WiseObserver: a Real Experience with Wireless Sensor Networks
Juan A. Castillo, Antonio M. Ortiz, Vicente López, Teresa Oliva res, and Luis Orozco-Barbosa
Albacete Research Institute of Informatics
University of Castilla-La Mancha
Albacete, Spain
0034 967599200 Ext.2693
jcastillo, amanuel, vicente, teresa, lorozco@dsi.uclm.es

ABSTRACT
This Wireless Sensor Network Data Visualization Framework tries to be a tool for the visualization and analysis of data collected by a wireless sensor network in a generic scope of application. It also tries to establish a sensor network control interface.

The tool will include several facilities to treat sensor network data. It allows the generation of evolution charts, interpolation maps, evolution data videos, and report generation. It also includes modules to add external data not collected by nodes, but related to the network conditions. Node management will be possible thanks to the execution of commands in network nodes, to perform changes in network operation.

All these functionalities have been implemented taking into account a reasonable performance level. It also guarantees reliability making use of security restrictions. The tool will be easily maintainable and will allow the portability between different hardware platforms, both regarding to the platform in which it will be executed and network node hardware, allowing data management and data treatment of different platforms.

Categories and Subject Descriptors
B.4.1 [Data Communications Devices]: receivers, transmitters, D.0 [Software]. Software development.

General Terms

Keywords
Wireless Sensor Network Visualization Tool, Software Development, Data Analysis,

1. INTRODUCTION
Data collected by sensor networks is usually hard to interpret. The development of a software tool for visualization and analysis of monitoring data obtained by means of a Wireless Sensor Network (WSN) as well as the management of the network will help to understand and improve network operation.

Specialists who interpret data from the network are not used to be computer experts, and they need an easy tool to manage data collected by the network. Although there exist several similar tools in the literature like SpyGlass [1], Surge Network Viewer [2], MoteVIEW [3], or TinyViz [4], they are oriented to network developers and their use is difficult for people not very familiarized with computers. WiseObserver carries out this task, its intuitive and powerful interface allows all kind of users network data treatment and visualization, enabling easy understanding and interpretation of network data.

WiseObserver tool was created under the framework of the Wisevine project [5], but it can be used for any WSN monitoring application which can be catalogued in class 5 in the classification defined by the Networking Working Group Internet Draft [6]. To switch between applications, it is only necessary to configure packet format to specify data to manage.

WiseObserver has been developed in C# .net and uses a database with data coming from the WSN. One important requirement is that the system must be independent of the hardware platform and software of the WSN, as well as independent of the scope of application altogether. In order to obtain these technological independence and use flexibility it is necessary the division of the application in modules to make easy certain functionalities and satisfy certain requirements.

WSN technology independence is going to affect mainly to the management of the WSN, because of the node family to use, software with which nodes are programmed, etc. Some things will be able to be done and some other not, and some commands will be admitted and some other not. On the other hand, use independence affects to how network data is received and what to do before visualizing it. For example, it is necessary to provide the different WSN data packet formats, and the permission to allow different data types (electrical units, international units…). In order to fulfill this requirement, WiseObserver incorporates the Execution Command module, which allows command sending to network nodes and Node Management module that manages the information provided by the network (node type, battery state, localization, node grouping…).

Furthermore, the application includes several modules with different functionality to fulfill all its requirements: Evolution Chart includes different graphic formats to represent data evolution in a certain monitoring period. The tool facilitates graphic generation in different formats: block, evolution, bars, etc. And also statistical functions like maximum, minimum, average, etc. Another important characteristic of this module is that it allows the comparison between graphs. Interpolation Maps
is an special visualization module which generates contour maps, and offers very fast instantaneous data visualization. Evolution Video Generation module generates videos mounted with sequential contour maps to cover a certain period of time. External Data module makes more flexible the scope of application of the tool. It incorporates external data not generated by the WSN but that are related with data collected by the network. These data that cannot be provided by the network must be added manually. The tool uses these data to perform evolution charts and contour maps and can be compared with data coming from the WSN. Report Generation module generates report documents using data, graphics, maps, etc.

