MAPISA, a web based framework for structural health monitoring

Michela Basili, University of Rome “La Sapienza”, Rome, Italy
Berta Buttarazzi, University of Rome “Tor Vergata”, Rome, Italy

ABSTRACT. The paper describes a Multi Plug-In Software Architecture, MAPISA, designed mainly to enable users and researchers involved with sensor network data to manage relevant information, thereby improving process interoperability, composition and heterogeneity. MAPISA can easily be extended to support new types of sensor data sources in such a way that is completely transparent for end users.

MAPISA functional characteristics enable professional users to perform various specific tasks on a given sensor dataset, such as mathematical model uploading to perform simulations on different, dynamically selected datasets and data visualization. Even non-expert users can utilize applications based on MAPISA, since it stores all significant data collected from monitored sites which can easily be read and visualized in summarizing reports.

The entire framework has been designed by applying the fundamental principles of software engineering; in particular, modularity and interoperability concepts have been given a fundamental role in order to facilitate the implementation of new features into the platform. In particular, MAPISA is ideal as a base framework for web 2.0 applications. Therefore MAPISA functions as a middleware that enables stored data to be exchanged between potential users and fully-deployed sensor networks so that relevant information can easily be accessed. In order to demonstrate the advantages of this framework, this study presents the development of a web application together with a plug-in and also the integration of a mathematical model. The latter application will also show the output graphs resulting from the running of the mathematical model.

KEYWORDS: Applications, Modeling and simulation, Sensor heterogeneity, Software architecture, Structural health monitoring

Sensor networks are mainly used for Cultural Heritage and Structural Health Monitoring purposes. Thanks to wireless hardware, there has recently been an increase in the use of these networks. Sites that were previously inaccessible to
monitoring processes can now be studied more easily by deploying wireless sensor networks. Different deployment methods are used due to the various types of sensors, but they do not take into account the possibility of software reusability. Each and every sensor network is strictly designed around the subject of study and must be correctly deployed in order to yield the expected results.

Once the network is appropriately deployed, sensors will start to collect data that will be stored in order to be analyzed, studied and processed by ad hoc software applications. Experts must often take a direct approach: reading raw, unprocessed data to extract useful information. These results can then be used to adjust sensors or model parameters.

Recently, this kind of deployment is increasingly accompanied by software applications that can automatically analyze sensor data and take action. In fact Chintalapudi and colleagues (2006) describe a sensor network system, called NetSHM, that enables structural engineers to program Structural Health Monitoring applications in Matlab or C at a high level of abstraction. This method shows how it is possible to build software that can aid experts in their operations.

Moreover Chintalapudi and colleagues (2006) and Lynch and Loh (2006) clearly illustrate how various types of sensor networks can differ greatly, according to the type of sensors deployed and the different types of data collected. During the whole design process of the MAPISA framework, great consideration was given to the heterogeneity of sensors and data.

In fact our primary goal was to define a storage model that could combine data from various sensors in a single format, thus enabling all users to quickly and easily access the status of a monitored site.

The next sections will be structured as follows: section two will describe the main features of MAPISA and the requisites it meets with particular attention to sensor data modeling, the basis for the framework. Section three will present W-MAPISA: a web application built on top of MAPISA for visualizing sensor data and performing other basic operations on sites and sensors.

Finally, a test case will be described to demonstrate the integration of a mathematical model.
MAPISA features

MAPISA is a software framework based on the J2EE (Java 2 Platform, Enterprise Edition) technology developed to offer the following main features:

- **Interoperability**: the system and the user can interact via a unique interface and connect together by means of other sensors still not installed, without any restriction for the access or the implementations.
- **Data independency**: in order to obtain data independency a data model based on XML (Extensible Markup Language) is used.
- **Transparency**: MAPISA can operate transparently with various Sensor-Data sources by offering the same set of operations to users.
- **Modularity**: the software architecture revolves around a compact core that acts as a dispatcher to fulfill requests and various external modules that are loaded at run-time only when needed.
- **Expandability**: new modules can be easily developed and registered into the framework for immediate use.

