P1.47 INTERNET TECHNOLOGIES FOR ENABLING FLOOD EMERGENCY EARLY WARNING SYSTEMS: THE ARNO RIVER BASIN CASE

Paolo Mazzetti * , Stefano Nativi, Lorenzo Bigagli, and Marco Mancini
University of Florence, Florence, Italy.

1. APPLICATION DOMAIN

This work describes the MIMI system developed as part of a Project funded by the Italian National Authorithy for the Arno river basin.

MIMI system has been developed to implement a solution for flash flood emergency early warning for the Arno river basin that is the third biggest river basin in Italy. The design and implementation of the system architecture was addressed to the following main requirements:

- access to legacy data stored in existing information resources;
- nearly real-time processing;
- scalable approach;
- visualization of warning situations through Intranet terminals;
- multi-channel management of alert information;

1.1 The Starting Scenario

Flash flood early warning is a very complex activity, involving a decision-making process based on meteorological and hydrological situation description and forecasts.

Anyway, the reliability of information resources is not sufficient to make a good early warning system since communication language and semantics problems arise.

Indeed, the decision-makers are often not technical persons, although they can be supported by technicians, they are not scientists, or computer engineers, instead they are often politicians.

This is the main reason because they could not effectively utilize available information resources data, unless they have been processed to provide high-level integrated views, that is far from the raw dataset plotting.

The starting scenario for the considered application domain was made of several public bodies and research centers each managing different data and information resources dealing with the basin hydrological and meteorological situation.

Those information were delivered to the decision-makers with few harmonisation orocessing, often utilising diverse rendering solutions (e.g. diagrams, images, statistics table, ets.). Moreover, the syncronisation of information often resulted cumbersome.

Figure 1 depicts that situation. The main information sources are shown: the Universities and National Research Council (CNR) sites provided raw and processed satellite data; the National Hydrographic Service manages a really effective raingauge network; the Regional Meteorological Laboratory runs a Local Area Model (LAM) working out short-term meteorological forecasts.

Decision-makers could access those information connecting to the proper data servers. The information were not integrated or otherwise processed to make them more suitable for decision-makers.

Moreover they must make a decision in a very short time reducing the risk of false alarms and missing detections. Thus, it was very difficult to extract useful information about possible hazards from that variable amount of data, and, due to the lack of integration, synchronization and high-level abstraction rendering, their usefulness was questionable.

? *Corresponding author address*: Paolo Mazzetti, University of Florence - Polo Didattico di Prato
Piazza Ciardi 25, 59100 Prato, Italy; e-mail: mazzetti@pin.unifi.it

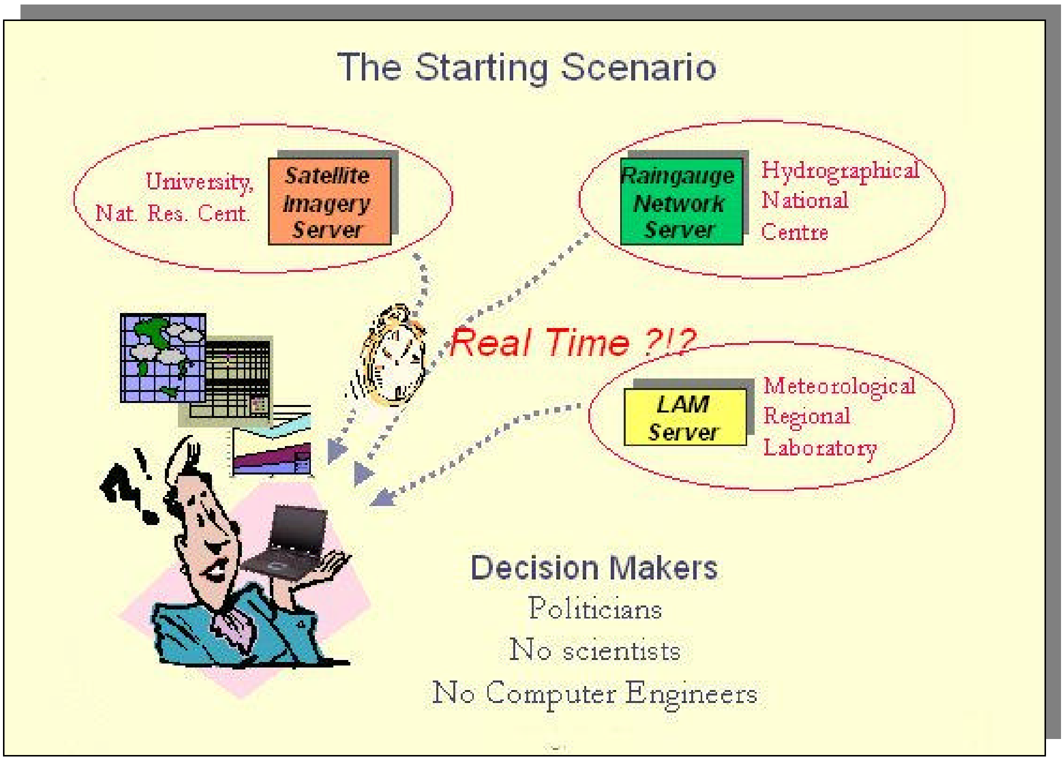

Fig. 1 Starting scenario

1.2 The present scenario

To face the previous situation a different approach was developed. A complex framework was conceived to extract and synchronize useful information from heterogeneous data sources. The framework was based on an hydrological model developed by the Polytechnic of Milan. The hydrological model integrates data about the current situation (i.e. raingauges) and data about the forecasted situation (i.e. data from the LAM) to provide high-level information integration and rendering about the possibility of flash flood emergency situations for the Arno river network.

Figure 2 shows the present scenario. A model server runs the application implementing the hydrological model. It accesses the required data sources through the Internet. The current situation is provided by the raingauge network data properly processed to build a 2D rain map, while the forecasted situation is provided by the LAM also through the processing of satellite data. The decision-makers could use different devices to access the results of the Model Server processing, visualizing them in a format that is suitable for that particular device and user.

In particular the framework model provides information at high-level of abstraction that can be easily utilized in a tool for supporting decision-makers.

