Visual Information Technologies GmbH

Data Declaration System

What's new in Version 5.2.0?

Jun 26th 2007

DDS Contact
email: dds@visu-it.de
Internet: http://www.visu-it.de/dds

Contents

Visual Information Technologies GmbH

1   Purpose and scope

This document describes the improvements and changes of DDS V5.2.0 compared to V5.1.1.

2   What's new in Version V5.2.0?

2.1 Highlights

2.1.1   DDX Concept

The objectives behind the DDX (Data Declaration eXchange) file format are:
- provide a comprehensive XML file/schema which is able to cover the current data declaration contents, e.g. ADD and DDS and ASAP2 contents
- use standards as much as possible
  -> The DDX format follows the "ASAM MCD 2 Harmonized Data Objects" data model for base elements like conversions, dataTypes, etc.
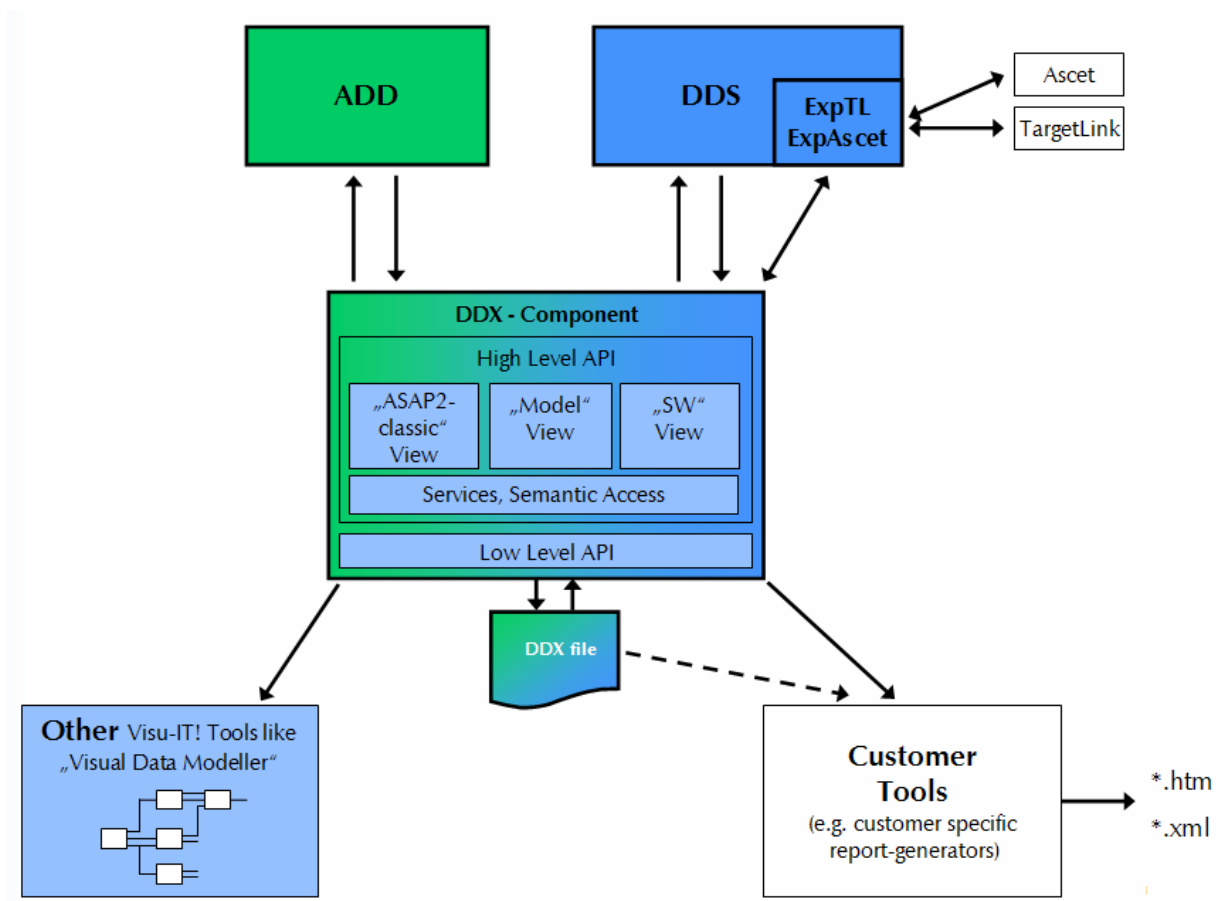
Note: The components of the DDX package are not included in the standard DDS installation but are available as Add'on.

