**说 明 书 摘 要**

本发明实施例提供了XML数据解析和生成方法，通过创建先进后出的栈空间，来控制XML数据的解析和创建。在解析时，在开始标签出现时将数据入栈，在结束标签出现时将数据出栈，和出栈的标签比对以及解析，并存储到数据链表上。在生成时，在输出元素开始标签时数据入栈，在该元素所有的子元素都输出后，标签出栈，输出匹配的结束标签，以形成一个完成的元素。上述方法解析和生成方法提高了XML数据的处理效率。

**权 利 要 求 书**

１、一种XML数据的解析方法，所述XML数据包含结构化的元素，包括：

从XML数据中读取一个或多个字符；

针对每次读取的所述字符，将所述字符和预定义的字符进行比对；

如果确定所述字符为所述XML数据的一个元素的起始符号，将所述元素存储到数据链表的一个数据单元上，并将所述元素的标识符存储到栈空间，所述栈空间为预申请的预定大小的存储单元；以及

如果确定所述字符为第一节点的第一结束符号，则将所述第一节点和所述栈空间的栈顶数据进行比较，如果所述第一节点和所述栈顶数据不匹配，则所述XML数据解析失败，如果所述第一节点和所述栈顶数据匹配，则所述栈顶数据出栈。

2、根据权利要求１所述的解析方法，其中，所述将所述元素存储到数据链表的一个数据单元上包括：

为所述元素在所述数据链表上申请第一数据单元；

获取所述元素的标签、标签值以及结点类型；以及

将所述标签、标签值以及结点类型存储到所述第一数据单元。

3、根据权利要求2所述的解析方法，其中，所述将所述元素存储到数据链表还包括：

确定所述元素是否包含属性字段；

针对包含属性字段的所述元素，为每个属性字段在所述数据链表上申请第二数据单元，获取所述属性字段的属性名称和属性值，并将所述属性字段、属性值和结点类型存储到所述第二数据单元。

4、根据权利要求2或3所述的解析方法，其中，所述结点类型包括根结点、子结点和属性结点。

5、根据权利要求3所述的解析方法，还包括：针对包含属性字段的所述元素，如果确定所述字符为第二结束符号，将所述栈顶数据出栈。

6、根据权利要求2所述的解析方法，其中，所述将所述元素的标识符存储到栈空间包括:将所述标签或所述第一数据单元的地址存储到所述栈空间。

7、根据权利要求1所述的解析方法，还包括：所述从XML数据中读取一个或多个字符之前，将所述XML数据读取到计算机内存中。

8、根据权利要求1所述的解析办法，还包括：在解析失败或成功后，释放所述栈空间。

9、一种根据数据链表生成XML数据的方法，所述数据链表的每个数据单元对应所述XML数据的一个元素或属性字段，包括：

读取所述数据链表的一个元素；

针对每次读取的的元素，将所述元素的标识符存储到栈空间，并按照第一格式输出所述元素的标签和标签值，所述栈空间为预申请的预定深度的栈空间；

从栈空间读取栈数据；

针对每次读取的栈数据，在确定所述元素的所有子元素都输出后，输出所述元素的结束标签，并将栈顶数据出栈。

10、根据权利要求9所述的方法，还包括：针对每次读取的属性字段，按照第二格式输出所述属性字段。

11、根据权利要求9所述的方法，其中，所述按照第一格式输出所述元素的标签和标签值包括：

判断所述元素是否包括标签值；

判断所述元素是否包括子元素；以及

根据是否包含标签值和子元素不同的组合输出所述元素的开始标签、标签值和结束标签。

12、根据权利要求9所述的方法，其中，所述将所述元素的标识符存储到栈空间包括:将所述元素的标签或者所述元素对应的数据单元的地址存储到所述栈空间。

13、根据权利要求9所述的解析办法，还包括：在解析失败或成功后，释放所述栈空间。

14、一种XML数据处理系统，包括：

XML数据解析模块，用于从XML数据中获取元素和属性字段，并将所述元素和属性字段分别存储到数据链表的一个数据单元上，其中，通过将所述元素的标识符的入栈出栈控制所述元素的解析，在所述元素的起始符号出现时，将所述标识符入栈，在所述元素的结束符号出现时，将所述标识符出栈；