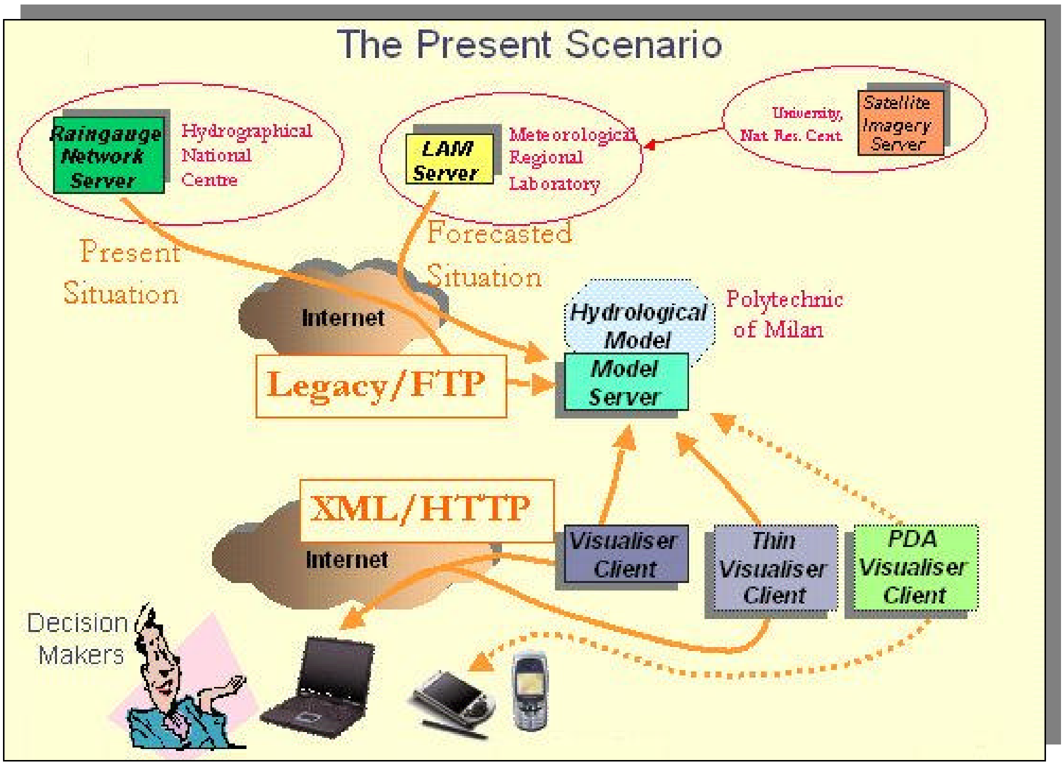

Fig. 2 Present scenario

1.3 *The future scenario*

In the near future, the support of a more heterogeneous and complex scenario is planned. According to this scenario many other useful information resources will be federated.

Figure 3 shows that situation, where a hydrometers information resource, a soil humidity data resource and a radar observations resource could be accessed through the Internet joining the overall framework and therefore, feeding in the model server. Obviously, the model needs to be extended to consider all these new information sources making result much more reliable and richer for decision-making process.

Moreover a new type of client device will be supported: a PDA (Personal Digital Assistant); hence, decision-makers will be able to consult the basin situation in real-time also when they are on the field to monitor the emergency.

This extension are possible due to the component-based nature of the system architecture as well as the webservices technologies utilized to implement it.

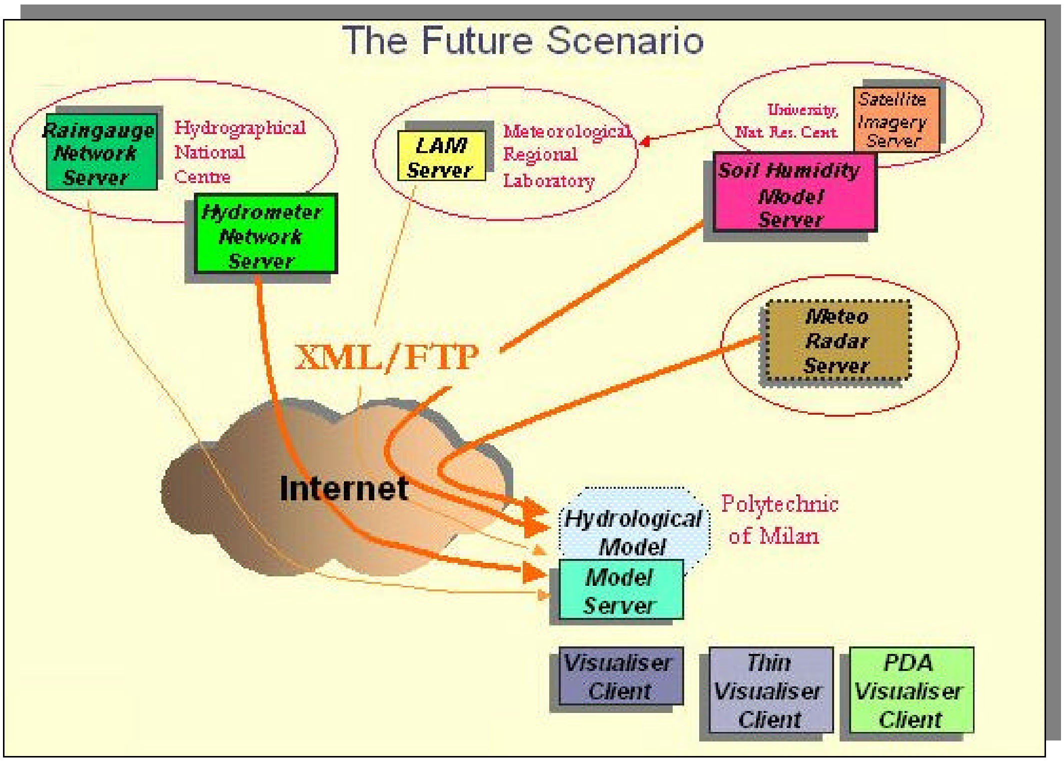

Fig. 3 Future scenario

2. USER ONTOLOGY

The first step to provide such demanding users with a suitable and useful tool, is the definition of the system ontology: a controlled, hierarchical vocabulary for describing a knowledge system. In short we need to use the users' vocabulary to make them easier to understand the provided information. For example, in our case, decision-makers generally are not familiar with technical terms like Antecedent Moisture Content (AMC) for soil humidity estimate. Even if it is an important hydrological parameter it can be confusing to use it as an alert indicator. Decision-makers could instead easily handle concepts like critical thresholds for cumulated rainfall and similar.