WiseObserver can be used with different WSN technologies and different applications. This tool satisfies the current technological requirements, but also will have to deal with future developments. For WiseObserver, the sensor network is a black box which accepts certain commands and provides data. It is necessary that the WSN developer facilitates network characteristics to the tool to be able to interact with it. Information that can be provided is for example packet format, supported data format, accepted commands, etc. This information is provided using a generic interface.

The remainder of the article is organized as follows: Section 2 details evolution charts generation module. In Section 3, Interpolation Maps and Evolution Videos are explained, Section 4 discuss the addition of External Data. Report Generation module is commented in Section 5. The execution of network commands is showed in Section 6. Section 7 exposes Data Comparison interface and finally, Section 8 gives some conclusions and future work.

2. EVOLUTION CHARTS
This module generates graphs to see data evolution from a sensor or a group of sensors. These data can belong to a time slot, an average of each day, the maximum or minimum of each day, a value of each day at the same hour, etc. Charts are fully configurable; it is possible to change some options like colors, chart ranges, chart type, nodes or group of nodes to generate the chart, etc.

Following the purpose of the tool, generated graphics are always marked by an evolution in the time, in such a way that the variation of the same data for a node or several is showed. For this generation, WiseObserver access to the database with the purpose of collecting the necessary data. As well as basic parameters, this tool allows the election among others, date rank for the calculation, data type that we want and nodes that will appear in the graphic. Once finished data collection, simply we will have "to paint" each data in its place and corresponding color, taking care of the data date, data value, and node to that it belongs. The objective is to be able to see the evolution of nodes with respect to others in any data type, as much taken by the sensors as introduced by the users (External Data).

Besides of chart generation, this module also handles chart exportation. This functionality performs chart exportation to an image so that the user can work with it out of the system. The exportation of the chart to other module in the system like the module that makes the generation of reports is also allowed. Format in which evolution graphs is exported is JPG format [7].

Figure 1 shows WiseObserver evolution graph interface. Graphs are generated using network and/or external data and can be compared with other graphs in the same interface.

3. INTERPOLATION MAPS AND EVOLUTION VIDEOS
This functionality creates contour maps from interpolation of data stored in the database. These data can come from the sensor network and/or from external data introduced by users. These maps give a vision space of the data distribution in the sensor network. Maps are created considering a single data type (Temperature, Humidity, etc.) and it is possible to define which data will take part in the visualization.

The sensor network obtains one value from each node in the network and the final goal of WiseObserver is to create a map in that each pixel of the image has a value calculated from these values collected by nodes using an interpolation algorithm. The application creates a color map based in the sensor network, and this image can be over a real map of the sensor network location. This real map can be provided by the user or extracted from other external application of real map generation.

On the basis of the established configuration parameters, and having data taken by sensors (or introduced by user), the system is ready to start map generation process. Once everything is ready, the system locates nodes in the map, and starts an interpolation process considering data coming from the network and data added manually by the user. There are more factors that must be taken into account, for example, the selected scale, and the appropriated colors. When colors of each zone have been calculated, the system performs a single subtraction to create the image and show it by the screen. In addition, it is possible to conduct some additional battles, like exporting the map to different image formats, to visualize map data, to see the options used for it, etc.

Figure 2 shows an example of several maps that can be generated with this application. These maps are, from left to right, a continuous form map, a map with 10 layers and a map with 25 layers. Number of layers, as well as color scale can be fully configured.
Figure 2. Example of maps (Continuous form, 10 layers and 25 layers)

Figure 3 shows WiseObserver interpolation maps interface. As in the Evolution Chart section, maps can be generated using network and/or external data and can be compared with other maps in the main interface.

Figure 3. WiseObserver Interpolation Maps interface.

