the scientific community. Over the past decades, the risks to communities and transportation routes have been reduced due to improvements in structural techniques for damage avoidance. At the same time, improvements in the quality and availability of local avalanche and weather forecasts has meant that localised warnings in ski resorts reduce the risk to individuals through the closure of ski pistes. The focus has therefore turned to reduce the number of accidents to individuals such as mountaineers and ski tourers, who travel in uncontrolled areas of the Alps.

For a long time, avalanche predictions were based solely on the knowledge and experience of domain experts. In the 70s, research institutes started to develop prediction models that led to homogeneous, reproducible and more precise forecasts. The current network to measure the relevant input factors for effective avalanche forecasting consists of 65 automatic measurement stations installed by MeteoSchweiz (ANETZ), 75 SLF comparison stations and 60 automatic measurement stations operated by SLF (IMIS/ENET). The data set is further extended by feedback of tour guides and both local and regional specialists in terms of questionnaires.

SLF has access to collected measurement data, together with weather forecast data, historical avalanche data and models for snow coverage and movement. SLF uses its own model, NXD-REG, for regional avalanche prediction based on the statistical method of nearest neighbours [1]. The model splits Switzerland up into a hierarchy of regions starting with supra regions and going down to regions, partial regions, areas and, finally, partial areas. Avalanche forecasts are generated at the fine granularity of partial areas, but are then dynamically merged based on similarities between neighbouring regions into regions of higher and variable granularity. This avoids having unnecessary detail in terms of hundreds of small regions, while at the same time allowing the forecast to differentiate between small regions where there is currently significant variability in terms of forecast conditions. However, this means that, given a geographical position, we must be able to dynamically determine the smallest containing region that has an associated forecast.

The final generated regional forecasts include information such as height, exposition and a hazard level in the range 1 to 5. The forecast data is stored in a relational database and this data is used to generate avalanche bulletins as well as for future data analysis. SLF produces both national and regional bulletins as detailed in [2]. The national bulletin is a textual description of the avalanche situation in Switzerland along with a corresponding graphical representation in terms of a shaded map of Switzerland showing the various hazard levels.

A regional bulletin editor is used to produce a regional bulletin such as the example shown in Fig. 1. At the top of the bulletin there is a textual description of the avalanche forecast which is followed by a regional map indicating hazard levels. Detailed information about the weather and snow conditions is given in the lower portion of the bulletin.



Regionales Lawinenbulletin für das Berner Oberland
Bulletin Nr. 055 - Donnerstag, 31. Januar 2002 (gültig von 08:00 h bis 17:00 h)

Gebietsweise mässige Lawinengefahr

Die Setzung der Schneedecke schreitet voran. Mittlere und tiefe Schichten sind jedoch oft noch locker und schwach. Gefährlich sind vor allem Stellen mit leicht verfestigten dünnen Triebschneeansammlungen. Überlegte Routenwahl ist besonders an Windschattenhängen und in Rinnen weiterhin nötig.

Gefahrenstufen:

- □ **1 GERING**
- ▨ **2 MÄSSIG**
- ▦ 3 erheblich
- ▦ 4 gross
- ▦ 5 sehr gross

Das Wetter heute auf dem Schilthorn auf 2970 müM.:
(Quelle: MeteoSchweiz)

Vormittag

Wind und Lufttemperatur am Mittag

mässiger Wind aus W

-5 °C

Nachmittag

Aktuelle Schnee- und Wetterdaten von heute (07:00 h):

Schneestation	Schneehöhe	Neuschnee 24h	Schneestation	Schneehöhe	Neuschnee 24h
Färmel 1970m	94 cm	0 cm	Rotschalp 1870m	78 cm	0 cm
Fisi 2160m	61 cm	0 cm	Guttannen 2110m	78 cm	0 cm
Schilthorn 2360m	199 cm	0 cm	Gadmen 2060m	158 cm	0 cm

Windstation	Lufttemperatur	Temperaturänderung 24h	Windrichtung	Windstärke	Böen 24h
Schilthorn 2970m	-2 °C	-2 °C	SW	15 km/h	49 km/h
Rotschalp 2320m	-1 °C	-4 °C	NW	21 km/h	45 km/h
Guttannen 2530m	1 °C	-2 °C	N	9 km/h	50 km/h

Eidg. Institut für Schnee- und Lawinenforschung, Flüelastr. 11, 7260 Davos Dorf Tel: 081/417 01 11 Fax: 081/417 01 10

Fig. 1. Regional avalanche bulletin

Since for both the national and the regional bulletin the user has to detect his location on the hazard map, SLF additionally offers a local avalanche bulletin. The local bulletin effectively extracts local avalanche forecast data from the national bulletin based on the user's current position.

