Data Declaration System

What's new in Version 5.6.R1?

May 07th 2009

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

Contents

1 Purpose and scope

This document describes the improvements and changes of DDS V5.6.R0 compared to V5.5.R1.

2 What's new in Version V5.6.R0?

Improvements and changes of DDS V5.6.R0 compared to V5.5.R1.

2.1 New Features

2.1.1 Support of Hightec compiler V3.4.5.1

DDS now supports the Hightec compiler V3.4.5.1.

2.1.2 Rewrite Conf. units

Via the menu entry "Project | Rewrite Conf. Units…" it is now possible to select Conf. Units (GRL files) which should be rewritten, e.g. in order to create GRL files with a deterministic order of entities.

2.1.3 Multiple ELF

DDS is no capable to import ELF files composed of more than one compiler.

2.1.4 Individual filter settings

Each filter option dialog has now the functionality to SAVE and OPEN a special *.gpp file which is individual for the current filter. That means, these options overwrite all other options (e.g. from DDS.gpp) in any case, means even if not set. The same behavior can be achieved in command line via the new overall filter option "`IndirectOnly`"

2.1.5 Performance improvement for huge ASAP2 files

Huge ASAP2 files can now be imported into DDS much faster as before.

2.2 Minor Improvements / Correction of Bugs

2.2.1 Performance improvement when processing big arrays

Big arrays can now be processed much more efficiently. The following components have been 'tuned':

- Export ASAP2
- Export DDX
- Export MDX
- Import ADD
- Import ASAP2
- DDS Editor

2.2.2 Protocol specific raster-ID

The raster-ID can now have an individual value for different protocols (e.g. raster ID for CCP starts at '0' but for ETK it could start at '1')

2.2.3 DDS Editor: Miscellaneous improvements

Improvements:

- Improved calculation of offset- and resolution values at linear conversions

2.2.4 Import ADD: Miscellaneous improvements

Improvements:

- Handling of arrays
  Arrays (of onlines and parameters) can now be treated as VAL_BLK definitions when activating the new filter option "`TreatArraysAsVALBLK`"
- Improved generation/assignment of calFunction
- Improved check: check non-proxy INPUT vs. proxy INTPUT

2.2.5 Import ELF: Miscellaneous improvements

Improvements:

- Detection of alignment issues at mappingScheme 55 (already provided for DDS V5.5.R2)
- Support of option "AllowArrayDimMismatch" also for structures
- Support of ‚bit' datatype
  DDS now supports 'real' bits which are located in one byte, means the bitOffset is > 0
- Toleration of mappingSchemes without member-Names
  When a map with embedded axes has a mappingScheme which has no member-Name inside, DDS emits a warning but tries to match the members via index position.
- Toleration of "0" addresses
  In same cases, variable have no real address in the DEBUG part of the ELF file but contain only the address „0". DDS parses all compilation units to find the correct address. If no address is found, DDS asks the user to take the address from the GLOBAL section.
- When compiling sources with some TriCore specific datatyes, the layout of the DEBUG sections changes. DDS can now handle this layout change properly
- Support of Hitachi SHC:V9.02.00.003:SH2A-FPU

2.2.6 Import ASAP2: Miscellaneous improvements

Improvements:

- Improvement/correction of map-(embedded)axis assignment within structures. The assignment within the structure-instance was redundant and caused a problem in CmpMrg.

2.2.7 Import CVX: Miscellaneous improvements

Improvements:

- Improved handling of axes with a reduced axes size in CVX
  -> DDS ensures that the axes values are monotone even if some values are missing in CVX
- Correct handling of maps with dimension 1

2.2.8 Export ASAP2: Miscellaneous improvements

Improvements:

- Custom AML files possible via new filter option "`CustomAMLFiles`"
- Creation of RESERVED elements within RECORD_LAYOUTS for detected alignment issues at mappingScheme 55 (already provided for DDS V5.5.R2)

2.2.9 Export Source: Miscellaneous improvements

Improvements:

- Correction of line break within preprocessor directives resulting from code variant configurations in ADD (already provided for DDS V5.5.R3)
- Pointers will be now declared as 'const' when they have an symbolic init value

2.2.10 Export DDX: Miscellaneous improvements

Improvements:

- Correction of BYTE-ORDER in DDX

2.3 AddOn components

2.3.1 Export to Ascet: Miscellaneous improvements

2.3.2 Export to TargetLink: Miscellaneous improvements

Improvements:

- Now also the init-values of online data can be exported to TL DD. Previously, only the init values of calibration entities were exported
- Correction of bug: Data types starting with 'B' were not always correctly mapped into TL data types

2.3.3 ADDS Interface: Miscellaneous improvements

Improvements:

- Process "TargetLink -> ADD" is now able to extract the essential ADD information from the main subsystem of a selected TargetLink model
- ADDS Interface is now able to support the model documentation tool FunDoc

3 What's new in Version V5.6.R1?

Improvements and changes of DDS V5.6.R1 compared to V5.6.R0.

3.1 Misc. Improvements

3.1.1 ImpADD

Improvements:

- Support of code sharing adapter "CsAdp"
  DDS now supports the special handling of the code sharing adapter "CsAdp_<*DDProject*>". For that reason, the two new filter options "CsAdpSystemConstantPattern" and "CsAdpSystemConstantDefFile" are introduced.

- Automatic code variants for "unassigned INPUT" definitions
  For definitions which are exclusively used as INPUTs (-> "unassigned INPUT" definitions), the import filter

combines all it's assigned (INPUT-)code variants in ADD into one source section in DDS and automatically assigns it to the definition. The automatic source section will only be created if:

- the container is imported into DDS in project scope (means, the option "DDProject" must be set)
- the INPUT definition has a code variant in every occurrence within the project (means, wherever the definition is used in a container, it must have a code variant)

The automatic combined source sections help to identify the actually needed definitions.

3.1.2 ImpA2

Improvements:

- Correction of erroneous behavior at filter option "MappingScheme4Maps"

3.1.3 ImpELF

Improvements:

- Bitfield-Detection
  The ELF import is now able to detect and correct a mismatch at bitfields. When the import detects a bitfield with one member in the ELF file and a non-bitfield but a 'normal' structure with also one member (which has a bitmask) in DDS, the import 'converts' the DDS structure into a bitfield structure.
- Automatic mapping scheme assignment
  The ELF import is now able to automatically assign a matching mapping scheme if the mapping scheme in DDS doesn't fit to the code representation in the ELF file. The automatic assignment can be suppressed via the new filter option "SkipAutoMappingSchemeAssignment"

3.1.4 ImpI3E

Improvements:

- Support of I3E files without "block-size" (e.g. compiler for M16 processor)

3.1.5 Export ASAP2

Improvements:

- Bugfix: The ASAP2 export created to many RESERVED attributes in some cases
- Bugfix: The ASAP2 export didn't correctly handle empty displayNames

4 No longer supported filters/components

5 Questions?

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

DDS Hotline	DDS Contact	DDS Product page
Tel.: +49 (0)941 / 49082 - 16 email: dds-hotline@visu-it.de	Tel.: +49 (0)9947 / 9040004 email: dds@visu-it.de	Internet: http://www.visu-it.de/dds