2.1.1.1   "Export DDX" improvements

Improvements:

- new filter options "PartialExportViaFunction", "PartialExportViaFunctionMode", "PartialExportViaConfUnit", "CompuMethodName", "ContentLevel", etc.
- support of further attributes "maxRefresh", "accessFrequency", "bitmask", "ddBitType", etc.
- etc.

2.1.1.2 *"Export to Ascet" improvements*

Improvements:
- new filter options "AscetProject", "ContentLevel", "ExistingDBUpdateMode", "BitHandling", etc.
- support of further mapping variables
- available as Ascet plug in
- etc.

2.1.1.3 *"Export to TargetLink" improvements*

Improvements:
- new filter options "GenerateMFilesOnly", "ContentLevel", "ExistingDBUpdateMode", "BitHandling", etc.
- support of further mapping variables, e.g. $STRUCT-INSTANCE$, $CONF-UNIT$
- support of regular expressions
- support of variant coding
- etc.

2.1.1.4 *NEW: "Import DDX"*

New DDS filter to re-import DDX files into DDS.

2.1.2 New component/filter: "Export/Update Data Record (ExpDR)"

There is a new DDS component "Export/Update data record".
This new filter allows to update (create) the data record of an Intel hex file with the (init-)values defined in the DDS database.

Restrictions:
- The data record must be located in an Intel hex file
- The first version only supports the 32bit Intel hex format
- The corresponding database must contain valid addresses (e.g. perform address import before the update of the data record)

Note: This component is not included in the standard DDS installation but are available as Add'on.

2.1.3 ExpA2, ImpA2: Deterministic conversion names

Topic #50, #47:
Keep names of COMPU_METHOD definitions deterministic…

Improvements:

Level 1: Generate deterministic conversion names
There is a new ExpA2 filter option „CompuMethodName" with the following values:

Traditional	Generate pre V5.2 conversion names (default)
Deterministic	Generate deterministic conversion names

| Deterministic_IgnoreLimits | ... as "Deterministic", except: Do not adapt the ranges of the (table-)conversions to the lower-/upper-limits of the using definitions |
| Deterministic_KeepExternal | ... as "Deterministic", except: Try to export the conversions marked with "External CompuMethod" without changing their name |

Level 2: Restore original DDS conversion name during ASAP2 Import
see "Level 3"

Level 3: Support non-DDS ASAP2 COMPU_METHOD names
For this purpose, there is the new ASAP2 Import filter option "ConversionName" with the following values:

Standard	Name of COMPU_METHOD (default)
RestoreDDSName	Restore the original DDS name, if possible
Set2External	As "Standard" but additionally mark all imported conversions as "do not modify/use inside DDS" (Purpose: Generate identical ASAP2 file)

2.1.4 ADD / DDS Bit-Handling

Topic (A29):
In ADD it is possible to declare definitions as bits/booleans by assigning a special datatype. However, when importing these bit/boolean definitions into DDS, each of them will occupy 1 byte of memory. Although it is possible to manually optimize the memory consumption of these bits, there should be a functionality in DDS to automatically do this.

Improvements:
There is a new concept in DDS which optimizes the memory consumption of ADD Bits without changing their software interface -> DDBit-Handling.
Several DDS components/modules are updated/improved: Database, ImpADD, GenEdit, SourceExport, Export Ascet, Export TargetLink.

2.2 New Features, Major Improvements

2.2.1   Support of C99 data type "_Bool"

Topic #178:
The C99 standard natively supports the data type '_Bool'.

Improvements:
DDS now supports '_Bool' as builtin data type.

