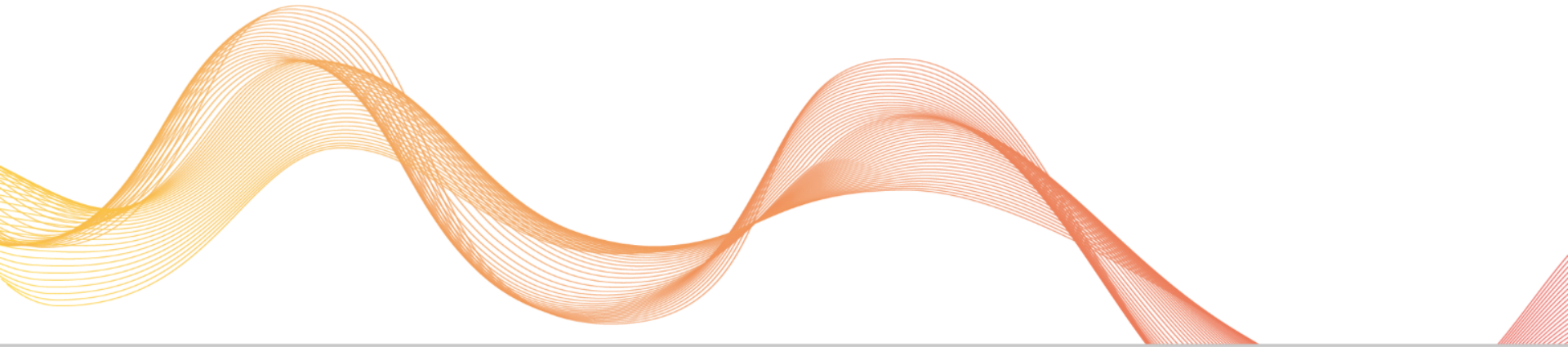




Managing MDF 4 with ODS: Requirements and Solutions

May 12, 2016, BMW Trainingsakademie, Unterschleißheim

Moderator

Thomas Thomsen

ASAM e.V.

Agenda

1	Welcome	Thomsen, ASAM
2	Introduction of the Participants	All
3	Brief Introduction to ASAM Implementation Projects	Thomsen, ASAM
4	Situation at BMW: Test Data Management with OpenMDM and Powertrain Teststand ODS Database	Kemle, BMW
5	Java MDF4 Sorter Library	Wöhrl, Audi
6	MDF 4 Experiences / Enhancement Proposal	Fischer, AVL
7	MDF4 Virtual File Server: A Comprehensive Approach for the Integration of the MDF4 File Format as a Binary Component into the ASAM ODS Standard	Koller, Peak Solution
8	Integration von MDF4.1.1 in ASAM ODS	Schuldlos, HighQSoft
9	Open Discussion	All
10	Next Steps	All

Open Discussion

Three Main Topics

1. Short-Term:

Development of a Virtual Server Component

- may includes short-term changes of the standards ODS (e.g. MDF rules) and MDF

16 votes

2. Short-Term:

File Converter

- may includes short-term changes of ODS

12 votes

3. Mid/Long-Term:

Further Development of MDF and ODS

- resolve incompatibilities between MDF and ODS
- create an API interface between MDF and ODS

20 votes



Development of a Virtual Server Component

Discussion Topics

1. Project Type
(standard development, software development, or both?)
2. Market Relevance
3. Technical Use-Cases
4. Technical Approach
(file server, API definition, implementation techniques?)
5. Which Components Shall Be Developed?
6. Required Features
7. Deliverables
(source code, dll, tool?)

1. Project Type

- ▶ Project type: implementation project

- ▶ Supported standard versions:
 - ASAM MDF 4.x
 - ASAM ODS 5.3.x

2. Market Relevance

- ▶ Which end-users would use the virtual server?
 - will primarily be used in conjunction with ODS servers
 - will make both standards more attractive and promote the proliferation of the standards
 - will make MDF4 available for all ODS-compliant products

- ▶ Which tool vendors would integrate the virtual server into their product?
 - abstracts file access and saves the extra effort for tool vendors to specify access to each individual file format
 - gain experience in standardizing a future data access API
 - pre-study for Big Data systems
 - allows client application to access MDF4 files, which have no MDF4 importer
 - concerns about performance: potential miss of opportunity, shall be tackled with MDF standard development

3. Technical Use Cases

- ▶ Seamless and transparent read-only access of MDF data towards an ODS server. Resolve incompatibilities between both standards.
 - Transform data types in a defined way
 - Split-records
 - Compression
 - Unsupported conversion types
 - For further information, see ASAM ODS chapter 2.3

4. Technical Approach

- ▶ Language: C++14
- ▶ OS: Windows Server 2012, Linux (e.g. RedHat)
- ▶ File systems: NFS, CIFS
- ▶ MDF4 Lib: use and recommend www.turbolab.de, can be distributed by ASAM
- ▶ Software specification shall be written prior of development of the server

5. Which Components Shall Be Developed?

- ▶ Channel Meta-Data Provider
(generates ATFX files)
 - channel names
(need solution for duplicate names from MDF)
 - channel conversions plus conversion parameters
 - units
 - submatrices
 - datatypes
 - and any further meta data that shall be known by the ODS data model
 - data access information
- ▶ File Server
 - Serves virtual files that are referenced in the external components according to the ATFX files

6. Required Features

- ▶ Read-only access to the measurement data of specific channels
- ▶ Generate and store data access information in the ATFX header
- ▶ Full support for all block types
(no multi-dimensional data, no structs, no events)
- ▶ Support for invalidation bits
- ▶ Convert not supported data types
- ▶ Apply all unsupported conversion methods of ODS

Not Required

- ▶ Support for bus data

Deliverables

- ▶ Installable executable
(e.g. as an OS service)
- ▶ Source code plus build instructions
- ▶ Test environment
- ▶ Documentation
- ▶ Examples