Access to the SLF bulletins is given in various ways including web access, fax registration services and the publication of national bulletin maps in newspapers. A limited voice telephone service consists of access to a recording of the national bulletin text. The desire to increase mobile access to avalanche warnings stimulated SLF to develop a service for WAP-enabled phones. This was done by generating WML from the HTML documents generated for web access to national bulletins. The basic WAP service only provides access to parts of the national bulletin texts. In agreement with Swisscom, they developed a further service for Swisscom customers where location-based text warnings are available based on geographical position as determined by Swisscom through triangulation of phone cells.

One can see that SLF is keen to disseminate avalanche information as widely as possible to all interested parties and to take advantage of all forms of information channels and new client technologies. However, the current solution to content delivery makes it difficult to react quickly to a rapidly evolving world of new technologies in such a way that full operational services are available on all channels. For this reason, we chose to develop a general information server framework for content delivery that can be easily adapted to both new applications and new technologies.

3 Architecture and Operational Overview

As described in the previous section, SLF gathers information from various input resources. This data is either further processed or directly stored in one of the avalanche forecasting system's storage containers as shown in Fig. 2. Based on the forecast database's content, SLF finally generates the bulletins to be distributed over different output channels. For every new output channel, a completely new application has to be built based on data stored in the avalanche forecasting system. Parts of existing solutions can be "reused", as is done in the implementation of the current WAP service where WML documents are generated by parsing the HTML output and extracting the desired content. It is desirable to avoid the resulting double indirection of first generating an HTML page, which involves dealing with visualisation aspects, and then having to parse the HTML document to build the requested WML page.

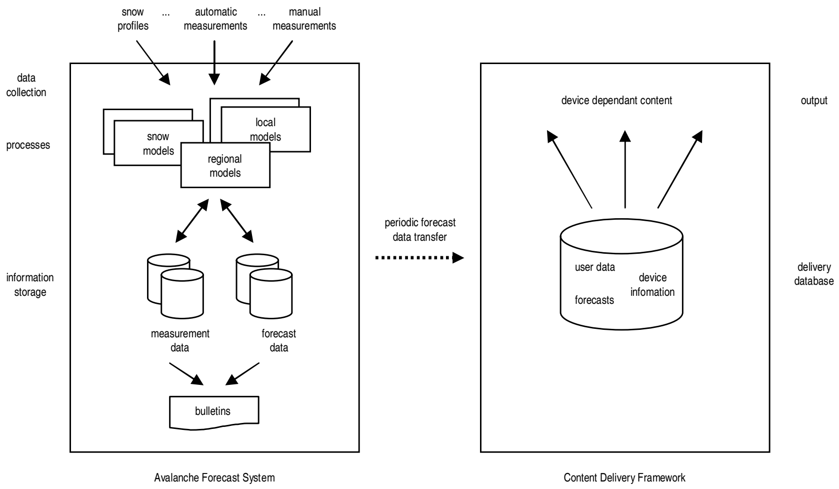


Fig. 2. Overall architecture

By introducing a content delivery framework with a clear separation of content and presentation, the task of adding support for new types of client devices is dramatically simplified. This was done through the introduction of a special component — the content delivery database — which deals with all aspects of client access to current avalanche forecasts.

The content delivery database shown on the right-hand side of Fig. 2 can be separated into three major parts. All relevant information about forecasts, weather information etc. is stored using the OMS Java data management framework [6]. Instead of building multiple "native" applications to support different output channels based on the information stored in this forecast database, we use an intermediate XML representation of the delivery database's content. The eXtensible Information Management Architecture (XIMA) generates the desired output format by applying XSLT transformations to the dynamically generated XML documents. For optimal support of different client devices, the database further contains device-specific information such as screen resolutions etc. Finally, the experienced user can customise the generated output by optionally providing personalised information.

Instead of having a persistent connection between the avalanche forecasting system and the content delivery framework, the dynamic delivery database's content is updated periodically. With this clear separation of the avalanche forecasting system and the content delivery framework, we can avoid increasing the complexity of the forecasting system by mixing content generation and content delivery.

