

Current defect report for ISO/IEC 24707:2007 as of **03-Feb-08**.

Note that “Grammar error” refers to an EBNF grammar, not an English syntactical mistake.

Defects that require additional action and/or comment are highlighted in yellow.

Defect no.	Submitter & contact info	Qualifier (error, omission, clarification needed)	References in document	Nature of defect	Proposed solution
24707/001	Pat Hayes phayes@ihmc.us	Typo error	p.3 - 3.8	'3.8' and 'dialect' on same line	'3.8' and 'dialect' should be on separate lines
24707/002	Pat Hayes phayes@ihmc.us	Clarification needed	p.7 - 5.1.1, 2 nd sentence	First sentence uses "shall", second sentence uses "will". It is not clear if this indicates a difference of sense. If it does, this difference should be clarified. If it does not, a single term should be used consistently.	Replace “will” with “shall”.
24707/003	Pat Hayes phayes@ihmc.us	Clarification needed	p.13 - 6.2 1 st sentence	the phrase "mathematical structures" is potentially misleading as it can be construed as being contrasted with other 'kinds' of structure. The qualifier "mathematical" is not necessary to the sense of the sentence, so should be deleted.	Delete “mathematical”
24707/004	Pat Hayes phayes@ihmc.us	Grammar error	p.14 4 th para, 2 nd sentence		Replace “differ with” using “differ from”
24707/005	Pat Hayes phayes@ihmc.us	Typo error	p.18 – 6.4 last sentence	Missing period	Replace “of the entailment” with “of the entailment.”
24707/006	Pat Hayes phayes@ihmc.us	Typo error	p.20 – 6.6.1 last sentence		Boldface “shall”
24707/007	Pat Hayes phayes@ihmc.us	Clarification needed	p.21 – 7.1.2 last sentence of para “Semantic extensions shall...”	“less complete” is ambiguous and misleading	Replace “may be less complete” with “may be incomplete”
24707/008	Pat Hayes phayes@ihmc.us	Technical error	p.16, 2 nd para, 4 th sentence	The qualification "... but shall not use irregular sentence forms to represent content that is expressible in Common Logic text." is unnecessary and potentially harmful to interoperability under certain conditions. It is not mentioned in section 7 defining conformity, so it may be left over from an earlier version by mistake.	Replace “, but shall not use irregular sentence forms to represent content that is expressible in Common Logic text.” With “.”

24707/009	Pat Hayes phayes@ihmc.us	Clarification needed	p.18, 1 st para, last sentence	Sentence seems to contradict the formal semantics	Replace "The module name may be used to identify a common universe of discourse associated with the dialect, or a local universe of discourse special to the text in the module." with "The module name indicates a local universe of discourse special to the text in the module, which can be asserted to be equivalent to a common universe of discourse associated with the dialect by sentences outside the module."
24707/010	Pat Hayes phayes@ihmc.us	Typo error	p.29, A.2.3.11, 2 nd production	Ambiguous production rule	Replace "cltext = { phrase }" with "text = { phrase }"
24707/011	Pat Hayes phayes@ihmc.us	Technical error	p.31	The ordering of the names X1 ... Xn is not specified, and there is no natural way to specify it. Hence this 'mapping' is undefined, and hence the CLIF semantics is undefined, for the guarded quantifier construction. Solution is to remove the 'guarded quantifier' construction from the CLIF specification in Appendix A. This solution was arrived at by a clear consensus among the technical committee which designed the specification. The guarded quantifier construction is currently the subject of on-going research, is not required by the main CL spec and has not been implemented or used by any known CLIF project.	(a) Delete lines 4 and 5 from table A.2 on page 31 (b) In A 2.3.8, page 29: remove the words "and may be guarded;" (c) In A 2.3.8, page 29: remove the option "[interpretablename]" from the first production (d) In A.1, page 24: remove item 11.
24707/012	Pat Hayes phayes@ihmc.us And others	Technical error	Annexes A and B	The CLIF syntax uses several reserved names with an enclosed colon, such as "cl:imports". It has been pointed out by commentators that this is likely to give rise to clashes with a widely used XML convention called Qnaming, whereby a 'root' URI is abbreviated by a short prefix defined in the XML header, which is prefixed to a local name separated by a colon, such as 'rdf:type' being an abbreviation for the full URIreference ' http://www.w3.org/1999/02/22-rdf-syntax-ns#type '. Solution: As this 'internal colon' convention serves no real purpose, simply replace it with hyphen. This does not change the language essentially and would require only minimal changes to any extant CL parsers. Since these 'words' are reserved, it is better to not use up commonly used words such as 'text' or 'comment', so rather than eliminate the 'cl:' prefix entirely, we propose abbreviating it without the comma, producing entirely artificial neologisms.	Throughout Annexes A and B, make the following textual substitutions: 'cl-module' for 'cl:module' 'cl-excludes' for 'cl:excludes' 'cl-imports' for 'cl:imports' 'cl-text' for 'cl:text' 'cl-comment' for 'cl:comment' 'cl-roleset' for 'roleset:' These occur on pages 27 - 31, page 52 (table B.1, line E5, second part), page 53, lines E12, E14, E17, E19, E20.
24707/013	Chris Angus chris.angus@btinternet.com	Omission	p.68 C.2 3.3	Example includes 'TBD' in place of valid XCL elements, thus they are neither correct XCL nor informative	Provide valid example of atomic sentence element, or omit Example section NEED SPECIFIC TEXT CHANGES HERE

