

XML exceptions that allow continuation

The following table provides the exception codes that are associated with XML event `EXCEPTION` and that the XML parser returns in special register `XML-CODE` when the parser can continue processing the XML data. That is, the code is within one of the following ranges:

- 1-99
- 100,001-165,535
- 200,001-265,535

The table describes the exception and the actions that the parser takes when you request it to continue after the exception. In these descriptions, the term "XML text" means either `XML-TEXT` or `XML-NTEXT`, depending on whether the XML document that you are parsing is in an alphanumeric or national data item, respectively.

Table 31. XML exceptions that allow continuation

Code	Description	Parser action on continuation
1	The parser found an invalid character while scanning white space outside element content.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
2	The parser found an invalid start of a processing instruction, element, comment, or document type declaration outside element content.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
3	The parser found a duplicate attribute name.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
4	The parser found the markup character '<' in an attribute value.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
5	The start and end tag names of an element did not match.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
6	The parser found an invalid character in element content.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
7	The parser found an invalid start of an element, comment, processing instruction, or CDATA section in element content.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
8	The parser found in element content the CDATA closing character sequence ']]>' without the matching opening character sequence '<![CDATA['.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
9	The parser found an invalid character in a comment.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
10	The parser found in a comment the character sequence '--' (two hyphens) not followed by '>'.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
11	The parser found an invalid character in a processing instruction data segment.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
12	A processing instruction target name was 'xml' in lowercase, uppercase or mixed case.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
13	The parser found an invalid digit in a hexadecimal character reference (of the form <code>&#xddd;</code>).	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
14	The parser found an invalid digit in a decimal character reference (of the form <code>&#ddd;</code>).	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.

Code	Description	Parser action on continuation
15	The encoding declaration value in the XML declaration did not begin with lowercase or uppercase A through Z.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
16	A character reference did not refer to a legal XML character.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
17	The parser found an invalid character in an entity reference name.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
18	The parser found an invalid character in an attribute value.	The parser continues detecting errors until it reaches the end of the document or encounters an error that does not allow continuation. The parser does not signal any further normal events, except for the <code>END-OF-DOCUMENT</code> event.
50	The document was encoded in EBCDIC, and the CCSID of the COBOL source member is a supported EBCDIC CCSID, but the document encoding declaration did not specify a recognizable encoding.	The parser uses the encoding specified by the CCSID of the COBOL source member.
51	The document was encoded in EBCDIC, and the document encoding declaration specified a supported EBCDIC encoding, but the parser does not support the CCSID of the COBOL source member.	The parser uses the encoding specified by the document encoding declaration.
52	The document was encoded in EBCDIC, and the CCSID of the COBOL source member is a supported EBCDIC CCSID, but the document encoding declaration specified an ASCII encoding.	The parser uses the encoding specified by the CCSID of the COBOL source member.
53	The document was encoded in EBCDIC, and the CCSID of the COBOL source member is a supported EBCDIC CCSID, but the document encoding declaration specified a supported Unicode encoding.	The parser uses the encoding specified by the CCSID of the COBOL source member.
54	The document was encoded in EBCDIC, and the CCSID of the COBOL source member is a supported EBCDIC CCSID, but the document encoding declaration specified a Unicode encoding that the parser does not support.	The parser uses the encoding specified by the CCSID of the COBOL source member.
55	The document was encoded in EBCDIC, and the CCSID of the COBOL source member is a supported EBCDIC CCSID, but the document encoding declaration specified an encoding that the parser does not support.	The parser uses the encoding specified by the CCSID of the COBOL source member.
56	The document was encoded in ASCII, and the CCSID of the COBOL source member is a supported ASCII CCSID, but the document encoding declaration did not specify a recognizable encoding.	The parser uses the encoding specified by the CCSID of the COBOL source member.
57	The document was encoded in ASCII, and the document encoding declaration specified a supported ASCII encoding, but the parser does not support the CCSID specified by the CCSID of the COBOL source member.	The parser uses the encoding specified by the document encoding declaration.
58	The document was encoded in ASCII, and the CCSID of the COBOL source member is a supported ASCII CCSID, but the document encoding declaration specified a supported EBCDIC encoding.	The parser uses the encoding specified by the CCSID of the COBOL source member.
59	The document was encoded in ASCII, and the CCSID of the COBOL source member is a supported ASCII CCSID, but the document encoding declaration specified a supported Unicode encoding.	The parser uses the encoding specified by the CCSID of the COBOL source member.
60	The document was encoded in ASCII, and the CCSID of the COBOL source member is a supported ASCII CCSID, but the document encoding declaration specified a Unicode encoding that the parser does not support.	The parser uses the encoding specified by the CCSID of the COBOL source member.
61	The document was encoded in ASCII, and the CCSID of the COBOL source member is a supported ASCII CCSID, but the document encoding declaration specified an encoding that the parser does not support.	The parser uses the encoding specified by the CCSID of the COBOL source member.
100,001 - 165,535	The document was encoded in EBCDIC, and the encodings specified by the CCSID of the COBOL source member and the document encoding declaration are both supported EBCDIC CCSIDs, but are not the same. <code>XML-CODE</code> contains the CCSID for the encoding declaration plus 100,000.	If you set <code>XML-CODE</code> to zero before returning from the <code>EXCEPTION</code> event, the parser uses the encoding specified by the CCSID of the COBOL source member. If you set <code>XML-CODE</code> to the CCSID for the document encoding declaration (by subtracting 100,000), the parser uses this encoding.
200,001 - 265,535	The document was encoded in ASCII, and the encodings specified by the CCSID of the COBOL source member and the document encoding declaration are both supported ASCII CCSIDs, but are not the same. <code>XML-CODE</code> contains the CCSID for the encoding declaration plus 200,000.	If you set <code>XML-CODE</code> to zero before returning from the <code>EXCEPTION</code> event, the parser uses the encoding specified by the CCSID of the COBOL source member. If you set <code>XML-CODE</code> to the CCSID for the document encoding declaration (by subtracting 200,000), the parser uses this encoding.

RELATED TASKS

[Handling errors in XML documents](#)

(C) Copyright IBM Corporation 1992, 2006. All Rights Reserved.