Thus a preliminary activity was aimed to define a possible ontology for decision-makers. This required:

- the definition of a taxonomy that is the vocabulary of basic terms defining the decision-makers view about the flash flood emergency early warning problem;
- the collection of the information about the relationships between these basic terms in the specific domain;

This activity was performed resorting to hydrologist expertise and through the interaction with the National Basin Authority of the Arno River personnel.

As far as the solution approach ontology is concerned, it is possible to distinguish the following main sections:

1. the Basin realm;
2. the Rainfall Event realm;
3. the Rainfall Map;
4. the Raingauge Network realm.

The following Figures depicts a diagram –according to the RDF graphical notation- for each realm.

According to such diagrams, for instance, the concept basin (*mimi#Basin)* is the area interested (*mimi#interstedArea)* by a rainfall event concept (*mimi#RainfallEvent)*. In the same way a rainfall map concept (*mimi#RainfallMap*) is the data reporting (*mimi#data*) a rainfall (*mimi#rainfall*); a raingauge network (*mimi#RaingaugeNetwork*) is the container (*mimi#container*) of raingauges (*mimi#Raingauge*). From the depicted diagrams is possible to work out all the other axioms that constitute the knowledge model shared between the system and decision-makers.

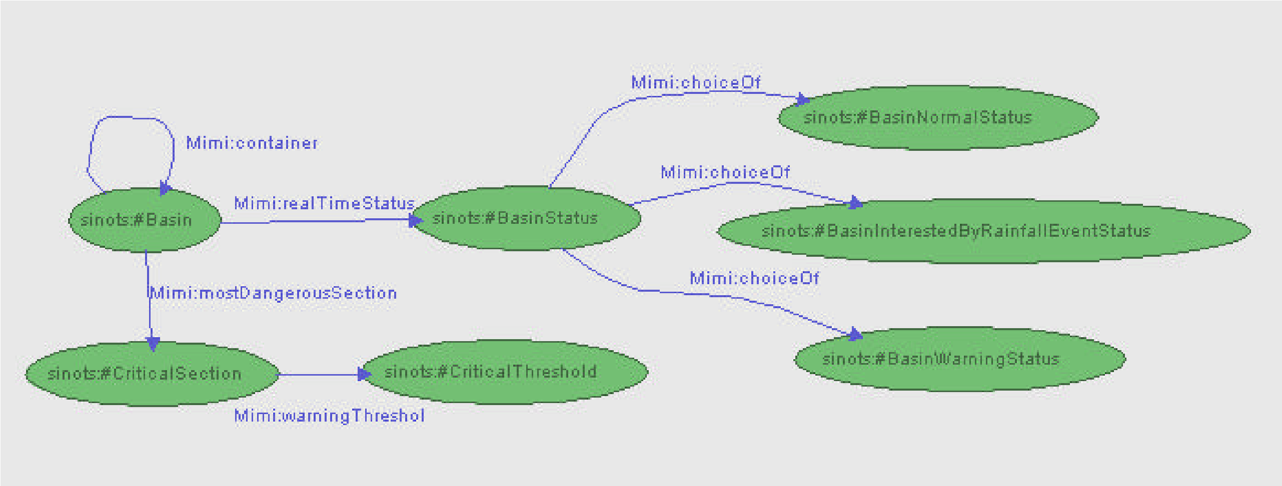

Fig. 4 Ontology diagram – Basin realm

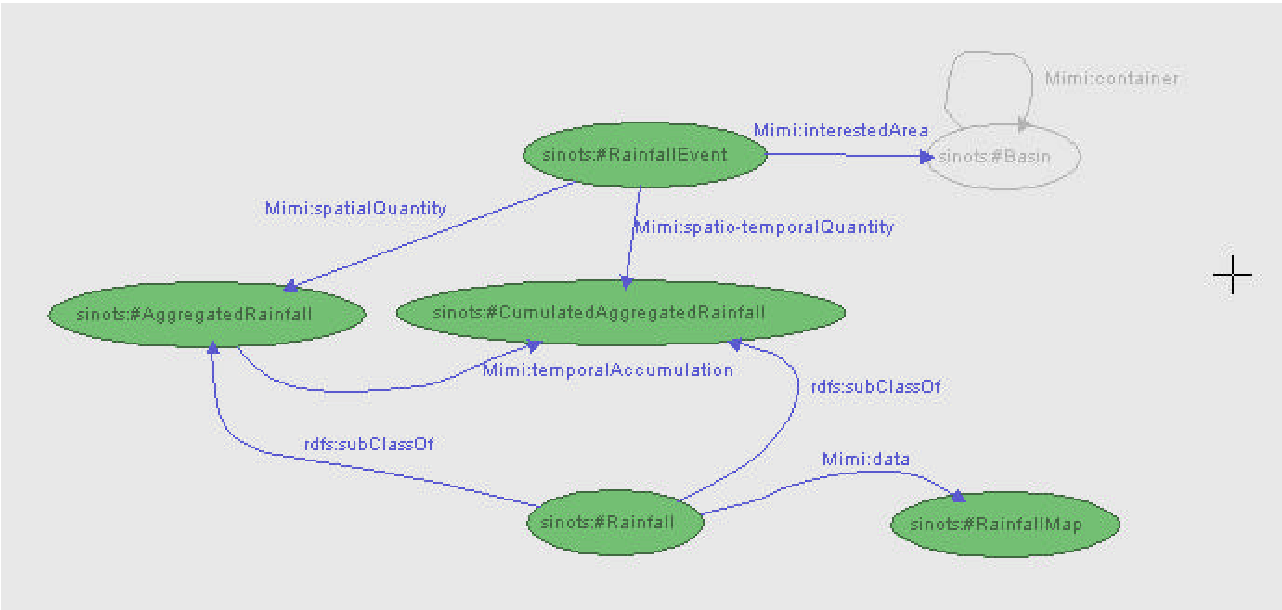

Fig. 5 Ontology diagram – rainfallEvent realm

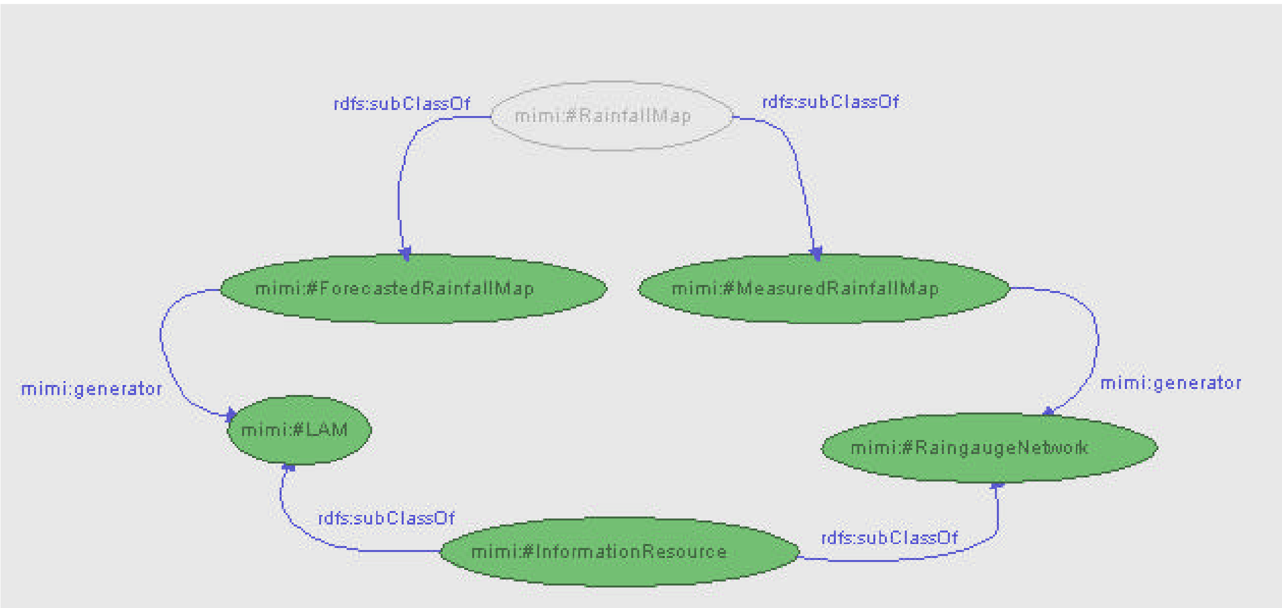

Fig. 6 Ontology diagram – RainfallMap realm

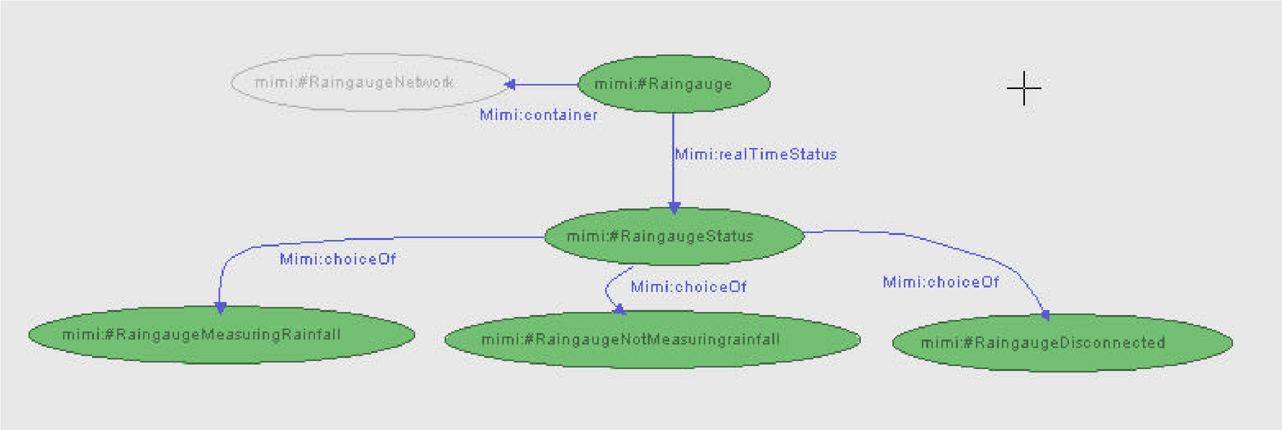

Fig. 7 Ontology diagram – RaingaugeNetwork realm

Making reference to the previous diagrams, a basin is made up of other basin (i.e. sub-basin) and is characterized by a real-time status property which can be (i.e. a choice of):

1. Normal status;
2. Interested by a rainfall event status;
3. Warning status.

A raingauge is characterised by a real time status property which is a choice of the following values:

1. Measuring rainfall;
2. Not measuring rainfall;
3. Disconnected.

The implemented GUI has been designed to represent such ontology in a user friendly manner. The GUI metaphor was taken from GIS tool interfaces.

