

The open XidML Standard - What are the Advantages for Users

Florian Mertl

Airbus Helicopters Deutschland GmbH, Industriestrasse 4, 86609 Donauwörth
Florian.mertl@airbus.com

Abstract:

XidML is the “eXtensible Instrumentation Definition Markup Language”. It is an XML-based, open and vendor-neutral standard to describe any type of FTI (Flight Test Instrumentation) configuration. XidML aims to store and exchange all information about how data are acquired, processed and packaged for transmission.

XdefML is part of the XidML standard. As for XidML, the XdefML schema is open, vendor-neutral. It is designed to describe all properties, specifications and parameter of an equipment. It is often called “electronic datasheet” and so it does e. g. not describe the actual value of a specific setting, but the limits of it.

XdefML can be adopted solely, it does not rely on XidML. User can take benefit of just implementing XdefML in their toolchain to having a standardized interface to describe FTI equipment.

The available schemata for XidML and XdefML allows the creation, parsing, verification and validation of XidML and XdefML files.

Key words: Open Source, vendor-neutral, electronic datasheet, markup language, data management standard

What is XidML?

XidML was born in 2004. At that time, it was mainly driven by single user/supplier in the field of flight test instrumentation. The idea was to find a way to describe, how FTI systems store and exchange information. The solution was found by developing a derivate of the XML eXtensible Markup Language. XML is both human- and machine readable. The structure of an XML file is based on tags (like e. g. HTML). One key element of XML is that XML files can be validated by applying a schema. A schema is a separate file which defines the legal structure of an XML file. Tools can validate XML to ensure the data integrity and that the file is well-formed (valid tags, order, data types etc.). In 2004, the XML schema was adapted to the needs for flight test instrumentation and this extension was named as XidML, where “id” stands for “Instrumentation Definition”. To ensure availability, flexibility, a wide acceptance and to foster collaboration, all information should be made free and open available. The initial focus was mainly the description of data acquisition systems, although the adoption to telemetry or recorder systems was already considered by the XML-based utilization.

Over the years, XidML evolved from version 2 to the current version 3

According to [1], XidML files consists of up to 8 main elements:

- **Auxiliary Files:** This element lists all the auxiliary files associated with the XidML document
- **Documentation:** This element can be used to document a XidML file. It can include information such as short and long descriptions, the creation date, who the author of the XidML file is, references to technical specifications, and other miscellaneous information.
- **Parameters:** This element contains a list of all globally defined parameters and their properties sourced or generated by the acquisition system(s) described in the XidML document.
- **Packages:** This element contains a list of all the packages used by acquisition system(s) described by the XidML file.

- **Algorithms:** This element contains all the algorithms used by the acquisition system(s) described in a XidML file.
- **Instrumentation:** This element lists all the instrumentation used by an acquisition system along with associated setup information.
- **Links:** This element contains a list of all the Links used by acquisition system(s) described by the XidML file.
- **Addenda:** This element contains a list of "Addenda" that can be used to place any information that is not catered for in XidML.

XdefML was introduced in parallel to XidML

As the name already implies, XML allows to extend the

Where does XidML & XdefML affect FTI

XidML and XdefML can have an impact in different areas in the daily FTI life, like:

- **FTI Configuration management:** When looking for new FTI devices, systems or sensors (instruments), there are multiple requirements which must be fulfilled by the candidate. Beside the technical specifications and properties, it is also important, how to integrate this new instrument in the existing toolchain. Every user needs to find a way to link tests to instruments to sensors to calibration information to movements and so on and on. This toolchain can reach from a simple spreadsheet approach to highly complex and customized databased framework solutions. Bringing new hardware in the loop should be easy and efficient. Due to the standardized format of XidML files, the configuration of an instrument can be easily imported in the toolchain. With the built-in version control information in the XML files, it's easy to keep track of modification and configuration evolutions. Implementing XML parser along with the XidML and XdefML schemata (XSD), allows the validation of an imported file to ensure a valid and consistent configuration of the instrument.
- **FTI Asset management:** Instruments like sensors requires regular calibration. Other devices should undergo maintenance. Information like

calibration or maintenance interval can be implemented in the XdefML file. This could simplify any notification or alerting by the toolchain to remind the user about upcoming maintenance actions. Instruments in the FTI world are not meant to stay on the same place for its entire life time. They are shipped for maintenance/repair or they are being used during test campaigns somewhere around the globe. To comply with the national/international laws about export control and customs, but also to company-specific classification rules, dedicated information can be placed in the XdefML description to enable e. g. automated tagging of the instruments for e. g. shipping notes.