XML数据生成模块，用于根据结点类型将所述数据链表上的数据单元生成XML数据的元素或属性字段，其中，通过所述数据单元的标识符的入栈出栈控制所述元素的解析，针对每个结点类型的元素，将所述数据单元的标识符入栈，在确定所述元素的所有子元素都输出后，将所述数据单元的标识符出栈。

**说 明 书**

**XML数据解析方法、生成方法以及处理系统**

**技术领域**

本发明涉及到移动终端系统，具体涉及一种避免移动终端的显示屏MIPI干扰的方法及装置。

**背景技术**

可扩展标记语言（XML）是用于表示数据的万维网协会（W3C）标准。XML数据包含形成层次的结构化元素。许多应用程序被设计成支持XML格式数据的输入输出。

在XML中，称为元素的数据项由开始标签和结束标签定界。元素还可包含在元素的开始标签中规定的属性。元素的标签之间的文本作为标签值，可表示诸如串、日期或整数的任意类型的数据值。元素还可具有一个或更多个子元素和兄弟元素。除了根元素之外，每个元素都具有父亲元素。下图是一个XML格式数据的示例。

<?xml version="1.0" encoding="UTF-8"?>

<recipe>

<recipename name=”Ice Cream Sundae” />

<preptime >5 minutes</preptime>

<listitem>

<quantity>3</quantity>

<itemdescription>chocolate syrup or chocolate fudge</itemdescription>

</listitem>

</recipe>

在上图中，根元素<recipe>包含子元素<recipename>、<preptime>和<listitem>,子元素<listitem>包含子元素<quantity>、<itemdescription>，其中<recipename>、<preptime>和<listitem>由于是并列关系，互为兄弟元素，根元素<recipe>则为他们的父亲元素。在构造XML数据时，为每个元素设计开始标签、结束标签、标签值，还可以包含属性字段，如子元素<recipename>具有的属性字段name=”Ice Cream Sundae”。其中子元素也可以不包含标签值，如子元素<recipename>只包含属性字段，通过/>作为结束符号。

XML格式数据在表现具有层次结构关系的数据项方面应用广泛，从而使处理XML数据的应用接口成为各类编程语言的必需。DOM（Document Object Model，对象文档模型）是W3C组织推荐的处理可扩展置标语言的标准编程接口。DOM可以以一种独立于平台和语言的方式访问和修改一个文档的内容和结构。但对于大文件的XML数据，使用DOM加载的速度并不理想。另外DOM只提供封装好的有限接口，不支持自定义的内存池操作，也不能按照实际需要定制生成特定格式的XML字符串或文档。

**发明内容**

有鉴于此，本发明提供一种XML数据的解析和生成方法，以解决上述问题。

根据本发明的一个方面，提供一种XML数据的解析方法，所述XML数据包含结构化的元素，包括：从XML数据中读取一个或多个字符；针对每次读取的所述字符，将所述字符和预定义的字符进行比对；如果确定所述字符为所述XML数据的一个元素的起始符号，将所述元素存储到数据链表的一个数据单元上，并将所述元素的标识符存储到栈空间，所述栈空间为预申请的预定大小的存储单元；以及如果确定所述字符为第一节点的第一结束符号，则将所述第一节点和所述栈空间的栈顶数据进行比较，如果所述第一节点和所述栈顶数据不匹配，则所述XML数据解析失败，如果所述第一节点和所述栈顶数据匹配，则所述栈顶数据出栈。

优选地，所述将所述元素存储到数据链表的一个数据单元上包括：为所述元素在所述数据链表上申请第一数据单元；

获取所述元素的标签、标签值以及结点类型；以及将所述标签、标签值以及结点类型存储到所述第一数据单元。

优选地，所述将所述元素存储到数据链表还包括：确定所述元素是否包含属性字段；针对包含属性字段的所述元素，为每个属性字段在所述数据链表上申请第二数据单元，获取所述属性字段的属性名称和属性值，并将所述属性字段、属性值和结点类型存储到所述第二数据单元。

