**ISO/IEC JTC 1/SC 29/WG 03 N1116**

**ISO/IEC JTC 1/SC 29/WG 03  
MPEG Systems   
Convenorship: KATS (Korea, Republic of)**

**Document type:** Output Document

**Title:** Draft DoC on ISO/IEC DIS 14496-34 Syntactic description language

**Status:** Approved

**Date of document:** 2023-01-26

**Source:** ISO/IEC JTC 1/SC 29/WG 03

**No. of pages:** 1 (with cover page)

**Email of Convenor:** young.L @ samsung . com

**Committee URL:** <https://isotc.iso.org/livelink/livelink/open/jtc1sc29wg3>

**INTERNATIONAL ORGANIZATION FOR STANDARDIZATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC 1/SC 29/WG 03 MPEG SYSTEMS**

**ISO/IEC JTC 1/SC 29/WG 03 N1116**

**January 2024, Virtual**

|  |  |
| --- | --- |
| **Title** | **Draft DoC on ISO/IEC DIS 14496-34 Syntactic description language** |
| **Source** | **WG 03, MPEG Systems** |
| **Status** | **Approved** |
| **Serial Number** | **23483** |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| mNL-001 |  |  |  |  | The technical maturity of the DIS text is going in the right direction can be further improved. | Please consider WG03 N01036 as basis for further stage. | accepted |
| \*\*-002 |  |  |  | Ge | The document has been reviewed by an ISO editor. | Please use the file named “Word with trackchanges”, from ISO CS available [on ISO Projects](https://sd.iso.org/projects/my-projects?includeSC=false&stagesStatus=CURRENT&stagesStartDate=null,null&editionDate=null,null&correctedDate=null,null&filters=ACTIVE_PRELIMINARY_PUBLISHED&sort=alert-desc&pageNumber=1&pageSize=20) as a basis for any further drafting | accepted |
| US-003 |  | 5.1 |  | TE | broken precedence  Lists precedence in order, multiplication has higher precedence than division and addition has higher precedence than subtraction but, in fact, multiplication and division have same precedence and addition and subtraction have same precedence. | Use proper definitions of precedence, including tiers. See any Computer Science book on programming languages. | accepted in principle. simple precedence rule will be specified |
| US-004 |  | 5.1 |  | TE | Ambiguous operations with respect to side effects.  For example:  X=17; Y=X++ + X++;  is Y 34 or 35? | See C programming language (ISO/IEC 9899) for guidance on addressing side effects in programming languages. | accepted in principle. editors will study C/C++ cases and add appropriate text. |
| US-005 |  | 5.2 |  | TE | Lexical tokens not defined  No definition of a primitive token.  No definition of lexical token structure. | See C programming language (ISO/IEC 9899) for guidance on describing lexical structure. | accepted in principle. some work has been done in potential improvement. |
| \*\*-006 |  | 5.3 |  | Ed |  | Kindly note that the horizontal lines must be deleted before next submission stage | accepted in principle. Potential improvement already include this. |
| US-007 |  | 5.4 |  | TE | Endianness incompletely defined. It appears that the document presumes big endianness, but also allows little endianness with no mechanism for describing little endianness values. | Add missing text. | accepted. Only BIG ENDIAN will be supported. The 2nd sentence of 5.8 in Potential improvement will be removed. |
| US-008 |  | 5.9,7.1 |  | TE | Broken braced scoping and class definition braces - either broken or ambiguous specification.  The following is incorrect scoping, as described by the document:  {    // start of scope { { // deepest nesting } // end of scope } } // end of scope should be here | Text should talk about balanced or matching braces. Better yet, provide a formal grammar (and lexical description) | accepted in principle. it will be clarified that the scope ends with the 'matching' closing brace. Some other case beyond this rule, e.g. a variable introduced in the nested scope area needs to be used after the closing brace, are mentioned in Potential Improvement. They will be further discussed and implemented to the final text. |
| \*\*-009 |  | 6.2.1 |  | Ed | EXAMPLE 1 | In accordance with [ISO/IEC Directives Part 2, 25.3](https://www.iso.org/sites/directives/current/part2/index.xhtml#_idTextAnchor348): "Within a given clause or subclause, examples shall be numbered sequentially." Kindly amend throughout. | [NR: Addressed. In the new Pot. Impr. version attached the examples have been modified as follows: “Within a given clause or subclause, examples shall be numbered sequentially. The numbering restarts at each new subdivision. A single example in a subdivision shall not be numbered.” |
| US-010 |  | 6.7 |  | TE | UTF-8 and UTF-16 do not have escape mechanism for characters that cannot be typed.  Should use escaping mechanism in C programming language ISO/IEC 9899. | See C (ISO/IEC 9899) for escaping mechanism. | accepted in principle. Several escape codes will be added, eg. \u, \U, \r, etc. to the specification. |
