

```
=====
===== XML Information Representation Tool - XIR 1.0 =====
===== University of Puerto Rico - Mayaguez Campus =====
===== Department of Electrical and Computer Engineering =====
===== Luz V. Acaba-cuevas =====
===== Luz.Acaba@ece.uprm.edu =====
===== http://ece.uprm.edu/walsaip =====
===== March 2007 =====
=====
```

XML Information Representation (XIR) is a java-based tool for coupling/binding representation of data and metadata entities associated with physical sensors pertaining to environmental surveillance monitoring applications.

## IMPORTANT

This demo version provides limited functionality. In order to execute an encapsulation of a data and a metadata file, the files need to follow an specific format and parameters. For that reason, data and metadata files are provided in files directory. The "save" button on encapsulation window is not working. The file should be selected from the text area, and paste in a new text document and saved with the file extension ".xml"

## User's Guide

The following set of instruction guides the user through the process of encapsulation of a data and metadata files obtained as part of the test bed established by the Automated Information Processing Group of WALSAIP project. To obtain more information about WALSAIP please visit <http://ece.uprm.edu/walsaip>.

### Step 1: Load Data and Metadata files

- 1- Run XIR\_Demo.jar file.
- 2- Select "Encapsulate" menu item. The window Encapsulation should be visible.
- 3- Select "Browse" button under "Load Data File" text area to browse the data file provided for encapsulation.
- 4 - Use the browse window to search for "files" directory. This directory is located in the same directory as the XIR\_Demo.jar file.

5- Select "data.txt" file on "files directory" and click "open" on browse window. The content of the data file should be displayed on text area.

6- Select "Browse" button under "Load Metadata File" text area to browse the metadata file provided for encapsulation.

7- Use the browse window to search for "files" directory. This directory is located in the same directory as the XIR\_Demo.jar file.

8- Select "metadata.txt" file on "files directory" and click "open" on browse window.

Take few seconds to analyze both files: data and metadata. As you can see on "Load Data File" and "Load Metadata Files" after selecting the corresponding files, there is no meaning easy to perceive from these values. It is difficult to understand the significance of each piece of information. In some cases the format of the parameter can suggest meaning. For example, the format of the first parameter of the metadata file looks like a date. However, the second parameter also seems like a timestamp. What is the difference between those two parameters? XIR Tool provides answer to this and other question pertaining the meaning of each piece of information. Encapsulation function combines both files together adding XML Tags to enhance the description of each piece of information, add interoperability

## Step 2: Encapsulate the files

1- Select "Encapsulate" button. The encapsulated file should be visible.

After encapsulation, analyze the encapsulated file displayed on "Data-Metadata Encapsulated File". The first thing to notice is that at the beginning of the file, there is information that was not present on either data or metadata files. This is achieved by an enhanced function (not enabled in this demo) that allows the user to define a standard heading for each file including source and contact information for backtracking purposes. After the heading, each parameter of metadata file is displayed with XML tags. The XML tags serves as "labels" that describes the parameter. For example, the third parameter on metadata file "0" is displayed on encapsulated file as <nodeID>0</nodeID>. nodeID indicates that 0 is the node from which that data were collected. Other XML advantages are beyond the scope of this demo.

A wizard enables the user to generate XML tags in an abstract way. The user does not require XML knowledge to generate XML tags for each parameter on files. This feature is under construction.