3. HYDROLOGICAL MODEL

The hydrological model defines the procedure to extract the information contained in the input data and to map it on the user ontology. It acts as an Expert System collecting several data sources and translating them in a different output ontology. However it does not simply maps between input and output ontology. Indeed the hydrological model processes input data to obtain information about parameters that can be easily expressed according to the user ontology. For example, the user ontology includes aggregated rainfall values for a critical section, but the input data do not provide information about them. The hydrological model need to define how it is possible to compute the aggregated values starting from the rainfall maps. This is a hydrological problem and this is the reason the model is named a Hydrological Model.

Following a brief description of input data and hydrological model according to the present scenario is described.

3.1 Early Warning Issue

An effective flood emergency warning service requires a set of hydro-meteorological forecasts characterized by different reliability depending on the forecasting time. In particular quantitative local area meteorological forecasts within 24-48 hours can provide useful data for hydrological scenarios modeling. Also spatial scale has great importance for flood statistical forecasting and warning. For surfaces with an area less than 1000 km² meteorological forecasts of rainfall fields are needed and then the knowledge of rainfall thresholds is required. Each threshold is related to a particular hydrological section and it is represented by a curve on the *(h, t)* plane, where *h* is the sub-basin cumulated rainfall height and *t* is the duration of the precipitation event.

The Department of Hydraulics of the Politecnico di Milano (Milan, Italy) has designed an operative methodology to evaluate the flood warning rainfall threshold on a basin scale. In the course of a project carried out with National Basin Authority for the Arno River, such a methodology has been applied to an experimental estimate of the thresholds for a set of critical sections located on the Arno River Basin in Tuscany, Italy [1].

