

MDF4 Lib

High-Performance Function Library for Reading and Writing MDF3 and MDF4 Files

What is MDF4 Lib?

MDF4 Lib is a powerful function library; in its base variant you can use it to validate and sort MDF files as well as read them into your own applications. The new ASAM-standardized MDF4 is supported in addition to the widely used MDF3 format. The MDF4 Lib base variant can be expanded by adding an option that supports writing files, allowing users to both create and read MDF files. The library offers a convenient interface, which you can use to access signal data and supplemental information in an MDF file, regardless of the specific MDF version (3.x/4.x). MDF4 Lib is available for C++ and Python in Windows and Linux as well as for .NET in Windows.

Overview of Advantages

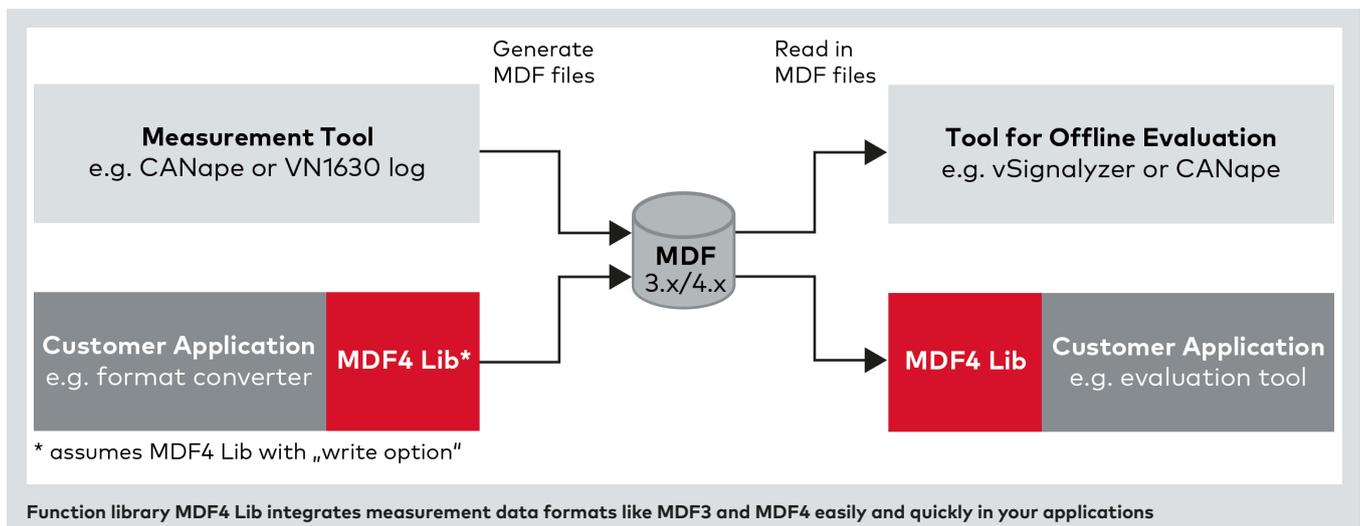
- > Easy to use function library for generating*, validating**, sorting** and reading MDF files
- > Version-independent access to MDF3 and MDF4 formats
- > Quick and memory-optimized opening of multiple MDF files
- > Efficient reading of signal data for sorted files
- > Easy generation of MDF objects and writing of signal data for the offline use case*
- > Convenient access to various types of MDF meta information

- > Seamless data transfer from Vector tools and data loggers, which offer MDF as an output or export format
- > Use of a tested and practice proven standard component reduces effort for training, development, testing and maintenance

* requires a "write option"
 ** currently only available in Windows

Highlights of Version 1.7

- > A new interface enables convenient access to all library methods with the popular Python programming language (Windows and Linux, Python 3.4 and higher)
- > Easy conversion of an existing MDF 4.x file to an MDF 4.1 file with compressed or uncompressed signal data (zip/unzip)
- > To enable compact storage of nested structures with arrays as members, the function library now supports "arrays of structures."
- > Read and write time synchronization information (name and class of time source)
- > Supports new flags and data types which will be introduced in ASAM MDF version 4.2 (release planned for mid-2019)
- > Performance optimizations for reading multiple physical signal values with just one call (MultiRead methods) if a linear conversion rule is used



Features

- > Multiple MDF files may be opened or generated* starting from a central File Manager Object. Related sub-objects might also be read or written, depending on the desired information. The specific objects and their methods are accessed via interface pointers. This lets you evaluate the MDF features independent of the MDF version.
- > For quick access to signal values, the MDF files can be sorted** before opening. In writing*, the MDF4 Lib only supports generation of already sorted MDF files, i.e. only records for one sampling rate may be written at a time (offline use case).
- > The object orientation and hierarchical structuring of the interface (API) lets users quickly learn and intuitively apply the available functionality. A detailed online Help and sample projects with documented source code permit quick training without requiring detailed knowledge of the MDF format.
- > The generation of objects "on demand" and their management by "reference counting" permit efficient and resource-economic implementations. In C++, when the supplied "smart pointer" class is used, it is even possible to omit an explicit release of the objects. This approach helps to avoid resource leaks. It also reduces the scope and complexity of your code, which generally improves its maintainability and readability.
- > All library calls are thread-safe. The library thus may be used in a multi-threading use case, e.g. to evaluate each MDF file in a separate thread. Additional optimizations may be enabled for this case.
- > For diagnostics, the logging capabilities of MDF4 Lib can be used to log any error and warning messages or optionally all library calls.

Variants

The base variant of the MDF4 Lib can be extended by a "write option" that makes it possible not only to read, but also to generate MDF files. As is the case with reading, all MDF versions are also supported in writing, but when writing an older MDF version certain features may not be available in the selected version. Currently, the MDF4 Lib only supports writing of sorted MDF files (offline use case). In Windows, MDF4 Lib comes with 32-bit or 64-bit versions. In addition, there is a Unicode and Multibyte version (MBCS). In Linux, it is available only as a 64-bit version with UTF-8 string support.

Supported MDF Versions

- > MDF 2.0 to MDF 2.1
- > MDF 3.0 to MDF 3.3
- > MDF 4.0 to MDF 4.1

Because of the special properties of the MDF format, MDF files of a future version 3.x/4.x will also be readable, provided that newly introduced features do not explicitly prevent this. Of course, access to the new features is not possible until the MDF4 Lib is updated.

More information: www.vector.com/mdf4_lib