24707/014	Chris Angus chris.angus@btinternet.com	Omission	p.68 C.2 3.3.2	Example includes 'TBD' in place of valid XCL elements, thus they are neither correct XCL nor informative	Provide valid example of relation element, or omit Example section NEED SPECIFIC TEXT CHANGES HERE
24707/015	Chris Angus chris.angus@btinternet.com	Omission	p.71 C.2 3.4.2	Example includes 'TBD' in place of valid XCL elements, thus they are neither correct XCL nor informative	Provide valid example of function element, or omit Example section NEED SPECIFIC TEXT CHANGES HERE
24707/016	Chris Angus chris.angus@btinternet.com	Omission	Annex C	Annex C makes no mention of how to represent sequence markers	NEED SPECIFIC TEXT CHANGES HERE
24707/017	Chris Angus chris.angus@btinternet.com	Omission	p.59-60 C.2 2.2	The abstract syntax metamodel explicitly makes importation a subtype of phrase, an importation may therefore form part of a text. The XCL specification does not allow an <import> element as a child of the top level <text> element, it limits it to being a child of a <module> element. This would appear to prevent XCL being syntactically fully conformant.	Allow <import> elements within the content of a <text> element. NEED SPECIFIC TEXT CHANGES HERE
24707/018	Chris Angus chris.angus@btinternet.com	Technical Error	p.59-60	The content for a <module> element is at variance with the abstract syntax metamodel in that it allows multiple <exclude> elements.	Amend the content to be an optional <exclude> element and a <text> element containing the body of the module NEED SPECIFIC TEXT CHANGES HERE
24707/019	Chris Angus chris.angus@btinternet.com	Technical Error	p.65 C.2 3.1.5	The abstract syntax metamodel does not support a Binding having comment associated with it (nor does CLIF), however XCL allows a <var> element to contain zero, one or more comments. This would appear to prevent XCL being syntactically fully conformant.	Either add support for comments being associated with a Binding to the abstract syntax metamodel and CLIF or remove them from XCL. NEED SPECIFIC TEXT CHANGES HERE
24707/020	Chris Angus chris.angus@btinternet.com	omission		XCL allows a <term> element to contain multiple <comment> elements. This conflicts with the CommentedTerm construct in the abstract syntax which allows for a single character string to be associated with the Term being commented. One can argue that the abstract syntax allows CommentedTerms to be nested, however there is no indication of how multiple <comment> elements would map to implicit nested Commented Terms.	NEED SPECIFIC TEXT CHANGES HERE
24707/021	jJohn Sowa (sowa@bestweb.net)	Redundancy	p.48 B.3.4	eitherOr = "[", [comment] ?cm?, "Either", [":"], {[comment], nestedOrs} ?ors?, [endComment] ?ecm?, "]"	Remove "{" and "}" in 2 nd line to read: [comment], nestedOrs ?ors?, [endComment] ?ecm?, "]"