3.2 Measured Rainfall Data System

A data collection system, named MARTE, is available for the Tuscany raingauge network. It is managed by the National Hydrographic Service. MARTE is able to provide data readings of all network sensors and to collect them in a legacy format file. Update time is 30 minutes, but it can decrease to 15 minutes if a critical event is in progress. For each sensor the cumulated rainfall value is provided along with a code for unavailable data or acquiring errors. Data are encoded in a text format according to a proprietary schema.

3.3 Forecasted Rainfall Data System

Lamma (Regional Laboratory for Applied Meteorology) provides rainfall forecasting maps for the Tuscany Region, resorting to RAMS Local Area Model. Spatial resolution is 4x4 km and the system works out one map per hour within 48 hours starting from the current time. Maps are encoded in a proprietary binary format file provided along with a text file containing metadata.

3.4 Data Fusion Approach

The early warning procedure is implemented exploiting both forecasted and measured data provided by the RAMS Local Area Model and the MARTE sensor network, respectively. Such data are collected in nearly real-time mode by the Arno River Basin Authority's system central database. Since it does not send any message to external components on the arrival of new data and no modification of the central processing is possible, the early warning procedure resorts to polling to detect updates. While the system time response can be negatively affected and concurrent access problems could arise, this approach has the advantage of a loosely coupled architecture implementation with minimal interaction between different components. Besides, it allowed the complete autonomy of the central database system. As a matter of fact the central data storing component and the early warning components were independently designed and developed by different working units.

Rainfall data have different characteristics since RAMS provides precipitation maps, while MARTE produces lists of sensor acquired values (e.g. raingauges values). So a pre-processing step is required to obtain measured maps from the sensor data values, and it is done dynamically assigning to each map cell the precipitation value of the closest active sensor, through a Thiessen polygon-based technique.

The system updates the situation (i.e. the sub-basin status) each time a measured or forecasted map is available. Forecasted maps within 24 hours from the current measured map are saved on a stack. When a new measured map is available, the antecedent forecasted maps are deleted from the stack, and the warning situation is updated. So the system presents the basin situation from the current time to the last forecasting time, based on the last measured map and all the ahead forecasted maps.

The updating is performed evaluating the situation separately for each map according to a physical-statistical algorithm that implements the proper early warning procedure. First of all the detection of rainfall is evaluated on each sub-basin. If the aggregated rainfall is greater than a given threshold – and an event is not already active – then a rainfall event starts; on the contrary, if the event is already active and the rainfall amount, specially aggregated on the sub-basin, is lower than a given threshold for a given time interval, then the active event is considered finished.

For sub-basin interested by active events, the temporal cumulated rainfall is calculated starting from the event begin time. An event shape is obtained through the cumulated rainfall variation, and the AMC at the event begin time, estimated by an aggregated rainfall temporal series. That shape is utilized to choose a correct warning threshold. In the case the cumulated rainfall exceeds the achieved threshold, a warning status is set. Such warning is geo-collocated on a “critical section” which is the most critical point in the sub-basin river network. So a section, in every moment, could be in one of three possible status: event not started, event started without warning, event started with warning.

4. SYSTEM ARCHITECTURE

The worked out system is based on the mediator/facilitator approach [2, 3] where mediators have the task of integrating applications and heterogeneous information sources in such a way that they keep their autonomy, and facilitators give access to information providers.

The actual implementation of the system is based on a four-tier architecture as shown in Fig. 8. This means that there are four distinguished sections hosting computational entities with different objectives. In particular it is possible to separate the following tiers:

- a data-tier that hosts the systems providing input data and information;
- a mediator tier that uniforms data views making them accessible to the hydrological model implementation;
- a business logic tier where the data processing to generate results is performed;
- a client-tier where results are presented;

These tiers are loosely coupled to allow for their separation and implementation on different servers. These means that the interaction between them is made as independent as possible by the particular implementation of each computational entity.

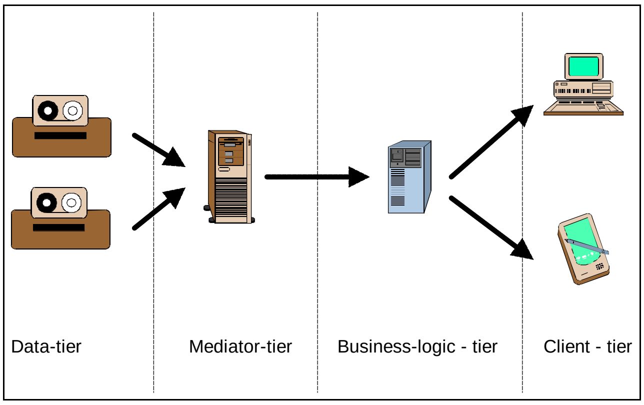

Fig. 8 Four tier architecture

The choice of this four-tier architecture is based on the logical existence of well-defined tiers separated by administrative, technological or logical boundaries. In particular the <u>data-tier</u> is made of all servers providing data useful as an input for the hydrological model. In the present scenario MARTE data and LAM forecasts are provided to the system. The <u>mediator tier</u> is required since the data-tier provides data in legacy format and non-uniform content, so a smart translation process is required, achieving data suitability for the model. The <u>client tier</u> is obviously required to access the system from remote terminals in a Intranet/Internet environment and in the presence of different device terminals.

4.1 Data tier

The data-tier hosts two data servers each providing one input data set. The former is the MARTE data server that stores the information about the raingauge network reads. The latter is the LAMMA data server storing maps produced by a Local Area Model (i.e. RAMS).

4.2 Mediator tier

The mediator- tier is made of a Mediating Server hosting the Rainfall Mediator application that collects raw data from the data tier and transforms them in a format suitable by the hydrological model. Its main functionality is the generation of rainfall maps for the actual situation starting from the raingauge data. This task is performed resorting to the Thiessen algorithm that assign to each map cell, the rainfall value measured by of the nearest sensor.

This task can be considered a high-level (smart) mediating task since it translates the information input format into the output format. However it is not a low-level operation that simply requires a different encoding of the information. A processing is required to present the same information in a different view. Data available as point sensors reads are processed to make them visible as a rainfall map.

The actual implementation of the Rainfall Mediator processes only point sensors measured data, but future implementation will include radar maps data to provide rainfall maps. Moreover in the future scenario the hydrological model will be able to perform a fusion of data different from rainfall rate, including soil humidity and atmospheric parameters. Specific Mediators (e.g. Humidity Mediator) will be developed to generate model input data (e.g. humidity maps) from several different data sources (e.g. satellite data, ground based sensors data).