优选地，所述结点类型包括根结点、子结点和属性结点。

优选地，还包括：针对包含属性字段的所述元素，如果确定所述字符为第二结束符号，将所述栈顶数据出栈。

优选地，所述将所述元素的标识符存储到栈空间包括:将所述标签或所述第一数据单元的地址存储到所述栈空间。

优选地，还包括：所述从XML数据中读取一个或多个字符之前，将所述XML数据读取到计算机内存中。

优选地，还包括：在解析失败或成功后，释放所述栈空间。

根据本发明的另一个方面，提供一种根据数据链表生成XML数据的方法，所述数据链表的每个数据单元对应所述XML数据的一个元素或属性字段，包括：读取所述数据链表的一个元素；针对每次读取的的元素，将所述元素的标识符存储到栈空间，并按照第一格式输出所述元素的标签和标签值，所述栈空间为预申请的预定深度的栈空间；从栈空间读取栈数据；针对每次读取的栈数据，在确定所述元素的所有子元素都输出后，输出所述元素的结束标签，并将栈顶数据出栈。

优选地，还包括：针对每次读取的属性字段，按照第二格式输出所述属性字段。

优选地，所述按照第一格式输出所述元素的标签和标签值包括：判断所述元素是否包括标签值；判断所述元素是否包括子元素；以及根据是否包含标签值和子元素不同的组合输出所述元素的开始标签、标签值和结束标签。

优选地，所述将所述元素的标识符存储到栈空间包括:将所述元素的标签或者所述元素对应的数据单元的地址存储到所述栈空间。

优选地，还包括：在解析失败或成功后，释放所述栈空间。

根据本发明的第三方面，提供一种XML数据处理系统，包括：XML数据解析模块，用于从XML数据中获取元素和属性字段，并将所述元素和属性字段分别存储到数据链表的一个数据单元上，其中，通过将所述元素的标识符的入栈出栈控制所述元素的解析，在所述元素的起始符号出现时，将所述标识符入栈，在所述元素的结束符号出现时，将所述标识符出栈；

XML数据生成模块，用于根据结点类型将所述数据链表上的数据单元生成XML数据的元素或属性字段，其中，通过所述数据单元的标识符的入栈出栈控制所述元素的解析，针对每个结点类型的元素，将所述数据单元的标识符入栈，在确定所述元素的所有子元素都输出后，将所述数据单元的标识符出栈。

本发明实施例提供了XML数据解析和生成方法，通过创建先进后出的栈空间，来控制XML数据的解析和创建。在解析时，在开始标签出现时将数据入栈，在结束标签出现时将数据出栈，和出栈的标签比对以及解析，并存储到数据链表上。在生成时，在输出元素开始标签时数据入栈，在该元素所有的子元素都输出后，标签出栈，输出匹配的结束标签，以形成一个完成的元素。上述方法解析和生成方法提高了XML数据的处理效率。

**附图说明**

通过参照以下附图对本发明实施例的描述，本发明的上述以及其它目的、特征和优点将更为清楚，在附图中：

图1是本发明实施例的XML数据的解析方法的流程图；

图2是本发明另一实施例的XML数据的解析方法的流程图；

图3是本发明实施中的优选的数据链表的示意图；

图4是本发明实施中的栈空间的示意图；

图5是本发明实施例的根据数据链表生成XML数据的方法的流程图；

图6是本发明实施例的按照第一格式输出所述元素的开始标签、标签值和结束标签的流程图；

图7是本发明实施例的XML数据处理系统的结构图。

**具体实施方式**

以下基于实施例对本发明进行描述，但是本发明并不仅仅限于这些实施例。在下文对本发明的细节描述中，详尽描述了一些特定的细节部分。对本领域技术人员来说没有这些细节部分的描述也可以完全理解本发明。为了避免混淆本发明的实质，公知的方法、过程、流程没有详细叙述。另外附图不一定是按比例绘制的。