Restriction:
-     '_Bool' is not supported for axes and maps but only for parameters and onlines (and cPointer)

2.2.2   Support of 'optimized' enum definitions

Topic (B):
In C99, the (byte-)size of an enum definition is not inevitable linked to the size of an integer. The C99 standard allows to minimize the (byte-)size of an enum definition if possible.

Improvements:
The DDS components 'ImpELF' and 'ExpA2' are improved in order to support an 'optimized' size of an enum definition. The ASAP2 export provides a new filter option "DetectSizeOfEnum" which activates the detection and handling of 'optimized' enum definitions.

2.2.3   ExpA2, ImpA2: Support symbolic array specifier

Topic #161:
The ASAP2 export only supports enum-values as array identifier for the DISPLAY_NAME, not for the real (software-)name.

Improvement:
There is the new filter option "SupportSymbolicArraySpecifier".
If this option is set, the Export ASAP2 filter handles symbolic array specifier (in form of enum values) for instrumentation data. e.g. exports "myData[RED]" instead of "myData[4]".
When re-importing those ASAP2 files back into DDS, the symbolic array specifies must be handled correctly.
The ASAP2 Import was adapted accordingly.

2.2.4   ExpA2: "Partial Export Via Function" Mode

Topic #153:
When exporting with "PartialExportViaFunction", implicit inputs are not exported
An implicit input might not be explicitly defined via the (REF-)function assignment but via the fact that the data is used:
a.)   from a map which is assigned to the function
        -> implicit input is an axis
b.)   from an axis which is assigned to the function
        -> implicit input is an online

Improvement:
There is a new ExpA2 filter option "PartialExportViaFunctionMode" with the following values:
•   With_SubFunctions (default)
        …behaves like today
•   With_SubFunctions_And_ImplicitInputs
        …additionally considers the implicit inputs

- Without_SubFunctions
  Export the selected functions(s) without their sub-functions

Note: This option is also available at the "Export DDX" filter.

2.2.5 ExpSrc, InstrPro: Default init-values

Topic #163:
Possibility to control the default init value needed.

Improvement:
There is a new ExpSrc and InstrPro filter option "DefaultInit" with the following values:
1) "SmallestValue" (default)
   Like in DDS V5.1.x
2) "Zero"
   The default value is '0'. If the value '0' is not possible/allowed, the smallest value permitted by the conversion will be used.

2.2.6 ExpSrc, InstrPro: Suffix for (init-)values

Topic #141:
Suffix 'U' should be used for 'unsigned' values …

Improvement:
- The user can assign an individual/custom suffix for each builtin dataType
- There is nevertheless the ExpSrc/InstrPro Filter Option "IgnoreDataTypeSuffix" in order to reproduce the 'old' behavior
- There is an additional new option "CastNegativeHexValues" in order to be able to cast negative hex values, .e.g cast "0x800" to "=(s32)0x8000U""

2.2.7 InstrPro: Support of enums and defines

Topic (B):
The instruction processor has to be extended in order to also support enums and cDefines.

Improvement:
- new label-interface directives: ENUM, DEFINE

2.3 Minor Improvements / Correction of Bugs

2.3.1 ImpADD: Upgrade history

Topic #117:
Possibility to display the history of a DDContainer Import needed.

Improvement:
There is a new ImpADD filter option "ActivateLogging". If this filter option is set, some information about the import/upgrade will be written into the "description" attribute of the corresponding DD-Container.

2.3.2 ImpADD: ADD LOC Parameter

Topic (A30):
In ADD it is possible to define parameters with the 'classification' LOCAL. When importing these parameters to DDS, no (cal-)function assignment can be established since both DDS and ASAP2 do not support local parameters/CHARACTERISTIC's.

