

Integration of MDF4.1.1 in ASAM ODS

Markus Schuldlos

Integration of MDF 4.1.1 in ODS

Today's status

ASAM ODS and ASAM MDF 4 are both standard specifications of ASAM.

- 🕒 MDF is a standard to capture mass data
- 🕒 ODS is a standard to describe, store, analyze and archive measurement / mass data

Objective of today's workshops:

Approach to integrate MDF 4 as ExternalComponent-Files in ODS

Integration of MDF 4.1.1 in ODS

Today's status

ASAM ODS and ASAM MDF 4 are both standard specifications of ASAM.

- 🕒 MDF is a standard to capture mass data
- 🕒 ODS is a standard to describe, store, analyze and archive measurement / mass data

Objective of today's workshops:

Approach to integrate MDF 4 as ExternalComponent-Files in ODS

Reminder:

The data is captured once (1x), read and analyzed multiple times (X times). A prototype revealed that reading from various MDF files may up to 7 times slower than from ODS compatible formats.

Integration of MDF 4.1.1 in ODS

Incompatibilities 1/3

MDF is a standard to capture mass data

- 1) MDF 4 writes mass data in small data blocks, which may be utilized with the help of information in (DL-)lists
- 2) MDF 4 uses „Invalidation Bits“. These are separate bit-channels to describe the validity of single values

Integration of MDF 4.1.1 in ODS

Incompatibilities 2/3

ODS is a standard to describe, store, analyze and archive measurement / mass data.

- 1) ODS describes the storage of values in a LocalColumn. For one data block there is an instance of an ExternalComponent. ODS may include a list of ExternalComponents per LocalColumn.
- 2) ODS utilizes flags, individually a 16-bit value with various information

Integration of MDF 4.1.1 in ODS

Incompatibilities 3/3

- 3) MDF 4 specifies „Rational Conversion“; **is not available in ODS**
- 4) MDF specifies FORMULA; **is not available in ODS**
- 5) MDF specifies ARRAY; **this may be covered with ODS MIME-types, approach to be done**
- 6) MDF stores Text-channel information in individual single blocks; **ODS stores this kind of information in a combined way (NUL-character separated)**
- 7) MDF utilizes UTF-16; **ODS has no support**
- 8) MDF utilizes LOOK-UP channels; **ODS has no support**
- 9) MDF may store preview tables of channels; **ODS has no support**
- 10) MDF may zip data; **ODS has no support**
- 11) MDF may store BUS data; **ODS defines a BUS data model with corresponding MIME-types**

Integration of MDF 4.1.1 in ODS

Proof of Concept: MDF 4 as ExternalComponents Files in ODS

MDF 4.1.1 Proof of Concept

Importer based on
ModelMapper



Enhancement of ODS-
data model*



Integration of
Zip-algorithm*



Web-based
MDF 4 preview**



General fact: Most MDF 4 files may be integrated into ODS. The result of the PoC contains proposals for the ODS base model enhancements* as well as MIME-type definitions.

- 🔗 Attribute „zippt?“ with values „yes/no“ in ExternalComponent*
- 🔗 Attribute „transpose?“ with values „yes/no“ in ExternalComponent*
- 🔗 Integration of MDF 4 preview in ODS with MIME-Type**

Integration of MDF 4.1.1 in ODS

Proof of Concept: MDF 4 as ExternalComponents Files in ODS

There are further challenges in the integration:

- 🕒 The number of ExternalComponents instances in ODS may increase heavily (if the MDF 4 files include lists and values are stored in many, small data blocks)

This problem may only reasonably be solved by a „sorting/converting mechanism“.

- 🕒 It is not possible to address „invalidation bits“ as flags (MDF 4 files may not directly be read with standard mechanisms)

Invalidation Bits must separately imported as „flags“. An approach is defined by HighQSoft. This, however, causes a data redundancy. For an identification of „invalidation bits“ it is necessary to enhance the ODS standard.

Integration of MDF 4.1.1 in ODS

Open Issues: MDF 4 as ExternalComponents Files in ODS

- 1) Integration of MDF 4 data blocks; Concept available; Number of ExtComp are high; „Value-Separation“
- 2) Invalidation Bits; ODS concept to be discussed
- 3) MDF 4 specifies „Rational Conversion“; will be available in ODS 6
- 4) MDF specifies FORMULA; MDF: is not utilized; ODS: not yet identified in files
- 5) MDF specifies ARRAY; this may be covered with ODS MIME-types, approach to be done
- 6) MDF stores Text-channel information in individual single blocks; ODS stores this kind of information in a combined way (NUL-character separated); re-defined storage in MDF required
- 7) MDF utilizes UTF-16; ODS has no support
- 8) MDF utilizes LOOK-UP channels; ODS MIME-types are defined and will be available in ODS 6
- 9) MDF may store preview tables of channels, ODS MIME-types are defined and will be available in ODS 6
- 10) MDF may zip data; concept is implemented and available
- 5) MDF may store BUS data; ODS defines a BUS data model with corresponding MIME-types

Integration of MDF 4.1.1 in ODS

Open Issues: MDF 4 as ExternalComponents Files in ODS

MDF may store data records – **this also refers to a single value** – within multiple / different data blocks. ODS only works with single, complete data blocks.

Example: data record with 15 Bytes

1. Channel double value (8 Bytes)
2. Channel float value (4 Bytes),
3. Channel short value (2 Bytes)
4. Channel byte value (1 Byte)

🕒 Size of data blocks: $256 * 1024$ Bytes.

🕒 Distribution of record in data blocks

1. Data block contains: 17476 records + 4 bytes from subsequent record (**value of first channel is distributed in two blocks**)
2. Data block contains: 11 Bytes + 17475 records + 8 Bytes from subsequent record

Integration of MDF 4.1.1 in ODS

Summary

Recommendation from HighQSoft for ASAM:

- 🕒 Pick up existing proposal to enhance the ASAM ODS standard (possibilities?)

- 🕒 ODS base model enhancements
- 🕒 MIME-Type definitions
- 🕒 Invalidation Bits (concept needed)
- 🕒 Zip / Transponse

- 🕒 Changes in MDF 4 standards towards (possibilities?)

- 🕒 „Data records / data block“ problem
- 🕒 text-channel information

- 🕒 Converting of MDF 4 files for optimization for reading (block size and ordinance)

Disclaimer: MDF 4 files will only be read by ODS server

Integration of MDF 4.1.1 in ODS

Summary

Recommendation from HighQSoft for ASAM:

🕒 Pick up existing proposal to enhance the ASAM ODS standard (possibilities?)

🕒 ODS base model enhancements

🕒 MIME-Type definitions

🕒 Invalidation Bits (cor)

🕒 Zip / Transponse

The objective: Make MDF work with ODS:
This results in compatibility, performance and low data
redundancy!

🕒 Changes in MDF 4 standards towards (possibilities?)

🕒 „Data records / data block“ problem

🕒 text-channel information

🕒 Converting of MDF 4 files for optimization for reading (block size and ordinance)

Disclaimer: MDF 4 files will only be read by ODS server

Integration of MDF 4.1.1 in ODS

Thank you



HighQSoft GmbH
Schloßborner Weg 6b
61479 Glashütten
Germany



Markus Schuldlos
+49 171 2415972
markus.schuldlos@highqsoft.de