# 混合XML并行解析详细设计文档v1.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 修改内容 | 修改人 | 审核人 | 日期 |
| V1.0 | 初稿制定 | 杨宇 | 杨宇 | 2019-1-15 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1 目的 3](#_Toc535322258)

[2 XML 3](#_Toc535322259)

[2.1 Standard 3](#_Toc535322260)

[2.2 Abstract 4](#_Toc535322261)

[2.3 Dependences 4](#_Toc535322262)

[2.4 Match rule 5](#_Toc535322263)

[3功能需求 6](#_Toc535322264)

[4 架构设计 7](#_Toc535322265)

[4.1 阶段划分 7](#_Toc535322266)

[4.2 数据结构 7](#_Toc535322267)

[5 代码实现 9](#_Toc535322268)

[5.1 开发环境 9](#_Toc535322269)

[5.2 目录结构 10](#_Toc535322270)

[5.3 运行结果 10](#_Toc535322271)

## 1 目的

通过动手敲代码和调试，实现一个完整的可行程序，给定的任何一篇xml文档，程序都能准确的判断其合法性，把理论知识运用到实践过程中，加深对理论知识的理解和达到学以致用的目的。

## 2 XML

### 2.1 Standard

[1]  element   ::=    [EmptyElemTag](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-EmptyElemTag) | [STag](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-STag) [content](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-content) [ETag](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-ETag)

[2]   EmptyElemTag   ::=   '<' [Name](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Name) ([S](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-S) [Attribute](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Attribute))\* [S](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-S)? '/>'

[3]   STag   ::=   '<' [Name](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Name) ([S](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-S) [Attribute](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Attribute))\* [S](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-S)? '>'

[4]   ETag    ::=     '</' [Name](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Name) [S](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-S)? '>'

[5]   content   ::=    [CharData](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CharData)? (([element](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-element) | [Reference](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Reference) | [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect) | [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) | [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment)) [CharData](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CharData)?)\*

[6]   Name   ::=   [NameStartChar](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-NameStartChar) ([NameChar](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-NameChar))\*

[7]   NameStartChar   ::=   ":" | [A-Z] | "\_" | [a-z]

[8]   NameChar   ::=   [NameStartChar](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-NameStartChar) | "-" | "." | [0-9]

[9]   S   ::=   (#x20 | #x9 | #xD | #xA)+

[10]   Attribute   ::=    [Name](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Name) [Eq](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Eq) [AttValue](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-AttValue)

[11] Eq ::= [S](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-S)? '=' [S](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-S)?

[12]   AttValue   ::=   ' " ' ([^<&"] | [Reference](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Reference))\* ' " '| " ' " ([^<&'] | [Reference](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Reference))\* " ' "

[13]   CharData ::=  [^<&]\* - ([^<&]\* ']]>' [^<&]\*)

[14]   Reference   ::=    [EntityRef](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-EntityRef) | [CharRef](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CharRef)

[15] EntityRef   ::=   '&' [Name](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Name) ';'

[16]  CharRef   ::=   '&#' [0-9]+ ';' | '&#x' [0-9a-fA-F]+ ';'

[17]   CDSect   ::=    [CDStart](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDStart) [CData](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CData) [CDEnd](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDEnd)

[18]   CDStart   ::=   '<![CDATA['

[19]   CData   ::=   ([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\* - ([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\* ']]>' [Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\*))

[20]   CDEnd   ::=   ']]>'

[21]   PI   ::=   '<?' [PITarget](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PITarget) ([S](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-S) ([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\* - ([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\* '?>' [Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\*)))? '?>'

[22]  PITarget   ::=    [Name](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Name) - (('X' | 'x') ('M' | 'm') ('L' | 'l'))

[23] Comment   ::=   '<!--' (([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char) - '-') | ('-' ([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char) - '-')))\* '-->'

[24] Char   ::=   #x9 | #xA | #xD | [#x20-#x7F]

### 2.2 Abstract

1. This document contains simplified XML Spec1.0 and dependences among those syntactic constructs.

2. Our work is to parse an XML document in a most paralleling way with FPGA. When parallel processing, flat and multiple rules can be checked in parallel.

3. In order to recognize characters matching the syntactic constructs properly, it is needed to first consider those dependences among the above rules. If a character or a string matching one syntactic construct can also make up another one, we define these two syntactic constructs are dependent on each other. So we pick up all these dependences or potential interrelations to improve parallel processing.

### 2.3 Dependences

1) Tag ‘<’

It can be a part of texts in [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

It is the first character of [STag](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-STag), [ETag](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-ETag), [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

“element” also begins with tag ‘<’.

2) [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment)

These three kinds of elements are very special. The texts in these three can be consisted of any characters in [Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char) except their corresponding closing tags.

Illegal examples:

<![CDATA[ Hello ]]> world!]]>

<?xml version="1.0" ?> encoding="ISO-8859-1" ?>

<!--CDATA[ Hello --> world!-->

<!--CDATA[ Hello, world!--->

3) Tag ‘>’

It can be a part of texts in [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

It is the last character of [STag](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-STag), [ETag](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-ETag), [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

It occurs anywhere of texts in any element except [Name](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Name).

4) Tag ‘</’

It can be a part of texts in [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

It is the first two characters of [ETag](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-ETag).

### 2.4 Match rule

XML documents consist of a lot of tags. The start tags ‘<’, ‘<?’, ‘<!--’, ‘</’, ‘<![CDATA[’ must be in pairs with ‘>’, ‘?>’, ‘-->’, ‘>’, ‘]]>’.

5) Tag ‘<?’

It can be a part of texts in [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

When it is the first two characters of [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI), it occurs in pair with ‘?>’.

Tag ‘<!’

It can be a part of texts in [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

It is the first two characters of [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect) and [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

6) Tag ‘<!--’

It can be a part of texts in [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

When it is the first four characters of [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment), it occurs in pair with ‘-->’.

7) Tag ‘-->’

It can be a part of texts in [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect) or [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI).