We now describe the information model of the delivery database shown in Fig. 3 in more detail. The delivery database can be divided into two main parts. The static part contains spatial information about the different regions and defines the hierarchical ordering of different areas. Since changes to this information are quite rare, the static delivery database part is built once and does not have to be synchronised during the periodic data transfer from the avalanche forecasting system to the content delivery framework. On the other hand, we have the dynamic database part containing the relevant avalanche forecasting information.

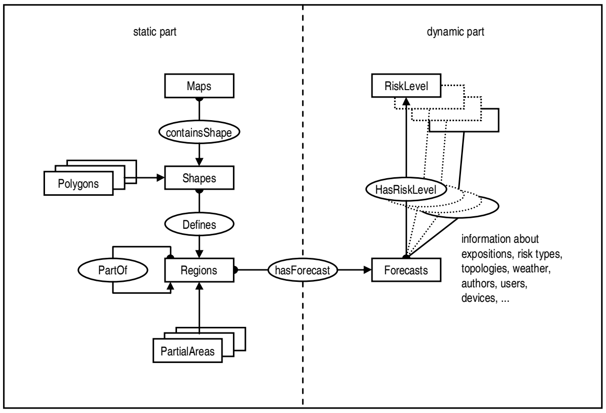


Fig. 3. Delivery database

To provide location-based information delivery, we define a shape (polygon) corresponding to each of the partial areas introduced by SLF. These shapes are combined to form a map of all different subregions of Switzerland. By providing the current position (x- and y-coordinates) as an input parameter, we are then able to get the required shape by making a lookup in the map. At the moment we support GPS coordinates and Swiss National Coordinates [17] as found on most Swiss maps. Once we have fetched the relevant shape, we get access to the corresponding region over the *Defines* association. For the sake of simplicity, in Fig. 3, we just show the most relevant components of the information model with respect to location-dependence and do not detail the regional and forecast information.

As explained in Sect. 2, SLF defines a whole hierarchy of partial areas, areas, regions and so on. Our information model classifies the regions into these different categories defined by SLF and builds up the required hierarchy using the *PartOf* association. Since the forecasting model merges equal forecasts for adjacent subregions to form a single forecast for the next higher superregion, it may happen that the region bound to the shape returned by the map lookup is not directly associated with a corresponding avalanche forecast. In such a case, we just have to recursively go up in the hierarchy of regions until a region with an associated forecast is found.

In Sect. 5 and 6, we discuss in detail the features of the voice and WAP interfaces, respectively. Here we describe how the system can, not only be used in one or the other of the two modes (either WML pages or voice input/output), but also in a mixed-mode manner. This means that a user can send a request by voice over his mobile phone and will receive the answer as an WML page which is displayed on his mobile phone.

To achieve mixed-mode functionality, we needed a way to send WML content to a client device without receiving an explicit WAP request (the corresponding request will come over a different input channel from a VoiceXML document). We therefore used the Openwave WAP Push Library, which provides Java APIs encapsulating the WAP Push Access Protocol (PAP) [11]. By using the Service Loading protocol (SL) we can push the appropriate WML avalanche forecast pages to the user's mobile device after having received a voice request.

The push technology is not only useful to enable switching between different modes, but also to automatically send new content to the end user without getting an explicit request. One possible application of such a service is that a user can register to be notified if he enters into a region which has a hazard level higher than a certain value. The avalanche warning system will then track the user's position and trigger a push notification as soon as the risk level is higher than the user-defined threshold value. One problem of using a pushing service to deploy WML pages is that the WAP push technology introduced by the WAP 1.2 specification is not yet supported by most existing mobile phones. A possible workaround for this temporary lack of push support is to use the short message service (SMS) for automatic information delivery (accepting the drawback that interactive browsing will no longer be possible).

4 XIMA Information Server

To meet the requirements of universal client access to information, it is vital that support for adaptable content delivery is an integral part of the information management framework, and not simply an afterthought. There must be a clear separation of content and presentation, with the underlying data management system managing the content and a special access layer generating the appropriate presentation document. The choice of presentation may depend on one or more access factors such as the client device, the user, the location and possibly other context factors such as the access history.

We developed the XIMA framework specifically to support the rapid development of application systems requiring access from various forms of devices. The framework is web based and uses Java servlets for request processing. An overview of the XIMA server architecture is presented in Fig. 4.