4.3 *Business logic tier*

The business logic tier hosts the Hydrological Model Server where the Hydro Model application run. It collects data from the Mediator(s) and processes them according to the hydrological model defined by the Politecnico di Milano previously described.

4.4 *Client tier*

The client-tier hosts the devices where the applications devoted to the visualization and presentation of the results run. Moreover the client applications implement the user interface designed for an easy access to the information by the decision-makers.

5. TECHNOLOGICAL CHOICES

The technological choices for the implementation of the previously described architecture are based on several requirements and constraints. In particular:

- *interoperability* between heterogeneous systems differing for operating systems, network capabilities and protocol support;
- *scalability* in terms of number of nodes in particular in the data-tier (increasing number of data servers) and in the client-tier (number of decision-makers and users accessing the system);
- *reliability* of the system that need to correctly work in situation where a fault can be critical;
- *security* of the system that must be accessible only to the authorized personnel according to the policy of the involved public bodies;
- *performance* in terms of processing capabilities and delivering of the results. The system must be able to compute a new result a few minutes after the receiving of new input data to offer nearly real-time situation estimates. Moreover the results should be delivered to the user without great delay even with low-bandwidth connections;
- *integration* with existing network infrastructure. In particular the system should be integrated with the Web Portal of the National Basin Authority of the Arno River;

All these requirements were addressed adopting open standards and multi-platform solutions. In particular in the proposed and implemented solution, the core enabling technologies are Java and XML.

The choice of Java allow to have a multi-platform solution addressing the problem of the heterogeneity of the technological infrastructure. Moreover its strong support for network communication makes easy to develop the communication modules. The only problem with Java platform could be the poor computing performances. However some tests showed that correct implementation of the hydrological model allowed to obtain optimal processing times, providing results in a few seconds after new data were available. The final choice has fallen on Java 2 J2SE platform integrated with specific packages XML and web-services support. The GUI is implemented basing on the freely available OpenMap Java library [4]. It provides basic functions for geographical data navigation such as raster and vector layer management, window pan and zoom. The OpenMap support of ESRI Shape layer format suggested its adoption as the vector format in the client. The library has been widely extended to support other functions like active layer, labeling and so on.

XML was the basis for a neutral encoding of the information. While Data Servers use legacy format for data provision, the Hydro Model Server encodes its results in XML format according to schemas based on the defined user ontology. This allow to separate the content structure from the presentation. Clients can access the XML-encoded information after a translation in a proper presentation language. The translation can be performed by the client or by the server. Anyway a wide range of devices and client applications can be supported simply providing a proper translation module.

XML is also used to transfer other structured information like city locations to help clients to dynamically build some layers. In such a case, once a client is started, it firstly connects to the web server to retrieve the XML files defining layers labels and parameters.

Concerning the communication networks standard solutions are adopted. While the choice of Web technologies and standard (e.g. TCP/IP stack and HTTP/HTTPS application protocol) was obvious, the adoption of SOAP-based Web-services technology was important for increasing of the interoperability.

6. IMPLEMENTATION

A system based on the previously described architecture has been implemented. In particular the implementation required to develop the Rainfall Mediator module, the Hydro Model module, and a Client module. The first two modules were installed on a server connected to the National Basin Authority of Arno River Intranet. Fig. 9 shows the implemented architecture, corresponding to the scenario depicted in the Fig. 2.

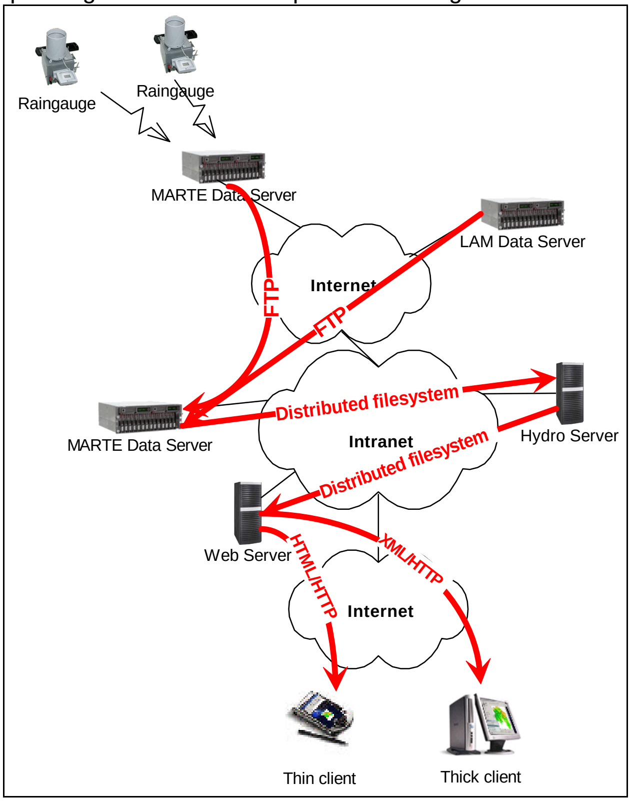

Fig. 9 The system implementation

The Rainfall Mediator module was implemented as a Java application running on a MS Windows NT 4.0 server. The server was then upgraded to MS Windows XP Pro operating system. The module polls for data transferred from the MARTE and LAMMA Data Servers to an internal data server according to legacy protocols. The adoption of the poor-performing polling procedure was due to constraints of internal policies of the National Basin Authority of Arno River Intranet.

The Hydro Model module was implemented as a Java application running on the same server than the Mediator module. It accepts data coming from the Rainfall Mediator module and applies the algorithm defined by the model specification of the Polytechnic of Milan. Its object oriented implementation allow for great scalability in terms of number of processed sections. The Hydro Model module uses a local database (implemented as a MS Access database accessed through JDBC/ODBC technology) to store hydrological and application parameters. The hydrological model server implements the processing of maps to generate warning situation records. The Hydro Model module polls for forecasted and measured maps coming from the Rainfall Mediator and sends situation files in output to a web server. Processing information, such as number and location of sections, threshold values, basin cells weights for cumulated rainfall processing, configuration parameters and so on, are stored in a local database. When a situation file is ready, the old one is stored for archiving purposes. Moreover the hydro server produces a log file reporting all started events, all warning situation and every problem occurred during the processing, giving the administrator useful information about it.