In the following sections the main functions of MAPISA are described.

Sensor-data modeling

An XML-based model has been developed to enable large quantities of data to be summarized in a compact model. The main idea behind the model is that only “interesting” time spans are recorded: an XML formatted information set can be easily read by a dedicated software to retrieve a summary of all the relevant activities registered by a specific sensor. Deploying a particular sensor to monitor a specific site usually requires various studies by experts that will involve, at a certain stage, the definition of one or more thresholds that will categorize a certain sensor dataset as “critical at a certain level” or “non-critical”. In other words, the experts examining raw data will know exactly what kind of data to expect from a certain sensor in order to classify them. Figure 1 shows an example of the data modeling process of a sensor that can subsequently be used to build our XML resources.
Moreover, while the monitoring process is operational, previously chosen thresholds can be adapted and/or completely changed in order to respond to particular modifications involving the monitored site. The solution we offer organizes all these pieces of information in such a way that relevant time spans will be summarized throughout the sensor monitoring process.

**Sensor information registration**

MAPISA Core allows entities such as Sites and Sensors to be managed using classic CRUD (Create-Read-Update-Destroy) operations.

*Sensor management*: experts can register a new sensor into MAPISA by specifying its type, location and thresholds. Once inserted, all these attributes can be modified at run-time to reflect dynamic changes in a particular sensor.

*Site management*: users can register a new site into MAPISA by specifying a description and its location. Additionally, other information can be added when modifying a site.
Critical events handling

The MAPISA Critical Events Handling system enables notification to be sent when a sensor-data set is considered to be critical. Applications can be developed to select the registered sensors that need to be monitored in a “pseudo” real-time mode: a process can trigger sensor-data analysis at regular time intervals, in order to monitor a particular sensor. Currently, MAPISA supports only e-mail notifications to users, but more action types can easily be developed.

Sensor data management

MAPISA offers a simple and well-defined interface to request data from external sources. This operational model is based on the “Sensor-Data” concept, which has the following attributes:

- **Accessible**: a set of sensor-data must be remotely accessible by MAPISA over the internet. Data sources can be plain files, databases, dynamic content retrieving methods or other methods.
- **Localizable**: a single item of sensor-data must be both temporally and geographically localizable, in order to be correctly accessed and visualized.
- **Displayable**: at least one method of data visualization must be available. For example, this can be numerical, descriptive or a more complex visualization can be developed.
- **Comparable**: sensor-data must establish one mathematical ordering system so that values can be compared.

In order to enable users to correctly access and visualize sensor data, researchers can develop plug-ins that will be registered into MAPISA.

Dynamic simulation support

Researchers often gather critical sensor data that can subsequently be used as input for the testing and simulation of the mathematical models they have developed. MAPISA allows users to submit a mathematical model (in the form of any executable program, script or routine they have) to be subsequently used with a subset of all sensors registered inside its framework database.
This feature enables different researchers to share each other their models, thus creating what can be called “dynamic cloud computing”: different users enrich the framework with applications that other can access. To be correctly registered into the framework, a script must comply with some specific requirements:

- **Prerequisites**: a routine must clearly define the set of input data which will be used during the process.
- **Output data**: the routine must define the set of output files and the visualization mode in which they will be presented to the user.

If any of these requirements cannot be met, then the user must develop a full MAPISA plug-in. Currently, the only supported types of routine are Matlab scripts (whose main structure can be seen in Listing 1) that must be analyzed by MAPISA administrators. As future development, a syntax where the direct upload of the programs can be easily done by the users which utilize MAPISA data is studied.

```
%% START SCRIPT %%
input1 = load('data1.dat');  %first data set load…
input2 = load('data2.dat');  %second data set load…
% other data loading…

% ...main calculations, on input data e results generation (out1, out2, …)

save('output1.dat', 'out1', '-ASCII');  %first data set save…
save('output2.dat', 'out2', '-ASCII');  % second data set save…
% other data saving…
% END SCRIPT %%
```

### W-MAPISA

W-MAPISA is a web application built on top of the MAPISA framework which uses almost all of the framework features to
manage entities and display sensor-data to users. Two types of users can access this service: non-experts, general users who can interact with it to receive summaries and reports about any unusual situation, and experts who can request raw data, visualize graphs and perform simulations.