| \*\*-011 |  | 7.6 |  | Ed | must | [**ISO/IEC Directives Part 2, 7.6**](https://www.iso.org/sites/directives/current/part2/index.xhtml#_idTextAnchor095)**:**  Do not use “must” as an alternative for “shall”. | Accepted. |
| US-012 |  | 9.1 |  | TE | Incomplete and ambiguous definition of conditional in if-then | Describe syntax and semantics. | accepted in principle. potential improvement has section on this. more rule will be specified in the final text. |
| US-013 |  | 9.1 |  | ED | Personalized phraseology, i.e., the "We" in "We could equally well have another entity instead" | Re-word text. | accepted in principle |
| US-014 |  | 2, 6.06 |  | TE | Should use ISO/IEC 10646 reference and specification of UTF-8 and UTF-16. Replace with 10646 references | Rewrite to reference ISO/IEC standards. | Accepted in principle. to be crosschecked with other specification, e.g. 14496-12. |
| \*\*-015 |  | Foreword |  | Ed | Foreword | We updated the foreword | accepted |
| US-016 |  | General |  | TE | Document doesn't define semantics, conformance, environment. Don't see how C, C++, and Java can be consolidated into 26 pages. | See below for details | accepted in principle. However, SDL is not a programming language. So, comparison with other programing language may not be reasonable. Scope and/or Introduction section will clarify this. |
| US-017 |  | General |  | TE | The document is littered with incomplete specification, defects, and ambiguities. Easily 50 defect reports could be generated from this document. | Have SC22/WG14 (C) and SC22/WG21 (C++) and SC32/WG2 (Metadata, including 11404 General Purpose Datatypes) review the document. | accepted in principle. However, SDL is not a programming language. So, review by the group developing the standards on programming language may not be needed. |
| US-018 |  | General |  | TE | One cannot define datatypes without also defining characterizing operations (see ISO/IEC 11404 General Purpose Datatypes) upon the data.  For example, there is no definition of the operators and special cases or exceptional conditions, such as negating the smallest negative number for a signed integer datatype, division by zero, temporary values in multiplication (e.g., is 16 bits TIMES 32 bits done in 32, 48, or 64 bit temporary values).  Floating point types are not described in terms for data formats (multiple standards) with accuracy/rounding options.  And how are exceptions handled?  There is no mention of NaN. | Rework and complete definitions of datatypes  Look at ISO/IEC 11404 General Purpose Datatypes and ISO/IEC 10967-\* Language Independent Arithmetic | accepted in principle. Potential Improvement has already specified data types suitable for our purpose. |
| US-019 |  | General |  | TE | There is little understanding of separating lexical description and processing from grammatical (syntactic) description and semantics.  There are no constraints nor operating limitations nor description of the operating environment where this processing occurs.  The document is littered with defects and ambiguities. | Look at C (ISO/IEC 9899) for an example of proper language definition. The C standard is simpler than the C++ standard, so the C standard should be used as guidance. | accepted in principle. However, SDL is not a programming language. Current specification has defined whatever suitable for our use. Scope and Introduction section will further clarify this. Formal grammar will be also added to the final text. |
| US-020 |  | General |  | TE | No conformance paradigm (e.g., conformance roles of code, processor, and processing environment), no description of conformance. | See C programming language (ISO/IEC 9899) and Metadata Interoperability and Buildings (ISO/IEC 20944-1, see conformance paradigms, conformance roles, strictly conforming vs. merely conforming). | accepted in principle. However, SDL is not a programming language. So, conformance similar level as C/C++ does not seem necessary. |
| \*\*-021 |  | Introduction |  | Ed | The International Organization for Standardization (ISO) draws attention to the fact that it is claimed that compliance with this document may involve the use of a patent. | Patent text is now covered by the foreword which states: " As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document" as we have not received any patent declarations for this document | accepted |
| US-022 |  | Introduction |  | TE | Document says patents are declared, but could not find patent listing for 14496-34 (or 14496-1). | Make consistent with patent declarations. | accepted. |