Two different Client Module were implemented. The former is a *thick* client and the latter is a *thin* client based on a browser. The thick client was developed as a Java application and offers much more functionalities. Obviously it also requires much more resources and it is then suitable only by rather powerful stations like desktop and laptops. In such case the integration with the National Basin Authority of Arno River Web Portal is obtained resorting to the Java Webstart technologies that, in the Sun Microsystem view, replaces the old applet technology. The thin client is simply made of a browser accessing pages that are generated dynamically from the Hydro Model XML-encoded results. The XML to HTML translation is obtained resorting to the XSLT (Extensible Stylesheet Language Transformation) technology. The thin client offers less powerful functionalities in respect of the thick one, but it can be used through any station with a web-browser.

7. HUMAN-SYSTEM INTERACTION MODEL

The Java-based thick client is specifically designed for the decision-makers. It offers a Graphic User Interface based on the typical use-cases defined in collaboration with the final users.

The GUI is designed to allow the users to visualize the time variations of spatial warning situation, and also the information about the features in the observed area. Since the great amount of available data, it is very important to carefully select the information that must be showed to the user.

In the adopted human-system interaction model only the sections statuses at a given time are visualized, since it is the most important information. The main window updates to visualize the correspondent situation showing different icons for each different status in the section locations. The user can choose the reference time by choosing the measured rainfall map or one of the available forecasted maps. Moreover he can navigate the map to show in more details a particular area.

To avoid the presentation of old data, each time a new data is available the client window is instantaneously updated without any user’s action.

The other information are shown on specific request. Once the global situation is shown, the user can inspect a specific section by double-clicking its icon to obtain charts of aggregated and cumulated rainfall on it. Measured and forecasted data are shown in different colors.

Labels of all the other features shown on the map are visualized in a specific area of the GUI.

In particular it shows the following main characteristics:

- main thematic layers included (e.g. DEM, rivers, main locations, raingauge locations, critical sections);
- basic map navigation functions (four directions pan, zoom in and zoom out);
- basic thematic layers management functions (show/hide, order change, add/delete);
- situation layer selection;
- section boundaries visualization and locking;
- features labeling (i.e. tips over maps);
- visual and sound alert for early warning situations;
- rainfall chart visualization;

The client operates continuously polling for the current situation file. When the file is updated it is retrieved and locally processed to generate the proper situation layers, one for each measured and forecasted available map. The layers are then added to the layer manager to make possible the user to select them.

8. APPLICATION USER SCENARIO

The typical user scenario is described with the help of Figures 10-13.

a. First of all an user equipped with the thick client connects to the web portal and asks to lauch the MIMI application. Authorized users that are not equipped with the client can utilize a common web browser to download it from the main system web site along with WebStart needed plug-in. The system requires an authentication step to verify user identity (Figure 10).
b. The client is started and it automatically connects to the web server to collect all needed data and then starts the polling procedure to retrieve the updated situation. No other operation is needed, allowing the user to pay attention to the observation of the present and future estimated situations (Figure 11).
c. The user is displayed the last measured situation, he/she can move over the map and read hints coming up when he/she passes over a meaningful feature (e.g. critical section, raingauge, main locations).
   It is possible to switch to the forecast mode: a scrolling panel with all the available forecasts pops up; selecting one of them the displayed map is automatically updated showing the future selected situation (Figure 12);
d. The user can analyze single section situation in much detail (Figure 13). The user, clicking on a critical section icon which is interested by a rainfall event (i.e. triangle shape icon) is presented two rainfall diagrams:

   - the special aggregated rainfall over a sub-basin;
   - the temporal accumulation of the aggregated rainfall over the sub - basin;

   Referring to Figure 13, diagrams blue segments represent rainfall actually measured, while green segments are derived from forecasted rainfall.
   The red line – depicted in the cumulated rainfall diagram – is the dynamically calculated warning threshold. When the cumulate rainfall value crosses such threshold a warning status is trigged on and the critical section icon becomes a red triangle.
   Resorting to this information and integrating it with

other sources, decision-makers can decide to activate emergency procedures. Basic warning communication functionalities (e.g. via telephone, fax, e-mail, etc.) can be easily implemented to automatically signal possible hazardous situation to users