In addition to the maps, this module also allows the creation of AVI format videos, using several maps in a time range given that show to user the evolution of data over time.

This video is an AVI stream, generate from an interpolation maps sequence. The options about the maps sequence are the same that in the module of Evolution Maps due to videos are made from a group of maps and some additional options like frame rate, compression type to generate the video in this format, size of frames, and path of the final video.

4. EXTERNAL DATA

This module tries to make more flexible the scope of use in the tool. From this module it is possible to introduce external data that WSN does not generate but which are interesting for the use of the tool and which are incorporated in the module of data visualization. For example during monitoring of an intelligent building it is going to be interesting to store the number of people who is at a certain moment in each room, these data cannot be requested by the WSN it is necessary to introduce by hand. These data are used in evolution charts, contour maps... and will be compared with data provided by the WSN.

External data must be associated with respect to one or several nodes to make comparisons with sensor data. For example, in a building monitoring scenario, like the last example, to compare the amount of present people in a room in a determinate moment, it will be necessary to associate “amount of people” data to the nodes present in the room.

WiseObserver was designed for Wisevine project, and because of that, external data module is adapt to data relative to vineyards.

Figure 4 shows an example of external data association; node specification and external data specification are associated by an intermediate table.

The idea is that in the future this module will import data through a standard mechanism, for example XML, or CVS, to configure dynamically the external data module and adapt it to each scope.

5. REPORT GENERATION

This module has been designed to generate automatic reports based on WSN data and allows the inclusion of graphs and maps generated by means of evolution charts and interpolation maps modules. The user will be able to perform report as well as keeping the configuration for later use.

Once configured, reports can be automatically generated using data of any date, and exported to several formats like pdf, rpt, etc. These reports are related with the other modules due to results from other sections of the application can be added to a report previously defined by user.

Figure 5 shows a simple example of generate reports interface, where it can be seen a report that show all data arrived in a concrete day.

6. COMMAND EXECUTION

This network management module allows sending and reception of commands to manage the network, check battery state, reconfiguration of network routes, request data to some nodes, etc.
To send management commands to the WSN it is necessary to establish previously, by means of a descriptive XML file, which commands and functions are accepted by the WSN, there will be differences depending on the technology and the software that the developer of the WSN defines.

To perform this command sending, Base Station send special packets to some node or group of nodes that will interpret them like a command. Note that all desired functionality must be previously implemented in the nodes.

This module is very interesting, because it allows the modification of the network behavior well requesting additional data and/or changing data routes and node roles.

7. DATA FINAL COMPARISON

In its main interface, WiseObserver allows the comparison of generated results like contour maps, evolution charts, etc. After painting evolution charts, contour maps, and/or generate evolution videos and reports, it is possible to put them together in the screen to compare data coming from different nodes, different periods of time, etc. Figure 6 shows a comparison between two evolution charts and two contour maps. It would be also possible include in this screen reports to compare data with maps and charts.

WiseObserver includes several modules with different functionality to manage network data and generate evolution charts, interpolation maps, evolution videos, and reports. It also includes a module to add external data related to network data, and a special module to send commands to nodes in the network which allows changing network functionality.

The ease to use of WiseObserver and its potential makes it ideal for network developers as well as for specialists not familiarized with computers.

Interpolation maps are designed to work with all kinds of outdoor monitoring applications, but future versions of the tool will include support to draw walls, and calculations for interpolation maps will be performed taking into account these data. With this additional functionality, WiseObserver will also work with indoor applications, extending its scope of use to almost all monitoring wireless sensor network applications.

WiseObserver will be re-implemented in Java. This improvement will make it able to be executed in almost every platform. The inclusion of additional image and video formats will also enhance the power of WiseObserver.

9. ACKNOWLEDGMENTS

This work has been jointly supported by the Spanish MEC and European Commission FEDER funds under grants Consolider Ingenio-2010 CSD2006-00046, TIN2006-15516-C04-02.

10. REFERENCES