Improvement:
Since the restriction above results from ASAP2, no improvement of DDS makes sense, but:
The already existing ImpADD filter option "GenerateFunctionBasedOnDDContainer" is extended like the following:

No	Do not automatically create a calFunction for each imported DDContainer (default)
Yes	a calFunction for each imported DDContainer
Yes_SetLocalCalEntitiesToOutput	**As "Yes" but additionally set 'LOCAL' calEntities (e.g. parameter, map, axes) to 'OUTPUT' to workaround an ASAP2 limitation ('LOCAL' calEntities are not supported in ASAP2)**

2.3.3 ImpADD: Misc. Improvements

Improvements:
- Location of (automatically) generated sourceSection and (unassigned-) c/h files can be controlled with the ImpADD filter option "InstrEntitiesBasedOnProxiesToUnit"
- Support of wildcard ('*') at ImpADD filter option "InputDefinitionsAndAttributesToUnit"

2.3.4 ImpA2: Bit detection

Improvements:
The ImpA2 filter is extended with a new filter option "BitDetection":
-> Try to detect and import 'real' bits (which are directly supported from the compiler, e.g. Tasking C167)
Prerequisites for "BitDetection":
- only applicable for onlines and parameters (no axes and no maps)
- only applicable for global definitions
  (no members of structures, since this might cause problems with bitfield-definitions)
- the ASAP2 datatype must have the size of 1 byte
- the ASAP2 definition in question must have a bitmask; the bitmask must only mask one bit

2.3.5 ImpA2: Misc. Improvements

Improvements:
- Support of Multi-Dim 'mixed' VAL_BLK definitions

2.3.6 ImpELF, LocatorImport: Misc. Improvements

Improvements:
- Toleration of TriCore '_bit' and improvement of bitfield-error message

2.3.7 GenEdit: Reference of (proxy-)axis to map

Topic #121:
The back reference from a ddProxy axis to a map is not displayed, since there is no 'real physical' (back-)reference form the axis to the map.

Improvement:
These 'virtual' back-references are now displayed in blue color.

2.3.8 GenEdit: Ignore filter options in command line

Topic #152:
When launching GenEdit.exe in command line, the filter options are not taken into consideration.

Improvement:
Fixed.

2.3.9 GenEdit: Persistence flags

Topic (B):
The persistence flags in the definition panel overlap partially. Furthermore, switching between 'extern' and 'static' does not work correctly.

Improvement:
Fixed.

Additional Improvements:
- Support of different options when switching between "Quick ASAP2 Export" and "ASAP2 Export"

2.3.10 CmpMrg: Misc. Improvements

Improvements:
- The filter option "AllowAdditionalSystemEntitiesInMerge" now also supports 'raster', 'onlineUpdate' and 'timePeriod' definitions
- The global rule
  "*Output proxy wins against Input proxy*"
  is refined with the following one:
  "*Output proxy **and related instrEntity** wins against Input proxy **and related instrEntity**"
  Constraints: Works only for global definitions without arraySize! Definitions with arraySize will be handled as before.

2.3.11 GenCheck: Check of axis-online attributes

Topic (B):
In general, when an axis references an online (working point) the attributes 'dataType (elemType)', 'conversion' and 'physUnit' should be the same for the axis and the online.

Improvement:
The data pool check now emits up to three different warnings, when these attributes are different.

2.3.12 GenCheck: Repair DB in command line

Topic #162:
Repairing a database in command line is not possible since "GenCheck.exe" never saves the database.

Improvement:
Done.

2.3.13 ExpSrc: Skip invisible data

Topic (A):
Local definitions which are marked as 'invisible' should not be exported to source code.

Improvement:
There is the new ExpSrc filter option "SkipInvisibleData". This option controls how to handle data which is defined as 'invisible' during source export. The following values are provided:

No	Do not skip data which is set to 'invisible' (default)
LOCAL_Onlines	Skip all onlines which are LOCAL and 'invisible'. In that case 'LOCAL' means: the assigned DDProxy definition must be defined as 'LOCAL' in the corresponding DDContainer
All_LOCAL_Data	Skip all definitions which are LOCAL and 'invisible'. In that case 'LOCAL' means: the assigned DDProxy definition must be defined as 'LOCAL' in the corresponding DDContainer

Additional Improvements:
- Fix: Double entries in "Software Interface Block"

2.3.14 ExpSrc: Misc. Improvements

Improvements:
- New filter option "WriteIncludeAboveCFileSourceSection" to control whether a possible implicit include statement (that means, an 'include' caused due to dependencies) should be written before/above the cFile 'begin' sourceSection or not.