It can be the closing tag of [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

8) ‘<![CDATA[’

It can be a part of texts in [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

When it is the first four characters of [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), it occurs in pair with ‘]]>’.

9) ‘]]>’

It can be a part of texts in [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

It can be the closing tag of [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect).

It can not be inside [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect).

It can not occur in [CharData](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CharData).

10) Tag ‘/>’

It can be a part of texts in [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment).

It can be the closing tag of [EmptyElemTag](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-EmptyElemTag).

[CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) or [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment)

These three can also be inside of each other.

11) [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect) and [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI)

When “ '<?' [PITarget](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PITarget) ([S](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-S) ([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\* - ([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\* '?>' [Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\*)))? '?>' “ is a part of [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), it is not Processing Instruction any more and it lose the [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) function because the text in a [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect) will not be parsed by a parse.

e.g. <![CDATA[ Hello World! <?xml version="1.0"?> ]]>

“[CDStart](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDStart) [CData](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CData) [CDEnd](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDEnd)” can be a part of [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI), it is still [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect) inside [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI).

e.g. <?xml version="1.0" <![CDATA[SSPKU]]> ?>

12) [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) and [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment)

“  '<!--' (([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char) - '-') | ('-' ([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char) - '-')))\* '-->' “ can be a part of [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI), it is still [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment) inside [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI).

e.g. <?xml version="1.0" <!--encoding="ISO-8859-1-->"?>

But when “ '<?' [PITarget](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PITarget) ([S](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-S) ([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\* - ([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\* '?>' [Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char)\*)))? '?>' “ is a part of [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment), it is not Processing Instruction any more and it lose the [PI](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-PI) function.

e.g. <!-- Hello World! <?xml version="1.0"?> -->

13) [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect) and [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment)

When “[CDStart](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDStart) [CData](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CData) [CDEnd](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDEnd)” is a part of [Comment](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Comment), it is not CDATA Section any more and it lose the [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect) function.

e.g. <!-- Hello World! <![CDATA[SSPKU]]>  -->

When “  '<!--' (([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char) - '-') | ('-' ([Char](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Char) - '-')))\* '-->' “ is a part of [CDSect](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-CDSect), it will not be parsed by a parse.

e.g. <![CDATA[ Hello World! <!--xml --> ]]>

14) [Attribute](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Attribute)

 ' " ' ([^<&"] | [Reference](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Reference))\* ' " '| " ' " ([^<&'] | [Reference](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-Reference))\* " ' " can be a part of [AttValue](file:///E:\%E7%A7%91%E7%A0%94\%E5%8F%AF%E9%87%8D%E6%9E%84%E8%AE%A1%E7%AE%97%E7%A0%94%E7%A9%B6\XML%E7%A0%94%E7%A9%B6\%E6%96%87%E6%A1%A3\Extensible%20Markup%20Language%20%28XML%29%201.0%20%28Fifth%20Edition%29.htm#NT-AttValue) itself.

e.g. CarNo.=”PKU99 CarNo.= ‘PKU99’ ”

15) '<' 可以出现在PI CDSect, Comment的内容中，或者作为Stag, Etag, PI, CDSect, Comment首字符，或者作为element开始

16) PI, CDSect, Comment的内容可以包含除了他们结束符以外的任何字符

< > , …….

17) '>' 可以出现在PI, CDSect, Comment内容中，或者作为STag, ETag, PI, CDSect, Comment结尾字符

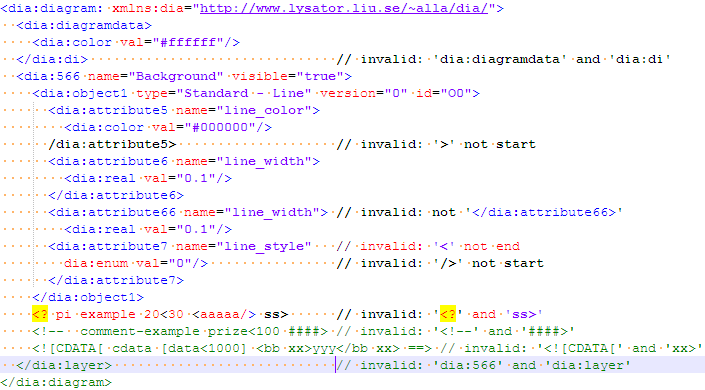
除了name 其他任何元素的内容都可以出现

17) '</' 可以出现在PI, CDSect, Comment内容中，或者作为Etag的前两个字符

## 3功能需求

根据第2节的内容，编出来的程序需要能够识别一切不合法的匹配并指出具体位置，举几种不符合要求的情形，如：





## 4 架构设计

### 4.1 阶段划分

1. 阶段一：开辟一个线程循环读取XML文件，一次读取一个字符，记录 ’<’ 在文档中的位置和类型，并记录下紧跟在 ’<’ 后面的 ‘>’ 的位置与 ’<’ 存储到同一个START\_INFO中，每读完一个START\_INFO后将其放入BLOCK\_INFO中，并把stage1\_finish置为1，在这个过程中需要根据每块的容量限制START\_INFO的个数.
2. 阶段二：开辟多个线程从vector<BLOCK\_INFO> ldata中获取块进行解析，对于BLOCK\_INFO中能够被检测到START\_INFO则从块中删除，未能检测到的继续保留。
3. 阶段三： 阶段二中所有块均已解析完成后，BLOCK\_INFO中只剩下未能被解析的START\_INFO，然后将所有块内的START\_INFO放到一个新块中，使用一个线程解析新块的合法性，即为最终结构。

### 4.2 数据结构

**enum START\_TYPE (记录‘<‘的类型)**

数据成员：

|  |  |  |  |
| --- | --- | --- | --- |
| StagorEmptytag\_start | 1 | int | <dia:color val="#ffffff"/>  <dia:layer name="Background" > |
| Etag\_start | 2 | int | </dia:layer> |
| PI\_start | ~~3~~ | int | ~~<? pi example 20<30 ?>~~ |
| CDSECT\_start | 4 | int | <![CDATA[ cdata section [data<1000] ]]> |
| COMMENT\_start | 5 | int | <!-- comment-example prize<100 --> |
| INVALID | -1 | int | ‘<’ 即非内容本身也非前面几种 |