24707/022	John Sowa (sowa@bestweb.net)	Grammar error	p. 44 B.2.11 p. 50 B.3.8	<p>The colon that separates the type field from the referent field of a concept node is usually optional. But in the definition of the category <code>*text*</code>, it is required. There would be no ambiguity if it were optional, and I suggest that the two grammar rules for <code>*text*</code> should contain brackets around ":" .</p> <p>Making the colon optional does not correct any technical error, but it makes the grammar more consistent by allowing colons to be optional in all concept nodes that have a type label.</p>	<p>B.2.11 text</p> <p>The first line of the grammar rule should be</p> <pre>text = "[", [comment], "Proposition", [":"], [CGname], CG,</pre> <p>B.3.8 text</p> <p>The second line of the translation rule should be</p> <pre>text = "[", [comment] ?cm?, "Proposition", [":"], [CGname] ?n?,</pre>
-----------	---------------------------------	---------------	-----------------------------	--	--

24707/023	nicolas.f.rouquette@jpl.nasa.gov	Grammar error	pp. 28-31 A.2.3.2 A.2.3.9 A.2.3.11	<p>CLIF semantics specifies no interpretation for comments which are currently defined in terms of interpreted names (quotedstring) instead of interpretable names (enclosedname)</p> <p>A.2.3.9 affects Table A.1, p.30, the unnamed entry between E12 and E13 which defines the interpretation for an expression of the form:</p> <p>A sentence (cl:comment "string" P) is I(P)</p> <p>This sentence expression is not syntactically valid according to A.2.3.9:</p> <p>commentsent = open, 'cl:comment', quotedstring , sentence , close ;</p> <p>because A.2.2.2 and A.2.2.5 define, respectively:</p> <p>stringquote = ""; quotedstring = stringquote, { ... }, stringquote ;</p> <p>A.2.3.11 affects Table A.1, p.31, the unnamed entry between E14 and E17 which defines the interpretation for an expression of the form:</p> <p>A phrase (cl:comment "string") is: true</p> <p>This sentence expression is not syntactically valid according to A.2.3.11 for similar reason as above:</p> <p>phrase = sentence module (open, 'cl:imports' , interpretablename , close) (open, 'cl:comment', quotedstring, cltext, close);syntax/semantic inconsistency for comment sentences & comment phrases (see clause 10 in: http://www.omg.org/cgi-bin/doc?ad/08-05-02) The syntax of commented sentences (A.2.3.9) and phrases (A.2.3.11) are in conflict with the key principle governing the *non* interpretation of comments in CLIF semantics succinctly stated in 6.2, on p. 16:</p>	<p>A.2.3.9: commentsent = open, 'cl:comment', quotedstring , sentence , close ;</p> <p>It should be changed to a definition that uses an interpretable name instead:</p> <p>A.2.3.9: commentsent = open, 'cl:comment', enclosedname , sentence , close ;</p> <p>Similarly, A.2.3.11 is also inconsistent with the principle stated in 6.2:</p> <p>A.2.3.11: phrase = sentence module (open, 'cl:imports' , interpretablename , close) (open, 'cl:comment', quotedstring, cltext, close);</p> <p>It too should be changed to a definition that uses an interpretable name instead:</p> <p>A.2.3.11: phrase = sentence module (open, 'cl:imports' , interpretablename , close) (open, 'cl:comment', enclosedname [cltext], close);</p>
-----------	----------------------------------	---------------	---	--	--