As shown in Figure 3, W-MAPISA communicates with the MAPISA framework by means of its Java Bean components. MAPISA main components can be triggered to perform the following operations:

- **Request analysis**: initializes the necessary information on the basis of the User request.
- **Sensor data access**: loads the correct plug-in on the basis of User request.
- **Sensor data visualization**: loads the plug-in that will set up the correct runtime in order to run all the operations needed to display the results (formatted using XHTML) back to the user. This operation may trigger a Matlab script, a “raw-data” file download or some graphical construction, depending on what sensor or site the user selected and which operation he chooses from those listed by the system.

---

**Test case**

This section illustrates an application representing a test case based on a mathematical model developed for structural health monitoring. The Matlab routine which describes the model and the analysis carried out is adapted to be registered into MAPISA and used to trigger simulations based on dynamical inputs. The aim is to show how external models can be implemented and utilized.
through MAPISA, utilizing information (sensor measures) available in the framework.

**Numerical model and dynamic analysis results**

The model here presented is representative of a four story tower. Given some external input, we are interested to monitor the model structural response in order to observe the health status of the building. The dynamical behavior of the tower, thought built regular in plant and in high, has been modeled in the plane with a four degrees-of-freedom (DOF) structure, in scale 1:5, Figure 3.

The example is used to simulate the behaviour of a real building subjected to base strong motion. The dynamical characteristics of the tower are synthesized in its mass and stiffness matrices:

\[
M = \begin{pmatrix}
132.34 & 0 & 0 & 0 \\
0 & 132.34 & 0 & 0 \\
0 & 0 & 132.34 & 0 \\
0 & 0 & 0 & 123.36
\end{pmatrix} \text{ kg}
\]

\[
K = \begin{pmatrix}
6.17 & -3.25 & 0.17 & 0.04 \\
-3.25 & 132.34 & -4.77 & 0.26 \\
0.17 & -4.77 & 8.14 & -3.54 \\
0.04 & 0.26 & -3.54 & 123.36
\end{pmatrix} \text{ kg}
\]

And a structural damping factor of 2% compared to the critical
value is assumed for each mode.
A dynamic input, imagined available from the MAPISA framework, is utilized as base excitation. This represents the registration of El Centro (1940) natural earthquake scaled with a PGA value of 0.3 g. The time is scaled of the factor $1/(5)^{0.5}$, in order to take into account that the model is in 1:5 scale.
The dynamic analysis of the system is carried out and the main relevant responses are estimated as output measures. In particular the following quantities are:

- $d_i$ for $i = 1\text{-}4$ relative displacement at each level measured in mm,
- $a_i$ for $i = 1\text{-}4$ absolute acceleration at each level measured in m/s$^2$.

Such data can be used in real time from remote positions, to make control decisions, such as sending alert signals, and to make other control and monitoring operations. As a result of the Matlab application, Figures 4, 5 and 6 illustrate the graphs generated which show the overall kinetic energy, relative displacement and absolute acceleration at each floor respectively, carried out by the dynamic analysis.
This simple application demonstrated how models created by several users can be shared in the MAPISA framework to make simulation and predict structural response, once known, for example, base excitation measured in the surrounding environment where the building is located or in other interest sites. The simulation allows to monitor the health status of the building subjected to a given excitation.
An other possible application can be thought if the real structure is equipped with sensors which measure structural response and external excitation and data are collected into MAPISA. Such data can be utilized by means of identification techniques to know the structural dynamic characteristics. System dynamic parameters, such as frequencies and damping factors, can change in relation with plasticization and damn occurred into the structure. The new results can be therefore utilized in a model updating procedure.
Conclusion

In this study a framework MAPISA, which allows to quickly develop web applications similar to W-MAPISA, has been developed. It is possible to guarantee to users the access and visualization of data from sensors conveniently located in interest sites. The most important aspect of MAPISA based software applications is that they can be used to keep track of various, heterogeneous sensor data sources and they offer different features for each of the supported data sources. These features fully respect user needs and can be applied to other similar data sources.