**struct START\_INFO (存储’<’的位置和属性及紧跟其后’>’的位置)**

数据成员：

|  |  |  |
| --- | --- | --- |
| stype | START\_TYPE |  |
| fileoffset | int | ‘<’ 在文件中的位置 |
| blocknum | int |  |
| blockpos | int |  |
| endpos | int | 紧跟’<’后的’>’在文件中位置 |
| attr | String | ‘<’ 的属性字段 |

主要的函数成员：

|  |  |
| --- | --- |
| START\_INFO(const START\_TYPE type, const int offset = 0, const int bnum = 0, const int bpos = 0, const int end = 0)  :stype(type)  ,fileoffset(offset)  ,blocknum(bnum)  ,blockpos(bpos)  ,endpos(end)  ,attr("")  { } | 构造函数 |

**struct BLOCK\_INFO (存储START\_INFO，每块的容量可自行设定)**

数据成员：

|  |  |  |
| --- | --- | --- |
| stage1\_start | bool | 阶段一已开始 |
| stage1\_finish | bool | 阶段一已结束 |
| Stage2\_start | bool |  |
| Stage2\_ finish | bool |  |
| Stage3\_start | bool |  |
| Stage3\_ finish | bool |  |
| info\_list | deque<START\_INFO> | 块 |

主要的函数成员：

|  |  |
| --- | --- |
| BLOCK\_INFO(bool s1 = false, bool f1 = false, bool s2 = false, bool f2 = false, bool s3 = false, bool f3 = false)  :stage1\_start(s1)  ,stage1\_finish(f1)  ,stage2\_start(s2)  ,stage2\_finish(f2)  ,stage3\_start(s3)  ,stage3\_finish(f3)  {} | 构造函数 |
| void clear()  {  stage1\_start = false;  stage1\_finish = false;  stage2\_start = false;  stage2\_finish = false;  stage3\_start = false;  stage3\_finish = false;  info\_list.clear();  } | 清空块 |
| ~BLOCK\_INFO()  {  info\_list.clear();  } | 析构函数 |

**class CXml (对应与xml文件总的数据结构，包含所有块)**

数据成员：

|  |  |  |
| --- | --- | --- |
| sfile | String | Xml文件 |
| block\_cap | bool | 块容量 |
| ldata | vector<BLOCK\_INFO> | 所有块 |
| commentAndCdata; | stack<START\_INFO> | 处理评论和注释合法性 |
| threads | vector<shared\_ptr<Thread>> | 线程容器 |

主要的函数成员：

|  |  |
| --- | --- |
| CXml(string file, int th = 2, int blockcap = 1 << 13)  :sfile(file)  ,th\_num(th)  ,block\_cap(blockcap)  { } | 构造函数 |
| void start() | 创建线程并启动 |
| void buildData() | 阶段一、三的线程函数 |
| bool checkValid() | 阶段二的线程函数 |

## 5 代码实现

### 5.1 开发环境

Linux: Centos7.x

G++: gcc version 4.8.5 20150623 (Red Hat 4.8.5-28) (GCC)

Vim: version 7.4.1099

### 5.2 目录结构

repository: [git@github.com:icoty/xml\_parser.git](mailto:git@github.com:icoty/xml_parser.git)

user: 杨宇

.