根据本发明实施例的一个方面，利用元素标识符的栈进栈出控制XML数据解析。在XML数据中，父亲元素的开始标签出现在儿子元素的开始标签之前，结束标签出现在儿子元素的技术标签之后，如果通过栈空间存储标签，在开始标签出现时存储，在结束标签出现时和栈顶的开始标签比对，匹配的情况下说明XML格式正常，元素解析成功，可以接着解析下一个元素，如果不匹配，说明XML格式错误，解析失败。

图1是本发明实施例的XML数据的解析方法的流程图。如图1所述的XML数据解析方法包括步骤101-步骤112。

在步骤101中，申请预定深度的栈空间。在本步骤中，调用mem\_alloc\_cb\_t函数申请预定大小的存储空间作为在本实施例中使用的栈空间。在使用栈空间时，遵循先进后出的原则，每次数据入栈时将数据推送到栈空间的栈顶，每次数据出栈时，栈顶数据先出栈。

在步骤102中，读取XML数据的一个或多个字符。XML数据可以是保存到计算机内存中的XML字符串或XML文件。在一个优选的实施例中，首先将XML文件加载到计算机内存中，在计算机内存中读取XML数据的字符。这样做能提高XML数据的解析速度。

在步骤103中，将字符和预定字符比对。在本步骤中，将步骤102中读取的字符串和<XXX和</XXX字符串比对，其中XXX表示不定长度的大小写的英文字符。

在步骤104中，开始符号“<XXX” 。

在步骤105中，结束符号“</XXX” 。

在背景技术中提到，每个XML数据具有开始标签和结束标签作为一个元素的分界。在步骤103-105中，通过和预定字符比较，判断当前字符是否为一个元素的开始标签<XXX>或结束标签</XXX>。

在步骤106中，解析该元素，将该元素存储到链表的数据单元上。在本步骤中，继续读取XML数据，获得该元素的相关数据并存储到数据链表上。在一个优选的实施例中，将元素存储到数据链表上包括:为元素在数据链表上申请一个数据单元；获取元素的标签、标签值以及结点类型；将标签和标签值以及数据类型存储到数据单元。其中，结点类型定义为根结点、子结点和属性结点。

在步骤107中，判断栈空间为空。如果栈空间为空，说明105中的结束符号</XXX没有对应的开始符号<XXX，所以栈为空时，程序异常退出，即步骤111，在栈不为空时，执行步骤109。

在步骤108中，将该元素的标识符推送到栈空间。其中，元素的标识符用来标识一个元素，推送到栈空间的标识符用于在步骤109中进行比对。该标识符可以是一个元素的标签或元素对应的数据单元的地址。

在步骤109中，取出栈顶数据的标签和当前元素的比对，是否匹配。如前所述，栈空间保存的是元素标识符，可以是元素标签或者存储元素数据单元的地址，如果存储的是数据单元地址，根据数据单元地址获取元素的标签和当前元素的标签比对，确定匹配，执行步骤110，否则执行步骤111。

在步骤110中，把栈顶数据从栈中弹出。表示一个元素已经匹配成功，可以将该元素的标识符出栈。

在步骤111中，异常退出。

在步骤112中，数据读取结束，释放栈空间。XML数据读取完毕，释放资源，程序正常结束。

图1所述的XML数据的解析方法，从XML数据中循环读取字符，每次读取的字符都判断是否是开始符号或结束符号，如果是开始符号，则解析元素，并将元素的标识符存储到栈空间，如果是结束符号，则和栈顶的标识符比对进行判断，确定匹配时，表明XML数据的一个元素解析成功，可以解析下一个元素，否则提示数据格式错误，程序退出。

图2是本发明另一实施例的XML数据的解析方法的流程图。图2所述的实施例包括图1的步骤101-110，且增加了对具有属性字段的元素的处理步骤201-204。由于附图的篇幅所限，在图2上只示出了图1的相关步骤101-108。