2.3.15 InstrPro: Misc. Improvements

Improvements:
- Support of evaluatable defines
- Individual (error-)messages for (optional) attributes which are neither defined in DDS nor in the label interface
- Complete and deterministic file name (no 8.3 notation) in first line of output files
- Support of "array_spec" directive for "kl_spec", "kf_spec" and "axis_spec"

2.3.16 ExpA2, ImpA2: Support of Mulit-Dim-VAL_BLK

Topic (A22):
Today all definitions of the type „VAL_BLK" are exported into ASAP2 as **one** definition.
The attribute „MATRIX_DIM" is used to specify the array dimension.

Improvement:
Now, there is a new ASAP2 export filter option "Force1DValBlks". If this filter option is set, only the last array-dimension is handled like VAL_BLK-arrays. The other array dimensions (if existing) are handled like 'normal' arrays (will be flattened into single array elements)
The ASAP2 import is updated accordingly in order to detect and correctly import 'mixed' multi-dimensional VAL_BLK arrays.

2.3.17 ExpA2: Support of 'MATRIX_DIM' in V1.41

Topic (B):
Today, the optional attribute 'MATRIX_DIM' is written to ASAP2 when the destination ASAP2 file version is at least 1.51. Since this keyword is valid since ASAP2 V1.41, it should already be written if the destination ASAP2 file version is at least 1.41.

Improvement:
Done.

2.3.18 ExpA2: Misc. Improvements

Improvements:
- Ignore address check for embedded axes
- Correction of 'PartialExportViaFunction' bug
- New filter option "ExportMaxRefresh"
- Unique name of "COMPU_METHOD"
- Correction of lower-/upper-limits of definitions using formula- and ratFunc-conversions

2.3.19 ImpCVX: Import of '0' value

Topic #183:
A problem occurs for definitions which:
a) have not yet an init value (in DDS)
b) should get the init value '0' during CVX import
In this case, the value '0' is regarded as 'default' value and the minimum init-value is written into the DDS DB.

Improvement:
Fixed.

2.3.20 DServer: Release database

Topic #146:
When opening a database fails, DServer does not always completely release the database.

Improvement:
Fixed.

2.3.21 SysConf: Scrolling menu within ODX part

Topic #150:
It's not possible to select a field in the scrolling menu in ODX part.

Improvement:
Fixed.

2.3.22 SysConf: Wrong handling with extended CAN ID

Topic #144:
According to the ASAP2 specification it depends on bit 31 if a CAN ID is extended (29bit) or standard (11 bit). In case of a extended CAN ID it is not possible to configure the correct CAN ID, in which bit31 is set to 1, for XCP DAQ Lists in the SysConf 5.1.0 version

Improvement:
Fixed.

2.3.23 SysConf: ETK Support

Topic #125:
At smartVariables, allow to select array- and structure-members as values

Improvement:
Done.

2.3.24 SysConf: Problem with maximum limit of memoryRegions

Topic #145:
The SysConf GUI does sometimes not display the correct (not negative) value.
Especially when calculating the "end address", the result might not be correct.

Improvement:
Fixed.

2.3.25 DPLib: "MaxLineLenInGrlFile"

Improvements:
There is a new registry setting "MaxLineLenInGrlFile" which controls the maximum line length in the Girl files (Default value: 2048 character)

2.3.26 Specific XML-Converter: "DDX" to "customer specific XML file"

Customer specific XML converter:
- converts DDX to customer specific XML file
- converts customer specific XML file to DDX
- uses XSLT technique
- can be launched in pre- and post-process
- etc.

3   No longer supported filters/components

4   Questions?

If you have any questions about this release, please contact the Visu-IT! DDS Hotline:

VISU IT!
Visual Information Technologies GmbH

DDS Hotline	DDS Contact	DDS Product page
Tel.: +49 (0)9943 – 943563	Tel.: +49 (0)9943 – 943561	Internet: http://www.visu-it.de/dds
email: dds-hotline@visu-it.de	email: dds@visu-it.de	