|-- debug.xml // 非法xml用于调试

|-- Makefile

|-- README.md

|-- runs.sh // sh run.sh自动编译执行

|-- test1.xml // 合法xml

|-- test2.xml // 合法xml

|-- test3.xml // 合法xml

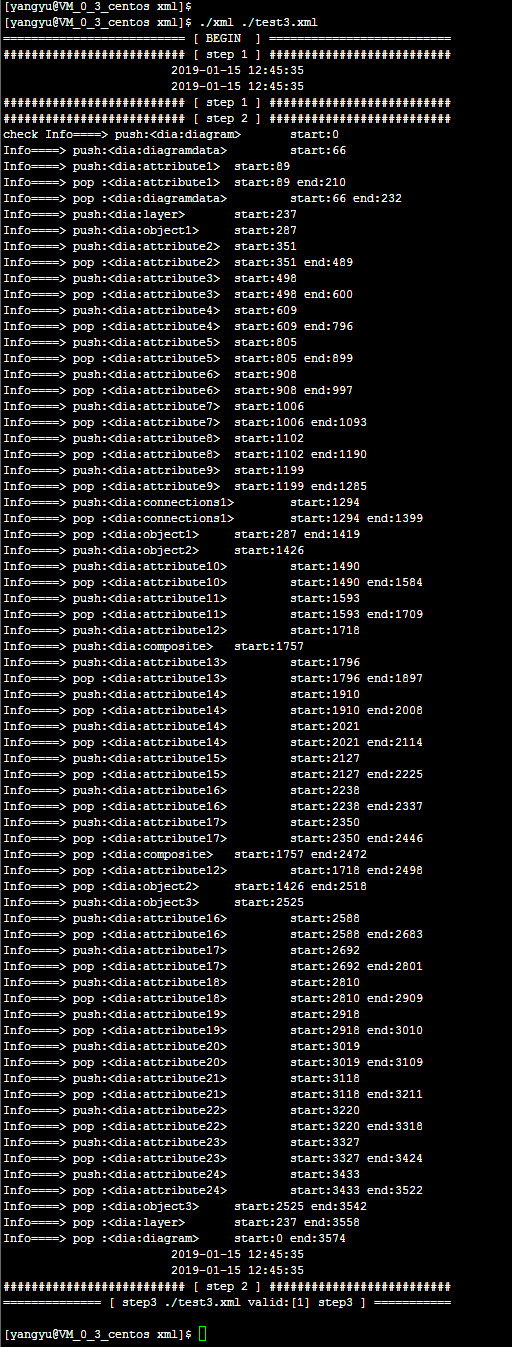
|-- xml // 可执行文件

|-- Xml\_1thread.cpp // 单线程

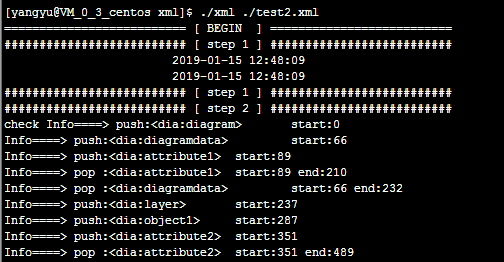
|-- Xml\_Nthread.cpp // 多线程

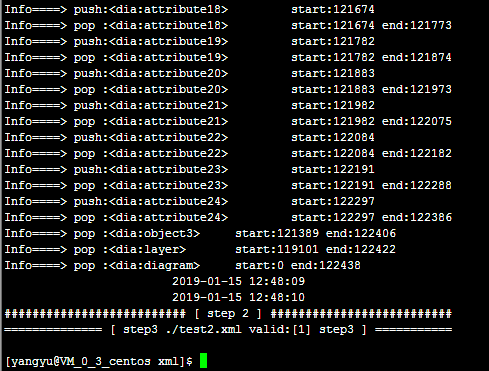
### 5.3 运行结果

test3.xml检测结果为真：

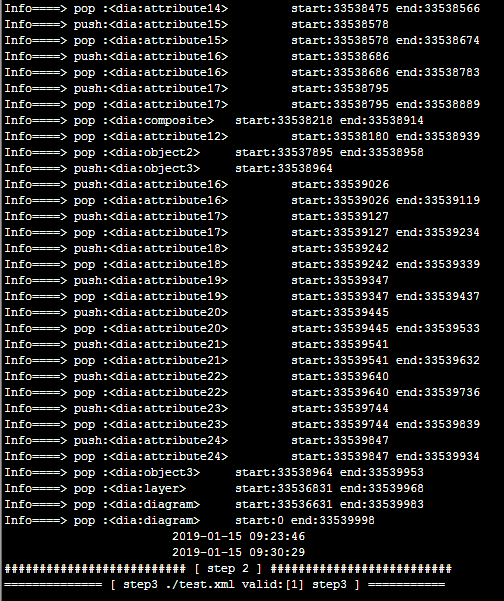


test2.xml检测结果为真，耗时1秒：



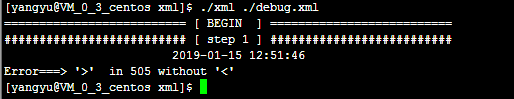


test.xml检测结果为真：



对于debug.txt非法文件可直接检测非法位置，请修改后缀名为.xml后测试：





**附录：**

**单线程Xml\_1thread.cpp**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <sys/types.h>

#include <unistd.h>

#include <memory>

#include <string>

#include <time.h>

#include <functional>

#include <pthread.h>

#include <iostream>

#include <deque>

#include <map>

#include <stack>

#include <vector>

using namespace std;

string getTime()

{

time\_t timep;

time (&timep);

char tmp[64];

strftime(tmp, sizeof(tmp), "%Y-%m-%d %H:%M:%S",localtime(&timep) );

return tmp;

}

enum START\_TYPE

{

StagorEmptytag\_start = 1,

Etag\_start,

PI\_start,

CDSECT\_start,

COMMENT\_start,

INVALID = -1

};

struct START\_INFO

{

START\_TYPE stype;

int fileoffset;

int blocknum;

int blockpos;

int endpos;

string attr;

//拷贝构造函数

START\_INFO(const START\_INFO & c)

{

stype=c.stype;

fileoffset = c.fileoffset;

blocknum = c.blocknum;

blockpos = c.blockpos;

endpos = c.endpos;

attr = c.attr;

}

START\_INFO(const START\_TYPE type, const int offset = 0, const int bnum = 0, const int bpos = 0, const int end = 0)

:stype(type)

,fileoffset(offset)

,blocknum(bnum)

,blockpos(bpos)

,endpos(end)

,attr("")

{ }

};

struct block\_info{

bool stage1\_start;

bool stage1\_finish;

bool stage2\_start;

bool stage2\_finish;

bool stage3\_start;

bool stage3\_finish;

vector<START\_INFO> startInfo\_array;

};

class CXml{

public:

CXml(string file)

:sfile(file)

{ }

void buildData()

{

cout << "========================== [ BEGIN ] ==========================" << endl;

cout << "########################## [ step 1 ] ##########################" << endl;

cout << "\t\t\t" << getTime() << endl;

FILE \*fpr;

if((fpr=fopen(sfile.c\_str(), "rb+")) == NULL)

{

cout << "open " << sfile << " error!" << endl;

exit(-1);

}

char ch = '\0';

while((ch = fgetc(fpr)) != EOF)

{

if('>' == ch) // 形如 <!--- <test pattern="SECAM" /><test pattern="NTSC" /> --> 需防止被覆盖

{

int end = ftell(fpr) - 1;

fseek(fpr, -3, SEEK\_CUR);

string stmp(3, 0);

fread((void\*)&stmp.c\_str()[0], sizeof(char), 3, fpr);

ch = fgetc(fpr);

if(' ' == stmp[1] || (ch >= '0' && ch <= '9')) //内容本身的 >

continue;

if(stmp == "]]>" || stmp == "-->")

{

if(commentAndCdata.empty() || (stmp == "]]>" && commentAndCdata.top().stype != CDSECT\_start) || (stmp == "-->" && commentAndCdata.top().stype != COMMENT\_start))

{

cout << "Error===> [" << sfile << "]:" << stmp << " in " << end << " invalid match CDSECT\_start or CDSECT\_start" << endl;

exit(-1);

}

commentAndCdata.pop();

continue;

}

if(ldata.empty() || (stmp.substr(1) == "?>" && ldata.back().stype != PI\_start) || (stmp[1] == '/' && ldata.back().stype != StagorEmptytag\_start))

{

cout << "Error===> [" << sfile << "]:'>' " << " in " << end << " without '<'" << endl;

exit(-1);