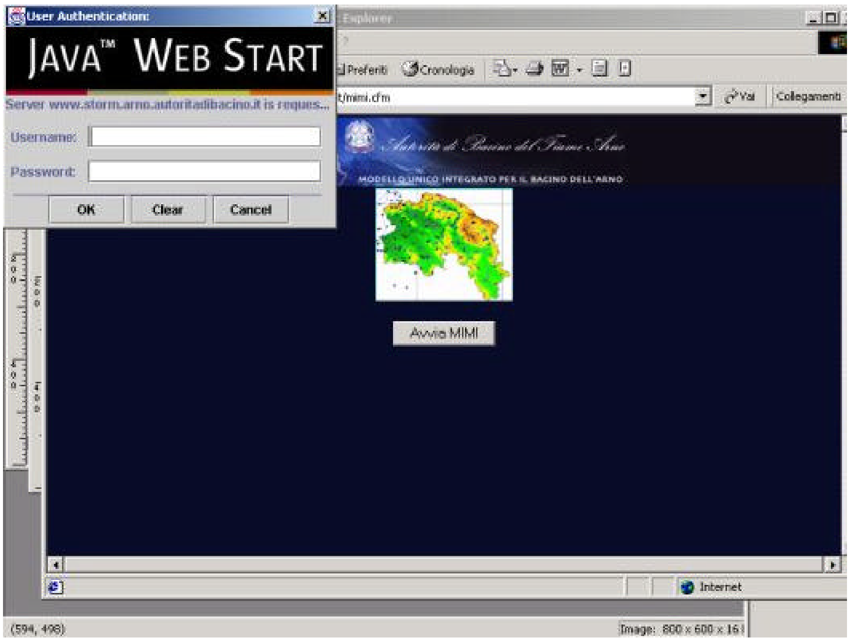

Fig. 10 The login screen

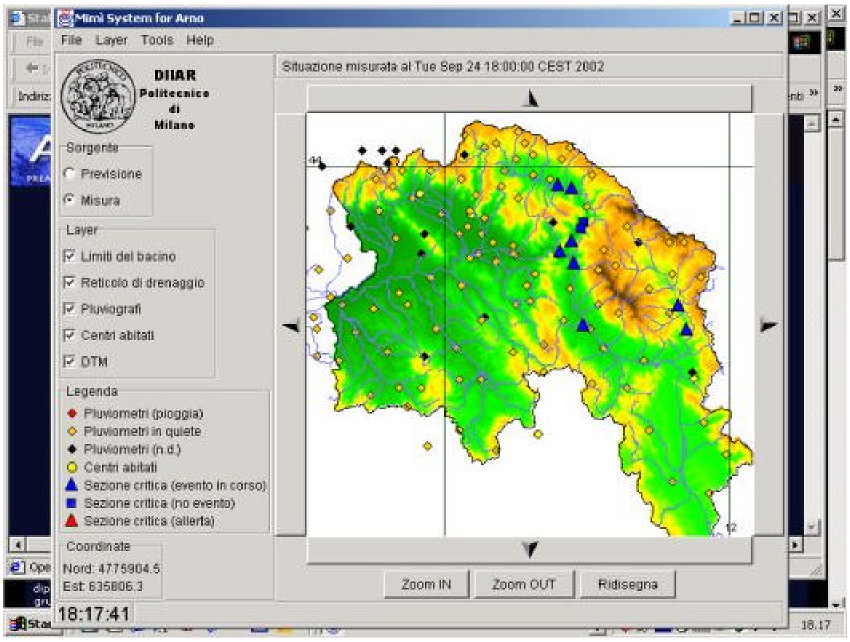

Fig. 11 The client shows the measured situation

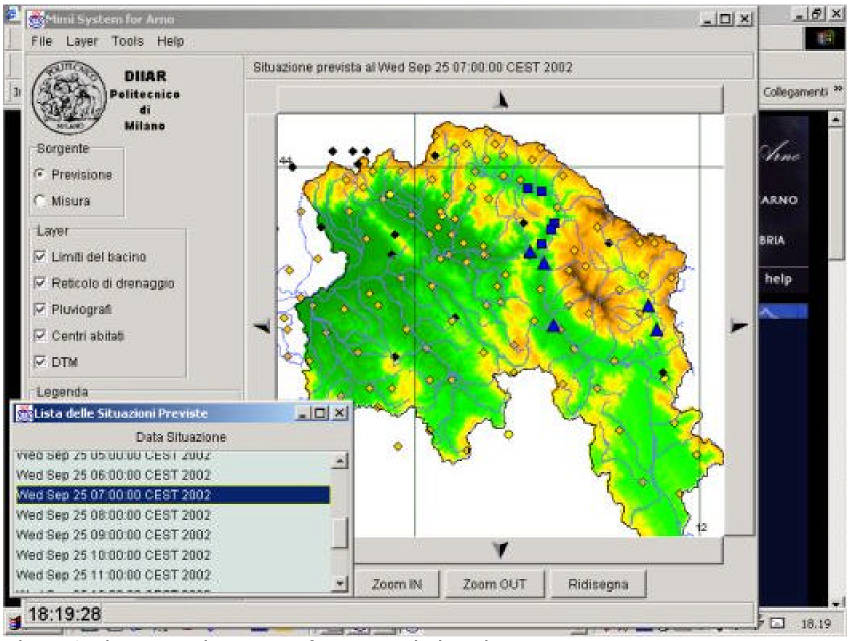

Fig. 12 The user chooses a forecasted situation

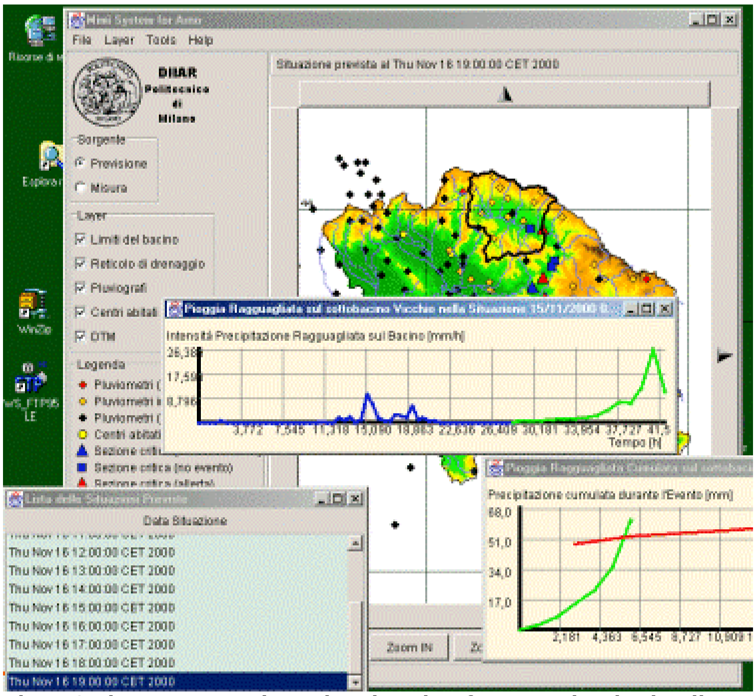

Fig. 13 The user examines the situation for a section in detail

9. CONCLUSIONS

The system described above has been deployed in the operational setting, where it has proven to be reliable and scalable as required, as far as the technological architecture is concerned. However an accurate test phase, comprising the observation of some critical rainfall event, is still on-going; it is useful to tune the several parameters the hydrological model depends on. Such essential work has been doing by the Polytechnic of Milan.

Changes in the global system architecture, particularly for data collection, are expected in the early future (i.e. see the previous future scenario paragraph); their impact on the warning model are expected to be minimal, that is due to the adopted open and extensible architecture. Moreover, it is likely that new emergency management policy will require the extension of the system. The presented solution is well designed to face such extensions.

Presently the system is working in the framework of the operational portal of the National Basin Authority of the Arno River; decision-makers can consult worked out situations in real-time through an Intranet connection.

10. REFERENCES

[1] M. Mancini, N. Montaldo, E. Regalini, R. Rosso: "Valutazione delle Soglie Idrometeorologiche di Preallarme delle Piene Fluviali nel bacino dell'Arno", DIIAR – Politecnico di Milano (2000)

[2] G. Wiederhold, "Mediation to Deal with Heterogeneous Data Sources", in Interoperating Geographic Information Systems, Springer LNCS 1580 (Proc. Interop'99, Zurich, March 1999), pp. 1-16

[3] G. Wiederhold, M. Genesereth, "The Conceptual Basis for Mediation Services", IEEE Expert, Vol. 12, n. 5, pp 38-47, 1997

[4] OpenMap web site, http://www.openmap.org/