EML格式解析及其访问实现

王倩倩,严莉莉,张燕平
(安徽大学 计算智能与信号处理重点实验室,安徽 合肥 230039)

摘 要:EML格式是各类电子邮件软件中所支持的一种通用格式,遵循RFC822及其后续扩展。文中对EML格式做了细致的分析,在此基础上,使用VC作为开发工具,实现了Windows平台下对EML文件中各类相关信息的读取和解码。所实现的CMail类应用于实际的项目开发中,取得了比较好的效果,能够满足一般用户的需求。用户也可根据自己的需求增加相应的处理函数,同时文中的分析过程对其它系统平台下的类似需求具有一定的指导作用。

关键词:EML;多用途互联网邮件扩展;RFC2046;Base64
中图分类号:TP393.098 **文献标识码**:A **文章编号**:1673-629X(2007)07-0067-03

Analysis and Access Implementation of EML

WANG Qian-qian, YAN Li-li, ZHANG Yan-ping
(Ministry of Education Key Lab. of Intelligent Computing & Signal Processing, Anhui Univ., Hefei 230039, China)

Abstract: EML is a kind of general format supported by all kinds of Email software and this format follows the standard of RFC822 and its extensions. Analyzes the format of EML and then implements how to read and decode the information in EML under Windows platform with VC6. Design a class named CMail and use it in the development of project. Through test, it achieves a good effect and can satisfy the needs of most users. Users can also add some functions according to their own needs. The developers who have similar needs in other platform can benefit from the analysis in this paper.

Key words: EML; MIME; RFC2046; Base64

0 引 言

EML格式是微软公司在Outlook中所使用的一种遵循RFC822及其后续扩展的文件格式,并成为各类电子邮件软件的通用格式。深入了解EML格式,是进行邮件客户端设计、垃圾邮件分析过滤等关于电子邮件方面开发和研究的必要前提。文中对EML格式做了细致的分析,在VC下设计实现了一个对EML格式进行分析处理的类并应用于实际的项目开发中,取得了理想的效果。

1 邮件结构

为使电子邮件在各种网络和服务器间正常地发送和接收,人们对电子邮件的格式进行了规定。最初的标准RFC822[1]规定了电子邮件的一些基本规范,但只能用于传输文本信息,无法满足用户的需求和网络技术的发展。因此人们对电子邮件格式陆续增添了新的内容,即MIME(多用途互联网邮件扩展),其基本内容定义于RFC2045~2049,文中主要讨论RFC2045[2]和RFC2046[3]。

1.1 邮件类结构分析和总体设计

RFC中定义的邮件结构包括两个部分,即邮件头和邮件体,两者由一个空行隔开。邮件头包括几个部分:主题、创建者、收信人以及邮件创建日期等。为传递多媒体信息,邮件头又增加了邮件体内容类型、邮件体内容传输编码方式等。邮件体部分包括正文和附件,结构较复杂,将在第三部分中详细介绍。

根据以上分析,可以设计一个CMail类,用于打开、分析和处理EML文件,并返回用户所需的信息,文中所涉及到的一些主要成员函数如下:

```
bool OpenMail()//读入邮件文件
CString GetFrom()//取得邮件创建者
CString GetSubject()//取得邮件主题
CString GetContent()//取得邮件内容
```

收稿日期:2006-09-18
基金项目:"九七三"计划国家重点基础研究(2004CB318108);国家自然科学基金资助项目(60475017,60135010);安徽省自然科学基金资助项目(050420208)
作者简介:王倩倩(1982-),女,安徽六安人,硕士研究生,研究方向为计算智能;张燕平,教授,硕士生导师,研究方向为人工神经网络、智能算法及其应用。

CString Base64_Decode(CString s)//Base64 编码的解码

CString GetBoundary(CString s)//读取邮件文本的边界定界符

CString GetTransferCoding(CString s)//获取邮件体传输的编码方式

CString GetContentType(CString s)//获取邮件体内容的类型

1.2 邮件头相关函数的实现

邮件头中各字段由一或多行文字组成,多行字段的附加行以一个空格作为开始。各字段由字段名、可选的空格、冒号、可选的注解空格、可选的字体段组成。如某 Content-Type 字段描述为:Content-Type: multipart/mixed; boundary="==========0041883896=="。