}

if(stmp.substr(1) != "?>" && stmp[1] != '/')

{

bool invalid = false;

if(ldata.back().stype != StagorEmptytag\_start && ldata.back().stype != Etag\_start){

invalid = true;

}else{

if(0 != ldata.back().endpos)

invalid = true;

}

if(invalid){

cout << "Error===> [" << sfile << "]:'>' " << " in " << end << " without '<'" << endl;

exit(-1);

}

}

if(0 != ldata.back().endpos){

cout << "Error===> [" << sfile << "]:'>' " << " in " << end << " without '<'" << endl;

exit(-1);

}

ldata.back().endpos = end;

//cout << ldata.back().fileoffset << "-" << ldata.back().endpos<< " ";

}

if('<' == ch)

{

int offset = ftell(fpr) - 1;

if(' ' == (ch = fgetc(fpr)) || (ch >= '0' && ch <= '9')) //内容本身的 <

continue;

START\_INFO tmp(START\_TYPE::INVALID);

tmp.fileoffset = offset;

tmp.blocknum = 0;

tmp.blockpos = offset;

bool s\_comment = false;

switch(ch)

{

case '/':

tmp.stype = Etag\_start;

break;

case '?':

tmp.stype = PI\_start;

break;

case '!':

{

string stmp(8, 0);

int ret = fread((void\*)&stmp.c\_str()[0], sizeof(char), 8, fpr);

if("--" == stmp.substr(0, 2))

{

tmp.stype = COMMENT\_start;

}else if("[CDATA[" == stmp.substr(0, 7)){

tmp.stype = CDSECT\_start;

}else{

cout << "Error===> [" << sfile << "]:Invalid string:" << stmp << " in pos:" << offset << endl;

exit(-1);

}

s\_comment = true;

commentAndCdata.push(tmp);

//fseek(fpr, -5, 1);

break;

}

default:

tmp.stype = StagorEmptytag\_start;

break;

}

if(s\_comment)

continue;

ldata.push\_back(tmp);

}

}

if(!commentAndCdata.empty())

{

cout << "Error===> [" << sfile << "]:CDSECT\_start or CDSECT\_start in " << commentAndCdata.top().fileoffset << " invalid match" << endl;

exit(-1);

}

cout << "\t\t\t" << getTime() << endl;

cout << "########################## [ step 1 ] ##########################" << endl;

fclose(fpr);

}

bool checkValid(int ith = 0)

{

cout << "########################## [ step 2 ] ##########################" << endl;

string step2\_start = getTime();

FILE \*fpr;

if((fpr=fopen(sfile.c\_str(), "r+")) == NULL)

{

cout << "Error===> open " << sfile << "error!" << endl;

exit(-1);

}

cout << "check ";

stack<START\_INFO> st;

for(int i = 0; i < ldata.size(); ++i)

{

if(CDSECT\_start != ldata[i].stype && COMMENT\_start != ldata[i].stype && 0 == ldata[i].endpos) // 形如 <![CDATA[ < >xx</ > < /> ]]> 或 <!--- <test pattern="SECAM" /><test pattern="NTSC" /> -->

{

cout << "Error===> [" << sfile << "]:Invalid format, '<' in " << ldata[i].fileoffset << " without '>'" << endl;

return false;

}

if((CDSECT\_start == ldata[i].stype || COMMENT\_start == ldata[i].stype) && 0 == ldata[i].endpos)

{

if(CDSECT\_start == ldata[i].stype){

cout << "Info====> push: <![CDATA[\t start: " << ldata[i].fileoffset << endl;

}else{

cout << "Info====> push: <!--\t start: " << ldata[i].fileoffset << endl;

}

st.push(ldata[i]);

continue;

}

if(!st.empty() && (CDSECT\_start == st.top().stype || COMMENT\_start == st.top().stype))

{

fseek(fpr, ldata[i].endpos + 1, SEEK\_SET); // 如果栈顶有评论和注释,则需要寻找它们的结束字段

char ch = '\0';

while((ch = fgetc(fpr)) != EOF){

if('<' == ch){

if(' ' == (ch = fgetc(fpr)) || (ch >= '0' && ch <= '9')) //内容本身的 <

continue;

else

break;

}else if('>' == ch){

int end = ftell(fpr) - 1;

fseek(fpr, -3, SEEK\_CUR);

string stmp(4, 0);

fread((void\*)&stmp.c\_str()[0], sizeof(char), 4, fpr);

if(' ' == stmp[1] || (stmp[3] >= '0' && stmp[3] <= '9')) //内容本身的 >

continue;

if((CDSECT\_start == st.top().stype && "]]>" == stmp.substr(0, stmp.size())) || (COMMENT\_start == st.top().stype && "-->" == stmp.substr(0, stmp.size())))

{

if(CDSECT\_start == ldata[i].stype){

cout << "Info====> pop :<![CDATA[\t start:" << ldata[i].fileoffset << " end:" << end << endl;

}else{

cout << "Info====> pop :<!--\t start:" << ldata[i].fileoffset << " end:" << end << endl;

}

st.pop();

break;

}

cout << "Error===> [" << sfile << "]:CDSECT\_start or COMMENT\_start in " << st.top().fileoffset << " not match end: " << end << endl;

exit(-1);

}

}

}

fseek(fpr, ldata[i].endpos - 2, SEEK\_SET); // 定位至 '>' 左边二位, 读取形如 ]]> --> ?> 等

string stmp(3, 0);

fread((void\*)&stmp.c\_str()[0], sizeof(char), 3, fpr);

if(StagorEmptytag\_start == ldata[i].stype && '/' == stmp[stmp.size() - 2])

continue; // 形如 <dia:point val="1.95,6.85"/> 为空元素,直接跳过,不用入栈

if(PI\_start == ldata[i].stype && '?' == stmp[stmp.size() - 2])

continue; // 形如 <?xml version="1.0" encoding="UTF-8"?>

if(CDSECT\_start == ldata[i].stype && "]]>" == stmp)

continue;

if(COMMENT\_start == ldata[i].stype && "-->" == stmp)

continue;

fseek(fpr, ldata[i].fileoffset, SEEK\_SET); // 定位至 '<'

string att = "";

char ch = '\0';

do{

ch = fgetc(fpr);

att.append(1, ch); // 截取属性,形如 <dia:diagramdata> 或 <dia:layer name="Background"