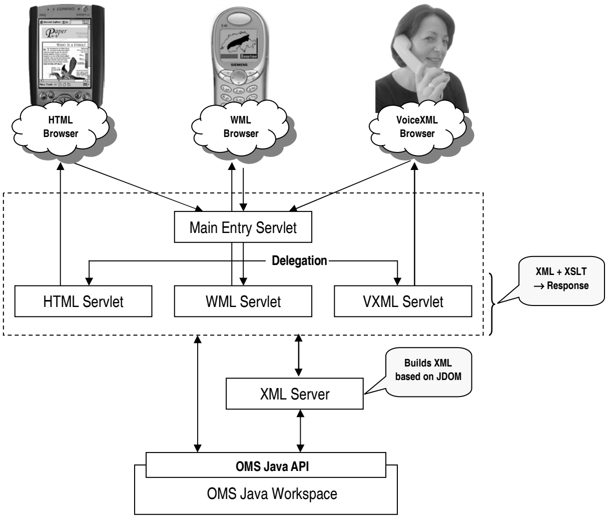


Fig. 4. XIMA framework for universal client access

For a specific application, all client access is via a single Java servlet which detects the type of client device from the user agent type specified in the HTTP request header and delegates the handling of the request to the appropriate servlet. For example, we show servlets to handle requests from HTML browsers, WAP phones in terms of WML browsers and voice telephones in terms of VXML (VoiceXML).

The underlying database management system is OMS Java, an object-oriented data management system based on the OM model [8]. The OM model is a generic, semantically rich model that distinguishes typing and classification of objects and includes support for managing associations between objects. It is beyond the scope of this paper to describe the OM model and OMS Java in detail. Further details of OMS Java can be found in [6, 7, 9] and a discussion of the features of OMS Java to support web site engineering is given in [15].

The request handling servlets access the database by connecting to an OMS Java workspace via the OMS Java API. The connection may either be direct or via the OMS Java XML server. Direct connections deal with requests that do not involve data retrieval such as updates or checking membership of an object in a collection. The XML server forwards requests to the OMS Java workspace and generates XML representations for any data objects to be returned to the servlets. The requesting servlets then use the appropriate XSLT templates to transform the XML results to the corresponding documents to be returned to the client. It is important to note that we are not storing XML documents, but rather generating them dynamically based on a hierarchical view of application data derived at access time.

To support rapid application development, we use generic XSLT templates to provide generic browsers and editors for the current set of supported client device types. Adding support for a new type of client device simply involves implementing the corresponding servlet and writing the appropriate XSLT templates. The system architecture and the implementation of a community diary using the framework are described in detail in [15]. This includes a description of how we dynamically split large objects in order to be able to provide access to these objects via devices such as WAP-enabled phones which have severe memory limitations.

Following the publication of the VoiceXML standard in March 2000 and the availability of various related tools (e.g. IBM Speech ML and Motorola VoxML), we extended the XIMA framework to support voice telephones as a type of client device. The difficulty here lay less on the technical side and more in the design of appropriate speech interfaces. Details of the technical implementation and the design of a generic voice browser are given in [16]. In the next section, we discuss the specific issues related to the development of a speech interface to the avalanche information system.

In general, the data managed by OMS Java will consist of application data and also any relevant client data. The client data may include information about client devices, users and their preferences or any other form of data that provides an element of context-dependent operation. For example, in the case of the avalanche database, the information to be delivered depends on location and hence the server database also manages geographical data as described in the previous section.

5 Voice Interface

The Voice Extensible Markup Language (VoiceXML) is an application of XML which enables interactive access to the web through standalone voice browsers or regular phones. One of its major goals is to bring the advantages of well established web-based content delivery techniques to Interactive Voice Response

(IVR) applications. Application navigation works by voice recognition or the use of Dual Tone Multi Frequency (DTMF) keypad input. The resulting aural responses feature digitised audio or synthesised speech output. Details of VoiceXML can be found in the VoiceXML on-line forum [20]. In this section, we discuss interface design issues that we encountered rather than details of VoiceXML itself.

During the design phase of the voice interface, the following three main aspects had to be considered. First of all, the users and their behaviour had to be analysed. Based on this analysis, the system functionality requirements were specified and the user interfaces designed. As a guideline for the design of the voice interface, we used the IBM Voice Server Programmer's Guide [18].

Two groups of users are targeted by the application: the regular *Business Users* (mountain guides, ski instructors, etc.), who use the system every day, and the *Holiday Users* (mountaineers, snow boarders, etc.), who require the avalanche forecasting services only occasionally. These two user profiles provided us with some general guidelines for the high-level design of the voice interface in terms of the frequency of use, a user's expertise in the application domain, etc.