实现中将邮件读入一个 CString 字符串中。对于邮件头部的字段,使用 CString 提供的 Find 方法查找各字段的位置,从该字段的冒号开始直到该字段结束为该字段的字段体部分。字段结束标志为找到 CRLF 结束符且下一行不以空格开始。例如对邮件头的 Subject 字段:

```
int where=0;
while(Mail.Find("Subject:",where)! =-1);
{if(Mail.GetAt(where-1)='\n')break;
    else where++;
}//查找 Mail 中 Subject 位置,且查到的 Subject 是一行的开头
int end=where;//end 为 Subject 字段结束的位置
while((Mail.GetAt(end)! ='\r')&&(Mail.GetAt(end+2)! =' ')//字段结束标志
end++;
CString subject=Mail.Mid(where+8,end-where-8);//subject 为所需的字段体
```

字段体内容可能是纯文本也可能进行了编码,如 Subject 字段,其字段体部分可能是如下形式:"=?GB2312?B?fM3GvPa49rrcsru07bXEsK5fx+lfzfi4+MTjLS1ieSC67tLSsv0=?=",此时需要进行解码。字段体中,以"=?"开始,以"?="结束,中间以"?"分隔[4]。第一部分指明字符集,如 GB2312,第二部分为编码方式,本例中的 B 代表 Base64 编码,最后一部分为具体的主题内容。根据前一步取出的字段体内容,通过在其字符串中查找"?"将几部分分别取出,根据其提供的字符集和编码方式,使用后文所述的解码函数获得文本内容。

其它与邮件头相关的函数都可以使用类似的方法实现。

1.3 邮件体相关函数的实现

MIME 中的邮件体部分比早期的单文本文件复杂很多,实现时要根据其 Content-Type 类型决定具体访问方法。Content-Type 一般包括文本的类型、文本使用的字符集等,文本为复合类型时还包括分隔不同部分的分界字符串(boundary)。

RFC2046 定义 Content-Type 顶层有 5 种离散类型和两种复合类型。离散类型是 text(文本信息),image(图像信息),audio(音频信息),video(视频信息),application(其它信息)。复合类型是 multipart(多部分信息)和 message(压缩信息),顶层类型又有其子类型。正常邮件中,正文文本多是 text 类型,其它类型大多作为附件方式存在。由于邮件中可能多种类型并存,且不同复合类型会相互嵌套,造成了邮件体分析时的复杂情况。

1)当邮件体中只出现某种离散的顶层类型时。比如邮件的 Content-Type 类型为 Text/plain[5],Text 是顶层类型,说明邮件体为文本类型,plain 是 Text 的一种子类型:纯文本类型。只需要读取邮件头部分的 Content-transfer-Encoding 字段,来判断该邮件体部分(可以根据邮件头和邮件体部分以空行分开这个特点来提取出邮件体)是否采用了某种编码方式。如果没有编码,可以在 GetContent()中直接返回前面已经得到的 content 值,否则调用需要的编码程序,将解码后的字符串返回。

2)当邮件中出现复合的顶层类型时,情况就比较复杂,下面以较常见的 Multipart 为例进行说明。在 Multipart 类型中,有两个常见子类 Mixed, Alternative。Mixed 类型说明文本内容有多个部分,并且各部分之间是有次序的,那么在分析邮件体的时候,必须考虑到其内容的顺序,按其原先出现的顺序来提取出邮件体内容。而 Alternative 说明文本是以多种类型出现的相同内容,接收者可以根据用户的需要来提取所需要的类型或者默认将所有的类型均显示出来。邮件体中各个部分之间以 boundary 分隔。

boundary 是边界定界符,一般由两个连字符'-'(0x2D)和紧跟着从邮件头 Content-Type 中取来的参数值组成(如 1.2 中所述),用于分隔邮件体内容的不同的各个部分。