}while(' ' != ch && '>' != ch);

att[att.size() - 1] = '>';

ldata[i].attr = att;

if(Etag\_start == ldata[i].stype)

{

START\_INFO tp = st.top();

string start = tp.attr.substr(1, tp.attr.size() - 1);

if(st.empty() || (StagorEmptytag\_start != tp.stype) || (StagorEmptytag\_start == tp.stype && start != att.substr(2, att.size() - 2)))

{

cout << "Error===> [" << sfile << "]:Invalid format, " << att << " in " << ldata[i].fileoffset << " without '<'" << endl;

return false;

}

cout << "Info====> pop :" << tp.attr << "\t start:" << tp.fileoffset << " end:" << ldata[i].endpos << endl;

st.pop();

continue;

}

if(st.empty() || StagorEmptytag\_start == ldata[i].stype)

{

cout << "Info====> push:" << ldata[i].attr << "\t start:" << ldata[i].fileoffset << endl;

st.push(ldata[i]);

}else{

cout << "Debug====> " << ldata[i].stype << endl;

}

}

cout << "\t\t\t" << step2\_start << endl;

cout << "\t\t\t" << getTime() << endl;

cout << "########################## [ step 2 ] ##########################" << endl;

return st.empty();

}

private:

string sfile;

stack<START\_INFO> commentAndCdata;

vector<START\_INFO> ldata;

};

int main(int argc, char\* argv[])

{

if(argc < 2)

cout << "invalid parameter, please input your file: argv[1]" << endl;

CXml myxml(argv[1]);

myxml.buildData();

cout << "============== [ step3 " << argv[1] << " valid:[" << myxml.checkValid() << "] step3 ] ===========" << endl << endl;

return 0;

}

**多线程Xml\_Nthread.cpp**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <sys/types.h>

#include <unistd.h>

#include <memory>

#include <unistd.h>

#include <pthread.h>

#include <string>

#include <time.h>

#include <functional>

#include <pthread.h>

#include <iostream>

#include <deque>

#include <map>

#include <stack>

#include <vector>

#include <list>

using namespace std;

string getTime()

{

time\_t timep;

time (&timep);

char tmp[64];

strftime(tmp, sizeof(tmp), "%Y-%m-%d %H:%M:%S",localtime(&timep) );

return tmp;

}

const int BLOCK\_SIZE = 1 << 20;

// 禁止赋值构造与拷贝构造

class Noncopyable{

protected:

Noncopyable(){ }

~Noncopyable(){ }

private:

Noncopyable(const Noncopyable &);

Noncopyable & operator=(const Noncopyable &);

};

// 锁

class MutexLock{

public:

MutexLock()

:\_islocking(false)

{

pthread\_mutex\_init(&\_mutex,NULL);

}

~MutexLock()

{ pthread\_mutex\_destroy(&\_mutex); }

void lock()

{

pthread\_mutex\_lock(&\_mutex);

\_islocking=true;

}

void unlock()

{

pthread\_mutex\_unlock(&\_mutex);

\_islocking=false;

}

bool status()const

{ return \_islocking; }

pthread\_mutex\_t \* getMutexLockPtr()

{ return & \_mutex; }

private:

MutexLock(const MutexLock&);

MutexLock& operator=(const MutexLock&);

private:

bool \_islocking;

pthread\_mutex\_t \_mutex;

};

// 智能管理锁

class MutexLockGuard{

public:

MutexLockGuard(MutexLock &mutex)

:\_mutex(mutex)

{

\_mutex.lock();

}

~MutexLockGuard()

{

\_mutex.unlock();

}

private:

MutexLock &\_mutex;

};

// 线程类

class Thread:private Noncopyable{

typedef function<void()> ThreadCallback;

public:

Thread(ThreadCallback cb)

:\_pthId(0)

,\_isRunning(false)

,\_cb(cb)

{ }

~Thread()

{

cout<<"Thread::~Thread()"<<endl;

if(\_isRunning){

pthread\_detach(\_pthId);//资源回收交给主线程进行托管

\_isRunning=false;

}

}

void start()

{

pthread\_create(&\_pthId,NULL,threadFunc,this);

\_isRunning = true;

}

void join()

{

if(\_isRunning){

pthread\_join(\_pthId,NULL);

\_isRunning = false;

}

}

static void \*threadFunc(void \*arg)

{

Thread\* p=static\_cast<Thread\*>(arg);

cout << (bool)p;

if(p){

cout << "\_cb";

p->\_cb();

}

return NULL;

}

pthread\_t getId()

{

return pthread\_self();

}

private:

pthread\_t \_pthId;

bool \_isRunning;

ThreadCallback \_cb;

};

enum START\_TYPE

{

StagorEmptytag\_start = 1,

Etag\_start,

PI\_start,

CDSECT\_start,

COMMENT\_start,

INVALID = -1

};

struct START\_INFO

{

START\_TYPE stype;

int fileoffset;

int blocknum;

int blockpos;

int endpos;

string attr;

//拷贝构造函数

START\_INFO(const START\_INFO & c)

{

stype=c.stype;

fileoffset = c.fileoffset;

blocknum = c.blocknum;

blockpos = c.blockpos;

endpos = c.endpos;

attr = c.attr;

}

START\_INFO(const START\_TYPE type, const int offset = 0, const int bnum = 0, const int bpos = 0, const int end = 0)

:stype(type)

,fileoffset(offset)

,blocknum(bnum)

,blockpos(bpos)

,endpos(end)

,attr("")

{ }

};

struct BLOCK\_INFO{

bool stage1\_start;

bool stage1\_finish;

bool stage2\_start;

bool stage2\_finish;

bool stage3\_start;

bool stage3\_finish;

deque<START\_INFO> info\_list;