A major question was whether to use recorded prompts or synthesised speech only. The region names, risk levels, risk types, topologies and directions are static data and therefore it would be possible to record each of these keywords to make the output sound more human. The problem is that the recording of single words is not sufficient because the same word can have different endings based on its position within a sentence. In the current version of our delivery database, we therefore used synthesised speech for all output.

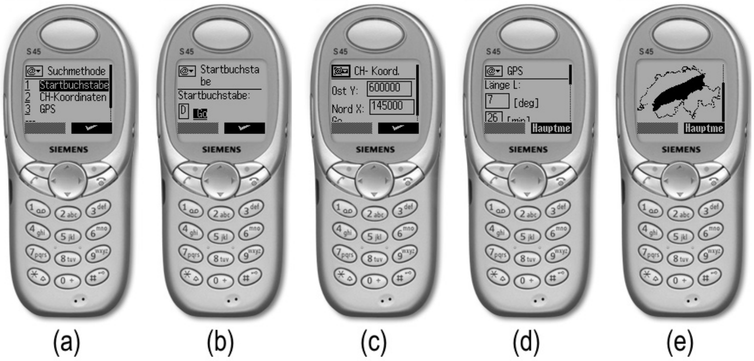
A crucial part in the development of the voice avalanche service was the design of the grammar since it has a direct impact on the quality of the voice recognition. This task was simplified due to the fact that a user does not have an unrestricted choice for possible inputs. In the main menu for instance, he can only choose between a national, a regional or a local forecast. A larger part of the vocabulary forms the names of the different regions. Finally, there are some commands to navigate through the application.

6 WAP Interface

The Wireless Application Protocol (WAP) defines a standard for content delivery over wireless communication networks. It defines a network stack and introduces several new networking protocols with similar functionalities as HTTP and TCP. The Wireless Markup Language (WML) is another XML application, which is used to publish content on WAP-enabled devices. WML is optimised for the relatively small displays and memory sizes of today's mobile devices and further introduces some new programming constructs (in contrast to HTML) such as variables and events. More details about the Wireless Application Protocol and the Wireless Markup Language can be found in the WAP Forum [21].

Since the users of the WAP application will more or less be the same as the users of the voice avalanche service described in the previous section, we could profit from the user analysis already performed for the voice interface. Further, we used several WAP user interface design guidelines to make the final application simple, fast and intuitive [4]. While navigating the application, the user should recognise the key tasks and the extent of the content as soon as possible. We decided to keep lists with links as short as possible to avoid time consuming scrolling. The overall information structure should rather be broad and flat than deep. The user is trying to find the relevant information in the shortest possible time and each indirection (link) leads to a delay. The number of clicks to reach the desired information is one factor to measure the efficiency of a WAP application. Whenever possible, we tried to avoid queries that force the user to enter alphanumeric text by using selection lists.

The main menu of the WAP user interface allows the user to choose between one of the three bulletins offered by SLF (national, regional or local). For each of the succeeding cards, we added either a soft key or an explicit link back to the main menu, allowing the user to go back to the main entry point. To access the information for a specific region, the local bulletin can be used. The current implementation of the local bulletin offers three different search methods to find the desired regional forecast. The user can either choose the region by name, by Swiss National Coordinates or by entering the GPS coordinates as shown in Fig. 5.a.



Startbuchstabe: D

Ost Y: 600000
Nord X: 145000

Länge L:
7 [deg]
26 [min]

Fig. 5. Local avalanche forecast on WAP phone

By entering the start letter of a region, the user obtains a list of all regions starting with the selected letter as shown in Fig. 5.b. He can then further choose the desired region from the list. Another possibility to access a region is by entering the corresponding Swiss National Coordinates (see Fig. 5.c). Last but not least, the region can be automatically selected by using the positioning information of a GPS-enabled mobile phone. At the moment, there exist no mobile phones with GPS support and the user has to manually enter the GPS coordinates of his current position as shown in Fig. 5.d. The interface will be adapted to support automatically generated GPS coordinates as soon as the corresponding mobile phones are available.

To improve usability for the inexperienced *Holiday User*, we added additional help functionality. If a user obtains a regional avalanche forecast for a specific part of Switzerland, but he does not know where the specific region is located, he always has the possibility to get a map of Switzerland showing the mentioned region as shown in Fig. 5.e.

Additionally, all keywords (scientific terms used by SLF) are linked to a WAP page giving a short description of the corresponding term. For example, if a user does not know a certain term which is used to describe the exposition of an avalanche forecast, he can just click on the exposition type to get additional explanations.