相关步骤101-108和图1的步骤相同，这里就不再赘述。

在步骤201中，判断该元素是否存在属性字段。在步骤104中，元素标志“<XXX ”，为元素的起始符号，在本步骤中，进一步判断该元素是否具有属性字段。如果具有属性字段，执行步骤202，否则执行步骤203。

在步骤202中，提取属性字段的名称和值，存储到数据链表。

在步骤203中，判断结束符号“/>”存在。如果存在，执行步骤204，否则执行步骤106。

在步骤204中，将栈顶数据弹出。

步骤201-步骤204处理属性字段如<recipename name=”Ice Cream Sundae” />,从中提取属性名”name”和属性值”Ice Cream Sundae”并存储到数据链表的一个数据单元上。“/>”符号是具有属性字段的元素的结束符号，当此符号出现时，表示当前元素结束，因此将该元素的标识符从栈空间弹出。

上述解析方法中申请的栈空间在解析结束之后释放，以免造成内存泄漏。

在上述XML数据解析方法中，XML数据链表用以存储结构化的元素。图3是给出了数据链表存储的示意图框图。如图3所示，A1-A4，A21-A22,A211为数据链表上的一个数据单元，每个数据单元对应XML数据一个元素或属性，A1-A2为兄弟元素，A21和A22为A2的子元素，A22为A21的兄弟元素，A211为A21的子元素。

图4是本发明实施中的栈空间的示意图，存储XML数据的元素标签。

在一个优选的实施例中，数据链表上的每个数据单元的结构定义如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **结构体名** | | xml\_node\_t | | |
| **序号** | **变量名** | **类型** | **长度** | **描述** |
| 01 | name | str\_t（char \*） | 1 | 标签名 |
| 02 | value | str\_t（char \*） | 1 | 标签值 |
| 03 | type | xml\_node\_type\_e | 1 | 结点类型 |
| 04 | next | xml\_node\_t \* | 1 | 兄弟结点 |
| 05 | child | xml\_node\_t \* | 1 | 孩子结点头(孩子/属性结点) |
| 06 | tail | xml\_node\_t \* | 1 | 孩子结点尾(孩子/属性结点) |
| 07 | parent | xml\_node\_t \* | 1 | 父亲结点 |
| 08 | flag | unsigned int | 1 | 记录结点是否有孩子、属性、结点值。 |
| 09 | temp | xml\_node\_t \* | 1 | 临时指针:遍历XML树时，提高效率(其他情况下，此指针值无效) |

每个数据单元对应一个元素，结点类型（type）表明该元素是属性字段、根结点和子结点，兄弟结点(next)存储第一个兄弟元素的地址，孩子结点头（child）存储第一个子元素或第一个属性元素的地址，和孩子结点尾（tail）存储最后一个子元素或最后一个属性元素的地址。

根据本发明实施例的另一个方面，通过链表生成XML字符串。利用元素标签或数据单元的地址进栈出栈控制XML数据的生成。栈处理思路比DOM处理虽然更加复杂，但其拥有更高的处理效率。

XML字符串格式包括1）有层次的XML数据和2）无层次的XML数据两种：

1）用回车换行符、空格字符、TAB字符等组织标签构建XML树

<ABC>

<DEF>asdasdljfasd</DEF>

<EFG>

<GDK>asdasd</GDK>

</EFG>

</ABC>

2）无层次XML:不适用回车换行符、空格字符、TAB字符等组织标签构建XML树，如下所示：

<ABC><DEF>asdasdljfasd</DEF><EFG><GDK>asdasd</GDK></EFG></ABC>

相应的，根据本发明第二方面，根据数据链表生成XML字符串。从数据链表中读取每个元素，按照一定格式输出元素的开始标签和标签值，并将元素的标识符（元素的标签或数据单元地址）入栈，对于每个元素，在元素的所有子元素都输出后，输出元素的结束标签，并将栈顶数据出栈。