BLOCK\_INFO(bool s1 = false, bool f1 = false, bool s2 = false, bool f2 = false, bool s3 = false, bool f3 = false)

:stage1\_start(s1)

,stage1\_finish(f1)

,stage2\_start(s2)

,stage2\_finish(f2)

,stage3\_start(s3)

,stage3\_finish(f3)

{}

~BLOCK\_INFO()

{

info\_list.clear();

}

void clear()

{

stage1\_start = false;

stage1\_finish = false;

stage2\_start = false;

stage2\_finish = false;

stage3\_start = false;

stage3\_finish = false;

info\_list.clear();

}

};

class CXml: public enable\_shared\_from\_this<CXml>{

public:

CXml(string file, int th = 2, int blockcap = 1 << 13) // 每块容量限制4096 Bytes

:sfile(file)

,th\_num(th)

,block\_cap(blockcap)

{ }

void start()

{

shared\_ptr<Thread> sp(new Thread(bind(&CXml::buildData, this)));

threads.push\_back(sp);

sp->start();

for(int i = 0; i < th\_num - 1; ++i){

shared\_ptr<Thread> th2(new Thread(bind(&CXml::checkValid, this)));

threads.push\_back(sp);

th2->start();

}

}

void buildData()

{

cout << "========================== [ BEGIN ] ==========================" << endl;

cout << "########################## [ step 1 ] ##########################" << endl;

cout << "\t\t\t" << getTime() << endl;

FILE \*fpr;

if((fpr=fopen(sfile.c\_str(), "rb+")) == NULL)

{

cout << "open " << sfile << " error!" << endl;

exit(-1);

}

while(1)

{

BLOCK\_INFO oneblock(true, false, false, false, false, false);

char ch = '\0';

while((ch = fgetc(fpr)) != EOF)

{

if('>' == ch) // 形如 <!--- <test pattern="SECAM" /><test pattern="NTSC" /> --> 需防止被覆盖

{

int end = ftell(fpr) - 1;

fseek(fpr, -3, SEEK\_CUR);

string stmp(3, 0);

fread((void\*)&stmp.c\_str()[0], sizeof(char), 3, fpr);

ch = fgetc(fpr);

if(' ' == stmp[1] || (ch >= '0' && ch <= '9')) //内容本身的 >

continue;

if(stmp == "]]>" || stmp == "-->")

{

if(commentAndCdata.empty() || (stmp == "]]>" && commentAndCdata.top().stype != CDSECT\_start) || (stmp == "-->" && commentAndCdata.top().stype != COMMENT\_start))

{

cout << "Error===> " << stmp << " in " << end << " invalid match CDSECT\_start or CDSECT\_start" << endl;

exit(-1);

}

commentAndCdata.pop();

continue;

}

if(oneblock.info\_list.empty() || (stmp.substr(1) == "?>" && oneblock.info\_list.back().stype != PI\_start) || (stmp[1] == '/' && oneblock.info\_list.back().stype != StagorEmptytag\_start))

{

cout << "Error===> '>' " << " in " << end << " without '<'" << endl;

exit(-1);

}

if(stmp.substr(1) != "?>" && stmp[1] != '/')

{

bool invalid = false;

if(oneblock.info\_list.back().stype != StagorEmptytag\_start && oneblock.info\_list.back().stype != Etag\_start){

invalid = true;

}else{

if(0 != oneblock.info\_list.back().endpos)

invalid = true;

}

if(invalid){

cout << "Error===> '>' " << " in " << end << " without '<'" << endl;

exit(-1);

}

}

if(0 != oneblock.info\_list.back().endpos){

cout << "Error===> '>' " << " in " << end << " without '<'" << endl;

exit(-1);

}

oneblock.info\_list.back().endpos = end;

cout << oneblock.info\_list.back().fileoffset << "-" << oneblock.info\_list.back().endpos<< " ";

}

if('<' == ch)

{

int offset = ftell(fpr) - 1;

if(offset >= block\_cap)

{

oneblock.stage1\_finish = true;

ldata.push\_back(oneblock);

oneblock.clear();

}

if(' ' == (ch = fgetc(fpr)) || (ch >= '0' && ch <= '9')) //内容本身的 <

continue;

START\_INFO tmp(START\_TYPE::INVALID);

tmp.fileoffset = offset;

tmp.blocknum = 0;

tmp.blockpos = offset;

bool s\_comment = false;

switch(ch)

{

case '/':

tmp.stype = Etag\_start;

break;

case '?':

tmp.stype = PI\_start;

break;

case '!':

{

string stmp(8, 0);

int ret = fread((void\*)&stmp.c\_str()[0], sizeof(char), 8, fpr);

if("--" == stmp.substr(0, 2))

{

tmp.stype = COMMENT\_start;

}else if("[CDATA[" == stmp.substr(0, 7)){

tmp.stype = CDSECT\_start;

}else{

cout << "Error===> Invalid string:" << stmp << " in pos:" << offset << endl;

exit(-1);

}

s\_comment = true;

commentAndCdata.push(tmp);

break;

}

default:

tmp.stype = StagorEmptytag\_start;

break;

}

if(s\_comment)

continue;

oneblock.info\_list.push\_back(tmp);

}

}

//cout << "444";

if(!commentAndCdata.empty())

{

cout << "Error===> CDSECT\_start or CDSECT\_start in " << commentAndCdata.top().fileoffset << " invalid match" << endl;

exit(-1);

}

}

cout << "555";

cout << "\t\t\t" << getTime() << endl;

cout << "########################## [ step 1 ] ##########################" << endl;

fclose(fpr);

}

bool checkValid()

{

cout << "########################## [ step 2 ] ##########################" << endl;

string step2\_start = getTime();

FILE \*fpr;

if((fpr=fopen(sfile.c\_str(), "r+")) == NULL)

{

cout << "Error===> open " << sfile << "error!" << endl;

exit(-1);

}

stack<START\_INFO> st;

while(1)

{

#if 1

int pos = -1;

{

MutexLockGuard(MutexLock());

for(int i = 0; i < ldata.size(); ++i)

