

# Configuring Applications based in XdefML

*José Antonio González Pastrana  
Airbus Defence and Space, Getafe-Spain,  
Jose.o.gonzalez@airbus.com*

## Abstract:

Developing applications to configure instrumentation is a never ending task as new equipment is constantly coming up in the market. Embedding this new instrumentation to existing software can be tricky if you have to deal with every single characteristic of the equipment. XdefML files facilitate adapt these new instrumentations in your internal developed configurations tools.

**Key words:** MDL [1] , XML [2], Open Standards, DAS[3], XidML [4], XdefML[5].

## Introduction

A XdefML file is a kind of XML file that contains all possible values that an instrument can have. It is an open standard, so everybody can use it freely. As any other XML file, it is based in its own schema. This schema represents all possible fields that a XdefML file can support.

The aim of a XdefML file is to have available all possible settings, parameters, channels, etc, that an instrument can have to create another file, which represents the real configuration of the instrument. This file that contains the configuration of the instrument can be written in different ways, such as XML or text files.

## What is XdefML?

XdefML is a metamodel that permits us to define the features supported by a card or equipment. We need it because configuration files tell us which the configuration of an equipment is. But in its schema there isn't a way to define what characteristic have to be set or which values are permitted for certain equipment/card.

So, for writing a XidML, MDL or any other kind of configuration file we need previously to know which are the features that an equipment have. A XdefML file give us the features and the values valid for such an equipment.

## XdefML files characteristics

There are different standards to define configurations for data acquisition systems as it can be XidML and MDL, also there are vendor dependent files too. Any of them need a file to describe which are the settings, parameters,

specifications, etc, necessary for define a card or equipment configuration. Usually these kind of files are not supplied because it is intended that the configuration software supplied with the equipment is in charge of this. If you have to create configuration files by yourself, beforehand you need to know which the admitted values are for each one of the named features. This is why you need XdefML, to create your own configuration files, instead of using the vendor's tools. If you are dealing with big data acquisition systems or you need to manage a great quantity of projects, you'd better use an automatic system to create the configuration files.

XdefML Schema have three main nodes 'InstrumentIdentification', 'InstrumentSpecifications' and finally 'InstrumentConfiguration'.

So, in any XdefML file all these three will be shown. The necessary information to identify the instrument as Manufacturer, PartReference, Serial Number, etc. is displayed in the 'InstrumentIdentification' node.

Under the 'InstrumentSpecifications' node, mainly information about mechanical characteristics and related documentation as its datasheet is shown.

Under the 'InstrumentConfiguration' node, the information that contains all possible channels, settings and parameters characteristics is displayed. It must be taken into account that some of the characteristics are recursive. For example, an instrument can contain another instrument and this one another one, and so on. A parameter can have subparameters, settings can have subsettings, etc. This recursion makes to deal with XdefML schema even more

difficult. Recursive schemas can't be imported with the libraries that can be used in most languages as C# and similar ones.

So, there is a SDK [6] that helps you to deal with XdefML files. This SDK is available from CW-Controls (former ACRAControl).

### Working with XidML SDK

XidML SDK contains libraries that help you to work with XidML and XdefML files. In this case our main concern is with XdefML related library. This library can load a XdefML file and navigate through it to extract all necessary information in order to fill-in data in your database or to generate XidML files, or any other kind of configuration file you may need to use.

A function of this SDK can load a XdefML file and from it generate a new XidML file, filling all its parameters with default values. This configuration file (XidML) can only be used by one vendor configurations. When you have this XidML file in your computer memory you can change default values for other ones you may prefer.

However, any developer can take all the features of a card/equipment from a XdefML file. Having them in the computer memory and then create a configuration file for any other vendor.

Obviously this way is harder than using the function already in the SDK, anyway you can always build your own function which outputs the file you need. To build your own classes/functions there is a library called LINQ (Language Integrated Query) from Microsoft that helps you to query xml files.

### Tools based in XdefML

Two tools have been developed based in XdefML files and on the library XidML SDK. These are the DAsp and DAC\_TOOL3.0 tools, another one is under construction and several other will be developed in the near future.

DAsp is a tool that generates configurations for any kind of acquisition systems, based on plugins that get the input as xml files which have been obtained, in our case, from our database.

DAsp extracts all necessary data for cards/equipment from XdefML files, it processes all this information and writes the output in the way that is needed. The first plugin creates configurations for XidML 3.0 files. These files are compiled to check if they are correct, to be loaded on the equipment.

DAC\_TOOL3.0 is a tool to save the card's configuration in our database. It gets a XdefML file from a card/equipment and present all its settings and parameters. This way the engineer can change the default values for any other useful for certain application. There is another tool under construction (FTIForm3.0) to check that instrumentation information corresponds with the one of the DAS [8]. Fairly, the information of the aircraft's parameters is displayed in an easy way, to check where exactly a parameter is connected to the DAS.

The ranges of the card's channels and the calibration can be correctly set. It queries the values from the database, if they are null it takes the default values from XdefML file that corresponds to the card.

In a near future some other tools will be developed. The main one is a suite of tools to manage all instrumentation processes, from requirements coming from the customer to DAS configurations, sensors calibration and storage. All this will turn up to be an integrated tool called FIDA. FIDA will use XdefML files to get all the information necessary to know which the features of equipment and the admitted values for them are. It will be a graphical tool that will contain objects linked to each XdefML file. Therefore for including a new acquisition system the only thing we will need to do, it'll be to copy the new XdefML files to the computer directory where all the others are stored.

### Conclusions

XdefML files are going to be a key technology for instrumentation and DASes in ADS [7]. These kind of files are suitable to define the characteristics for any type of acquisition system, no need to worry about the vendor, as they are not vendor dependent and are an open standard.

However, having these types of files supplied by the vendor is an advantage. As XdefML files are vendor agnostic all software needed to create configurations for different kind of acquisition systems is almost the same for all of them.

Therefore most of the software developed internally to create data acquisition systems configurations is the same for all vendors.

These characteristics make updating the tools to support new acquisition systems faster and easier.

## References

- [1] MDL - <http://www.irig106.org/docs/106-17/Chapter23.pdf>
- [2] XML - <https://www.w3.org/XML>
- [3] DAS – Data Acquisition System.
- [4] XidML - <http://www.xidml.org>
- [5] XdefML - <http://www.xidml.org>
- [6] SDK – Software Development Kit
- [7] ADS – Airbus Defense and Space
- [8] DAS – Data Acquisition System