图5是本发明实施例的根据数据链表生成XML字符串（有层次）的方法的流程图，包括步骤501-515。

在步骤501中，申请预定深度的栈空间。在本步骤中，调用mem\_alloc\_cb\_t函数申请预定大小的存储空间作为在本实施例中使用的栈空间。在解析失败或成功后，释放所述栈空间。

在步骤502中，读取数据链表的一个元素。在本步骤中，顺序读取数据链表的每个元素。

在步骤503中，元素入栈。在本步骤中，将元素的标识符存储到栈空间。

在步骤504中，打印结点名<name。

在步骤505中，是否有属性结点。判断是否有属性结点，如果有属性结点，执行步骤506，如果没有属性结点，执行步骤508。

在步骤506中，读取数据链表的属性结点。

在步骤507中，输出属性attr=”value” 。

在步骤508中，根据标签值和子元素的组合判断输出标签和标签值。本步骤的一个优选流程如图6所示，在步骤601-603中，判断是否具有标签值和子元素，根据不同判断的组合，分别执行步骤604-607步骤，按照一定格式打印标签和标签值。

在步骤509中，判断是否有子元素结束。如果有子元素结束，执行步骤510，否则执行步骤502。

在步骤510中，弹出栈顶数据。

在步骤511中，打印结束标签</name>\n。

在步骤512中，判断栈是否为空。如果栈为空，执行步骤513，否则执行步骤502。

在步骤513中，是否有兄弟元素。如果有兄弟元素，执行步骤513，否则执行步骤502。

在步骤514中，父亲元素出栈。

在步骤515中，有孩子结点。

在步骤516中，打印结束标签</name>\n。

根据本发明的第三方面，提供一种XML数据处理系统，包括：

XML数据解析模块601接收外部的XML数据，从XML数据中获取元素和属性字段，并将所述元素和属性字段分别存储到数据链表的一个数据单元上，其中，通过将所述元素的标识符的入栈出栈控制所述元素的解析，在所述元素的起始符号出现时，将所述标识符入栈，在所述元素的结束符号出现时，将所述标识符出栈；

XML数据生成模块602根据结点类型将所述数据链表上的数据单元生成XML数据的元素或属性字段，其中，通过所述数据单元的标识符的入栈出栈控制所述元素的解析，针对每个结点类型的元素，将所述数据单元的标识符入栈，在确定所述元素的所有子元素都输出后，将所述数据单元的标识符出栈。

本发明实施例提供了XML数据解析和生成方法，通过创建先进后出的栈空间，来控制XML数据的解析和创建。在解析时，在开始标签出现时将数据入栈，在结束标签出现时将数据出栈，和出栈的标签比对以及解析，并存储到数据链表上。在生成时，在输出元素开始标签时数据入栈，在该元素所有的子元素都输出后，标签出栈，输出匹配的结束标签，以形成一个完成的元素。上述方法解析和生成方法提高了XML数据的处理效率。

本领域技术人员而言，显然本发明不限于上述示范性实施例的细节，而且在不背离本发明的精神或基本特征的情况下，能够以其他的具体形式实现本发明。例如，在实际应用中，可以不同的需要将上述模块功能划分为和本发明实施例不同的功能结构，或将本发明实施例中的几个功能模块合并和分解成不同的功能结构。因此，无论从哪一点来看，均应将实施例看作是示范性的，而且是非限制性的，本发明的范围由所附权利要求而不是上述说明限定，因此旨在将落在权利要求的等同要件的含义和范围内的所有变化涵括在本发明内。不应将权利要求中的任何附图标记视为限制所涉及的权利要求。此外，显然“包括”一词不排除其他单元或步骤，单数不排除复数。系统权利要求中陈述的多个单元或装置也可以由一个单元或装置通过软件或者硬件来实现。

以上所述仅为本发明的优选实施例，并不用于限制本发明，对于本领域技术人员而言，本发明可以有各种改动和变化。凡在本发明的精神和原理之内所作的任何修改、等同替换、改进等，均应包含在本发明的保护范围之内。