DFDL4S++ Version 1.0 - Release Notes

1. NEW RELEASE DESCRIPTION

DFDL4S++ is a C++ library providing a sub-set of the methods exposed by the DFDL4S Java library.
DFDL4S is a generic binary data binding library written in Java and based on the Data Format Description Language (DFDL), see more details at:
https://eop-cfi.esa.int/index.php/applications/dfdl4s

1.1 Software and Hardware Requirements

DFDL4S++ is available for the following computer platforms:

- Linux (64-bit)
  - g++ compiler 64bit (v4.8+)
- Mac OS X Intel 10.5 or above (64-bit)
  - Apple LLVM v8.1.0 (clang-802.0.42) 64 bit
- Windows XP / Vista / 7 (64-bit)
  - Microsoft Visual Studio 14.0 Express 64 bit

The DFDL4S++ library requires:

- Version 1.8 or higher of the Java Development kit
- 50+ Mb of hard disk space
- 2 GB RAM

1.2 Installation Packages

A DFDL4S++ distribution package consists of one single archive (one per ache platform):

- DFDL4S-CPP-1.0-linux64.tar.gz
- DFDL4S-CPP-1.0-mac64.tar.gz
- DFDL4S-CPP-1.0-win64.tar.gz

1.3 Installation Hints

In order to install DFDL4S++, the distribution package needs to be unzipped into the selected installation directory.

The following directories are created:

- lib: contains the library binary files
- include: contains C++ header files
- examples: contains examples on how to write,build an run a C++ program using DFDL4S++
- docs: contains doxygen documentation

1.4 Documentation

For general information related to usage of the DFDL4S library, please refer to the documentation of DFDL4S Java API, in particular:

- Developer Manual: DFDL4S_Developers_Manual_S2G-DME-TEC-SUM078-1F.pdf

- Mission Specification Schemas: Mission_Specification_Schemas_S2G-DME-TEC-SUM092-1B.pdf

Both documents can be downloaded from the following web page:
https://eop-cfi.esa.int/Repo/PUBLIC/DOCUMENTATION/APPLICATIONS/DFDL4S/

For details related to C++ classes and methods prototypes, please refer to the doxygen documentation provided within the distribution package in the docs directory.

2. NEW FUNCTIONALITIES

This release 1.0 provides the following sub-set of classes and methods exposed by the DFDL4S Java library, the most important are mentioned below:

- DFDLLib class:
    - interpretDocument
    - createDocument
    - getSchemaDefinition
    - appendElements
- Document class:
    - childCount
    - childAt
    - close
- Element class:
    - getValueBytes
    - getValueFloat32
    - getValueFloat64
    - getValueInteger
    - getValueTime
    - setValueBytes
    - setValueFloat32
    - setValueFloat64
    - setValueInteger
    - setValueTime
    - getValueExadecimal
    - getValueAsString
    - getRangeMaximum / GetRangeMinimum
    - getIntrinsicType
- ElementFinder class:
    - getElement

3. CLOSED SPRS

N/A

4. KNOWN PROBLEMS

[DFDL4S-ANR-0029] Empty contructors not available
e.g. Element and Document cannot be initialised like this:
Element el;
but they can be only initialised as assignment using e.g. get methods.
Note that an object cannot be re-used (see also Example.cpp).

[DFDL4S-ANR-0030] childAt() does not work properly in certain conditions
e.g. in a loop that creates nofPackets packets and fills them:

```
for (id=0;id<nofPackets;id++)
  {
  dfdl_lib.appendElements( &document_1, &schema, data );
  Element el = document.childAt( id );
  }
```

This does not work, the accessed element is always the first.

The workaround (see also Example.cpp) is to create the file with all packets, close the file, re-open it and fill each packet:

```
std::vector<unsigned char> data( nofPackets*size, 0 );
dfdl_lib.appendElements( &document_1, &schema, data );
document_1.close();

Document document_2 = dfdl_lib.interpretDocument( schema_file, filename );

for (id=0;id<nofPackets;id++)
  {
  Element el = document_2.childAt( id );
```

5. REPORTING PROBLEMS

For any problems or questions please send an e-mail to the DFDL4S helpdesk:
dfdl4s@eopp.esa.int