{

if(ldata[i].stage1\_finish && !ldata[i].stage2\_start){

pos = i;

ldata[pos].stage2\_start = true;

}

}

}

if(-1 == pos){

sleep(1);

continue;

}

for(int i = 0; i < ldata[pos].info\_list.size(); ++i)

{

if(CDSECT\_start != ldata[pos].info\_list[i].stype && COMMENT\_start != ldata[pos].info\_list[i].stype && 0 == ldata[pos].info\_list[i].endpos) // 形如 <![CDATA[ < >xx</ > < /> ]]> 或 <!--- <test pattern="SECAM" /><test pattern="NTSC" /> -->

{

cout << "Error===> Invalid format, '<' in " << ldata[pos].info\_list[i].fileoffset << " without '>'" << endl;

return false;

}

// 可删

if((CDSECT\_start == ldata[pos].info\_list[i].stype || COMMENT\_start == ldata[pos].info\_list[i].stype) && 0 == ldata[pos].info\_list[i].endpos)

{

if(CDSECT\_start == ldata[pos].info\_list[i].stype){

cout << "Info====> push: <![CDATA[\t start: " << ldata[pos].info\_list[i].fileoffset << endl;

}else{

cout << "Info====> push: <!--\t start: " << ldata[pos].info\_list[i].fileoffset << endl;

}

st.push(ldata[pos].info\_list[i]);

continue;

}

//可删

if(!st.empty() && (CDSECT\_start == st.top().stype || COMMENT\_start == st.top().stype))

{

fseek(fpr, ldata[pos].info\_list[i].endpos + 1, SEEK\_SET); // 如果栈顶有评论和注释,则需要寻找它们的结束字段

char ch = '\0';

while((ch = fgetc(fpr)) != EOF){

if('<' == ch){

if(' ' == (ch = fgetc(fpr)) || (ch >= '0' && ch <= '9')) //内容本身的 <

continue;

else

break;

}else if('>' == ch){

int end = ftell(fpr) - 1;

fseek(fpr, -3, SEEK\_CUR);

string stmp(4, 0);

fread((void\*)&stmp.c\_str()[0], sizeof(char), 4, fpr);

if(' ' == stmp[1] || (stmp[3] >= '0' && stmp[3] <= '9')) //内容本身的 >

continue;

if((CDSECT\_start == st.top().stype && "]]>" == stmp.substr(0, stmp.size())) || (COMMENT\_start == st.top().stype && "-->" == stmp.substr(0, stmp.size())))

{

if(CDSECT\_start == ldata[pos].info\_list[i].stype){

cout << "Info====> pop :<![CDATA[\t start:" << ldata[pos].info\_list[i].fileoffset << " end:" << end << endl;

}else{

cout << "Info====> pop :<!--\t start:" << ldata[pos].info\_list[i].fileoffset << " end:" << end << endl;

}

st.pop();

break;

}

cout << "Error===> CDSECT\_start or COMMENT\_start in " << st.top().fileoffset << " not match end: " << end << endl;

exit(-1);

}

}

}

fseek(fpr, ldata[pos].info\_list[i].endpos - 2, SEEK\_SET); // 定位至 '>' 左边二位, 读取形如 ]]> --> ?> 等

string stmp(3, 0);

fread((void\*)&stmp.c\_str()[0], sizeof(char), 3, fpr);

if(StagorEmptytag\_start == ldata[pos].info\_list[i].stype && '/' == stmp[stmp.size() - 2])

continue; // 形如 <dia:point val="1.95,6.85"/> 为空元素,直接跳过,不用入栈

if(PI\_start == ldata[pos].info\_list[i].stype && '?' == stmp[stmp.size() - 2])

continue; // 形如 <?xml version="1.0" encoding="UTF-8"?>

// 可删

if(CDSECT\_start == ldata[pos].info\_list[i].stype && "]]>" == stmp)

continue;

if(COMMENT\_start == ldata[pos].info\_list[i].stype && "-->" == stmp)

continue;

fseek(fpr, ldata[pos].info\_list[i].fileoffset, SEEK\_SET); // 定位至 '<'

string att = "";

char ch = '\0';

do{

ch = fgetc(fpr);

att.append(1, ch); // 截取属性,形如 <dia:diagramdata> 或 <dia:layer name="Background"

}while(' ' != ch && '>' != ch);

att[att.size() - 1] = '>';

ldata[pos].info\_list[i].attr = att;

if(Etag\_start == ldata[pos].info\_list[i].stype)

{

START\_INFO tp = st.top();

string start = tp.attr.substr(1, tp.attr.size() - 1);

if(st.empty() || (StagorEmptytag\_start != tp.stype) || (StagorEmptytag\_start == tp.stype && start != att.substr(2, att.size() - 2)))

{

cout << "Error===> Invalid format, " << att << " in " << ldata[pos].info\_list[i].fileoffset << " without '<'" << endl;

return false;

}

cout << "Info====> pop :" << tp.attr << "\t start:" << tp.fileoffset << " end:" << ldata[pos].info\_list[i].endpos << endl;

st.pop();

continue;

}

if(st.empty() || StagorEmptytag\_start == ldata[pos].info\_list[i].stype)

{

cout << "Info====> push:" << ldata[pos].info\_list[i].attr << "\t start:" << ldata[pos].info\_list[i].fileoffset << endl;

st.push(ldata[pos].info\_list[i]);

}else{

cout << "Debug====> " << ldata[pos].info\_list[i].stype << endl;

}

}

#endif

}

cout << "\t\t\t" << step2\_start << endl;

cout << "\t\t\t" << getTime() << endl;

cout << "########################## [ step 2 ] ##########################" << endl;

return st.empty();

}

private:

string sfile;

int th\_num;

int block\_cap;

stack<START\_INFO> commentAndCdata;

vector<BLOCK\_INFO> ldata;

vector<shared\_ptr<Thread>> threads;

};

int main(int argc, char\* argv[])

{

if(argc < 2)

cout << "invalid parameter, please input your file: argv[1]" << endl;

CXml xml(argv[1]);

xml.start();

return 0;

}