Many application scenarios can successfully take advantage of the MAPISA framework: one of the most important involves risk management tasks in the Structural Health Monitoring of Cultural Heritages. Many different sensor networks can be deployed on different settings, each of which can then be monitored on-line by a web application based on an MAPISA distribution that has been expanded with different plug-ins. Each plug-in allows users to manage all of the different sensor-data sources and offers all the different visualization methods that users require.

An example of application of the framework for heath monitoring of a tower has been proposed. Considering the structure subjected to a base excitation, available from the sensors measured in interest sites through MAPISA framework, dynamic analyses are conducted on a mathematical model developed ad hoc and structural response is predicted. Results of the simulations can be easily shared by several users and measures to protect the structure or actions to be taken in case of risk can be discussed.

Further remarks

MAPISA is at an early stage of development, so many features can undoubtedly be improved and more can be added to the current framework, such as support for new plug-in types other than those for Data Access and Data Visualization, mobile devices adaptation and real-time access to data sources.
Sintesi

Negli ultimi tempi grazie alla miniaturizzazione dei sensori elettronici si sta sempre più diffondendo l’utilizzo di Wireless Sensor Networks (WSN) in vari settori dell’Ingegneria, ad esempio per il monitoraggio di parametri ambientali (temperatura, umidità, livelli idrologici) o per il controllo dell’inquinamento (acustico, da smog, etc.). In particolare nel campo dell’Ingegneria Civile, le applicazioni di maggior interesse riguardano il monitoraggio dello stato di salute (Structural Health Monitoring, SHM) di ponti, edifici e, in genere, di strutture soggette a vibrazioni di diversa natura (vento, traffico, terremoto).

Nonostante l’ampia diffusione di questa tipologia di reti, ancora oggi non esistono standard ed i problemi da affrontare sia nell’installazione (sincronizzazione dei sensori) sia nella visualizzazione dei dati rilevati sono molteplici. In questo contesto sono auspicabili sistemi software che permettano attraverso applicazioni avanzate (che utilizzano le tecniche del web 2.0, mashup programming, etc.), la fruizione su larga scala dei dati rilevati dai sensori rendendo possibile anche la gestione in tempo reale di fenomeni disastrosi.

Con tale obiettivo è stato progettato MAPISA (a Multi Plug-In Software Architecture), un framework di supporto allo sviluppo di applicazioni web 2.0 per la gestione di dati provenienti da WSN.

In particolare in questo articolo viene descritta un’applicazione web realizzata...
che consente:
- il monitoraggio dei dati provenienti dai sensori rappresentati su una mappa;
- l’analisi dei dati rilevati in una particolare finestra temporale attraverso la visualizzazione di grafici;
- l’impostazione sulla mappa di nuovi sensori e dei corrispondenti valori di soglia per definire criteri di allarme;
mettendo in risalto come l’architettura multi-plug-in su cui è basato il framework consente di supportare, in modo trasparente agli utenti, anche nuove tipologie di sorgenti di dati.
Per dimostrare i vantaggi derivanti dall’uso del framework nel favorire l’implementazione di nuove funzionalità che agevolino l’interazione fra diverse comunità scientifiche (ingegneri, architetti, progettisti), è descritto l’inserimento di un plug-in che consente l’upload di un modello matematico (realizzato tramite uno script Matlab) e la sua utilizzazione per effettuare simulazioni a partire da differenti dati di input provenienti dai sensori.