7 Conclusions

We have presented an application providing a telephone interface for avalanche warnings based on the XIMA framework. We have described how the system not only allows mobile users to obtain forecasts via both voice and WAP services, but also provides mixed-mode operation. For example, voice recognition could be used to recognise a request given as a spoken place name, and the result is returned as a WML document, possibly inclusive of map images.

In the case of the current avalanche system, content delivery is adapted based on geographical location and client device. The approach used can be generalised to deal with other forms of context-dependent delivery such as temporal or work activity contexts. Further, we consider it important to integrate user concepts in the system in order to support both personalisation and controlled information sharing. We have developed other applications where user profiling and customisation were central, but as yet we have not integrated this feature into the avalanche system. The avalanche warning system currently is evaluated by SLF and may provide input for the next generation of SLF's avalanche warning services.

References

1. B. Brabec. A Nearest Neighbor Model for Regional Avalanche Forecasting. In *Proceedings of the 14th International Workshop on Statistical Modelling*, Graz, Austria, July 1999.
2. B. Brabec. *Computergestuetzte regionale Lawinenprognose.* PhD thesis, ETH Zurich, 2001.
3. S. Ceri, P. Fraternali, and A. Bongio. Web Modeling Language (WebML): a modeling language for designing Web sites. In *Proceedings of WWW9, 9th International World Wide Web Conference*, Amsterdam, Holland, May 2000.
4. T. Grimstad, H. Stegavik, and E. Saastad. User Interface Design Guidelines for WAP Applications. *Telenor Mobile Communications*, 2000.
5. Interwoven Inc. Application Development Using Interwoven: Version 1.1. *Whitepaper*, December 2000.

6. A. Kobler and M. C. Norrie. OMS Java: Lessons Learned from Building a Multi-Tier Object Management Framework. In *Proceedings of OOPSLA'99, Workshop on Java and Databases: Persistence Options*, Denver, USA, January 1999.
7. A. Kobler, M. C. Norrie, B. Signer, and M. Grossniklaus. OMS Java: Providing Information, Storage and Access Abstractions in an Object-Oriented Framework. In *Proceedings of OOIS'2001, 7th International Conference on Object-Oriented Information Systems*, Calgary, Canada, August 2001.
8. M. C. Norrie. An Extended Entity-Relationship Approach to Data Management in Object-Oriented Systems. In *Proceedings of ER'93, 12th International Conference on the Entity-Relationship Approach*, Arlington, USA, December 1993.
9. M. C. Norrie, B. Signer, and A. Kobler. OMS Java - Object-Oriented Framework and Data Management System. Manual, Institute for Information Systems, ETH Zurich, 2000.
10. Obtree Technologies Inc., http://www.obtree.com.
11. Openwave Developer Program, http://developer.openwave.com.
12. K. Rindtorff and M. Welsch. Pervasive and Mobile Computing Middleware: How to Support Pervasive Internet Portals. *INFORMATIK, Journal of the Swiss Informaticians Society*, October 2001.
13. D. Salber, A. K. Day, and G. D. Abowd. The Context Toolkit: Aiding the Development of Context-Enabled Applications. In *Proceedings of ACM CHI'99, Conference on Human Factors in Computing Systems*, May 1999.
14. B. N. Schilit. *A System Architecture for Context-Aware Mobile Computing.* PhD thesis, Columbia University, 1995.
15. B. Signer, M. Grossniklaus, and M. C. Norrie. Java Framework for Database-Centric Web Engineering. In *Proceedings of WebE'2001, 4th Workshop on Web Engineering (in conjunction with 10th International World Wide Web Conference)*, Hong Kong, May 2001.
16. B. Signer, M. C. Norrie, P. Geissbuehler, and D. Heiniger. Aural Interfaces to Databases based on VoiceXML. In *Proceedings of VDB6, 6th IFIP Workshop on Visual Database Systems*, Brisbane, Australia, May 2002.
17. Federal Office of Topography, Wabern, Switzerland, http://www.swisstopo.ch, 2002.
18. IBM Voice Server Software Developers Kit (SDK) Programmer's Guide, http://www-4.ibm.com/software/webservers/voiceserver/progguide.pdf.
19. Vignette Corporation. Vignette Content Management Server. *Whitepaper*, February 2001.
20. VoiceXML Forum, http://www.voicexml.org/.
21. WAP Forum, http://www.wapforum.org.