XdefML – the electronic datasheet

XdefML is a way to describe instruments, in particular it's specifications, available settings and their limits. As for XidML, there's a schema available to validate XdefML files to ensure a valid file. The XdefML schema provides up to three main elements [2]:

- **Instrument Identification:** Contains all information required to identify the instrument. Required elements are ManufacturerName and PartReference. (optional e. g.: SerialNumber, ModelReference)
- **Instrument Specification:** Contains information on the physical and logical characteristics of the instrument as well as some additional information. Required elements are MechanicalClass and FunctionalClass. (optional e. g.: Standalone, SlotsOccupied)
- **Instrument Configuration:** Contains all configuration options for the instrument. Typical elements are Parameters, Settings, Channels.

It's important to note that a XdefML file does not contain the actual configuration of an instrument. It provides information about the instrument settings, especially which setting are available and what are the constraints for these settings. It also holds information about which parameter are provided by the instrument and how they are defined. The interfaces of an instrument are described by Channels (CAN, Serial, AR429, Ethernet...). Their definition is also part of the XdefML files.

Combine XidML & XdefML

XidML and XdefML can be adopted solely, but combining both leads to that unique benefit of “decoupling” instrument configuration from instrument specification. Compared to other description standards, this combination adds another validation layer to ensure that an instrument is configured in accordance to its limits.

- **XML:** an XML parser with syntax highlighting provides feedback, if the file is “well formed”. That means for e. g. that every opening tag has also a closing tag. It does not check if the minimum required elements are present.
- **XdefML:** A valid XdefML file ensures that the instrument description in this file is valid against the XdefML schema. Any violation (missing/unknown element, value out of bound) will be reported by the tool.
- **XidML:** As for the XdefML, a valid XidML file means that the instrument configuration is valid. All elements required are present. But in this stage, a wrong setting (e. g. Channel Gain out of bounds) is not detected.
- **XidML & XdefML combined:** Having the possibility to read XidML files and validates them with the schema, but also with the associated XdefML files for the instrument, ensures that the instrument is correct configured and all setting values are within the instrument perimeter.

Collaboration

Standards like XidML are driven by collaboration. There are several ways to increase the collaboration level:

- **Open Standard:** An open standard is more visible to the SW developer community. This wider recognition scope allows a broader adoption of the standard across different FTI ecosystems.
- **Visibility and Transparency:** All XidML & XdefML information are available in the Internet for free use. Along with the documentation and schemata, there are also sample files available for free download. This fosters trust and wider acceptance within the community.
- **Change Requests:** If a user requires a modification of the schema, a change request process was established to

submit that request via the webpage [3].

- **Vendor-neutral:** The open approach prevents vendor lock-in. This allows vendors and users to develop tools to handle XidML & XdefML. It helps to foster a more innovative community.
- **Community driven:** XidML & XdefML is a community driven project, which aims not be a product or a service provided a single vendor. In fact it targets the contribution of all stakeholder in the FTI community. These standards should benefit from the huge level of experience in the FTI toolchain development community. A large community accelerates the development of tools.
- **Standardization Body:** The XidML Working Group acts as the standardization body. Its member are from ever Anybody interested in the standard can participate. Regular working group meetings ensures a certain ritual, allows newcomer to join and provides the possibility to exchange the various topics around XidML & XdefML. Submitted change requests will be discussed and decisions will be taken accordingly. This working group lends credibility to the standard and it's open access is intended to eliminate any entry barrier.

Tools

When working with XidML & XdefML files, user can choose different solutions to create, modify and validate the files:

- **Notepad++ :** With the XML plugin it is possible to assign the XidML or XdefML schema and to validate those files accordingly. Users can edit, view and validate files. Notepad++ is free to use
- **XdefML Assistant:** The application is available on the XidML.org webpage for download and free usage. XdefML files can be created, modified and validated. It supports different views.
- **Altova XMLSpy:** XMLSpy is a professional XML editor with enhanced edit functionality. It provides multiple views and automatic validation. User can even create and modify the schema. It's a license-based application with different variants.

Summary

XidML and XdefML can help FTI users in various domains due to its open approach, its

vendor-neutrality and the resources it provides to adopt the standards. Everyone is welcome to get familiar with the schemata and the

documentation and to contribute to the evolution and wider acceptance of these standard.

References

- [1] XidML.org XidML Schema definition:
<https://www.xidml.org/library/XidML-3.0.0.html/html/XidML-3.0.0.html> (accessed 20.05.2024)
- [2] XidML.org XdefML Schema documentation:
<https://www.xidml.org/library/XdefML-3.0.0.html/html/XdefML-3.0.0.html> (accessed 20.05.2024)
- [3] XidML.org Change Request :
<https://xidml.org/change-requests/> (accessed 20.05.2024)