在分析邮件体的正文文本时(此处以邮件中出现的是 Multipart 的 Mixed 和 Alternative 两类为例,若要实现其它复合类及其子类可以相似的方式实现),可在 GetContent()中调用一个递归函数。其具体程序流程如下:

CString GetContent1(CString s)//s 为邮件字符串

{若 s 为空,则返回 NULL;

读出 s 中的 Content - Type 类型;

if(s 为离散类型)

{如果不是用户需要显示的类型,则返回 NULL;如果是,则根据邮件正文与邮件头以空行分开的特点,使用 s. Find("\r\n\r\n")来找出正文的开头,将正文提取出来赋值给 sn;

根据邮件头中的 Content - transfer - coding 的内容,来判断邮件使用何种编码方式,使用相应的解码程序来还原字符串 sn,还原为 content 字符串;

返回还原的正文;}

else{//s 为复合类型情况

判断复合类型的子类型为何种情况;

根据邮件头中的 boundary,读出邮件正文中的边界字符串;

同上操作提取出邮件体部分;

case(Mixed)://当其子类型为多个有次序的部分时;

{

用户所需要的类型置为 all;//all 说明在读入邮件时将所有内容都读入}

case(Alternative)://当其子类型为多个形式的单一类型时;

{

读入用户需要的类型标识,若无标识,则默认用户需要显示所有的类型,则用户所需要类型也置为 all;}

根据 boundary,将 s 分为第一个部分和余下部分分别赋值给 sfirst,srest;

content = GetContent1 (sfirst) + GetContent1 (srest);}}

同样,在提取邮件体中的附件的时候,程序框架同上述的提取邮件体的正文文本的程序基本相同,只是在 s 为离散类型的情况的开头加入如下判断:

读入 Content - Disposition 的内容,如果其放置位置为附件,即 Content - Disposition 的内容是 attachment 时才继续进行下面的操作,否则返回 NULL。

2　邮件编码格式

目前,MIME 标准是电子邮件体编码遵循的标准,大部分的邮件采用的都是 MIME 中规定的两种编码格式 Base64 和 QP(Quoted - Printable)。

2.1　Base64 编码

Base64 编码适用于不可读的二进制文件,如中文文档等。编码方法是将原始字符串每三个字符放入一个 24 位缓冲区并等分为 4 份,高位在先,根据每 6 位所对应的数字的大小,用 A～Z,a～z,0～9,+,/共 64 个字符重新表示。若编码后的位数不是 4 的整数倍,在最后以"="来填充。如字符串"hello"的二进制流为:"011010000110010101101100011010001101111",6 位一组并补零后得到:"011010,000110,010101,101100,011011,000110,111100" (a,G,V,s,b,G,8),则其 Base64 编码为"aGVsbG8=",据此可得到相应的解码流程[6]。

2.2　QP 编码

QP 编码主要适用于包含大量 7 位 ASCII 码的多媒体邮件。QP 编码方法是对于 7 位可打印的 ASCII 码数据,编码后的数据保持不变,对于非 ASCII 码的 8bit 字节数据,每个字节用等号及其十六进制表示。如"尊敬",各字的 GB2312 码是 D7F0,BEB4,则其对应的 QP 编码为:"=D7=F0=BE=B4",容易设计出解码算法。

2.3　字符集的讨论

文中讨论时使用的字符集为 GB2312,实际应用中邮件标题和内容中所使用的字符集可能还包括其它类型,如 BIG5 等,此时需要使用相应的算法进行解码,文中不再赘述。

3　结　论

通过以上讨论,可设计出 CMail 类用于 EML 邮件的访问并用于实际的项目开发和研究中,取得了满意的效果。根据具体需求增加相应的成员函数即可方便地扩充程序功能。

参考文献:

[1] Crocker D H. Standard for The Format of APRA Internet Text Messages[S/OL]. 1982. http://rfc.net/rfc0822.html.

[2] Freed N, Borenstein N. Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies [S/OL]. 1996. http://rfc.net/rfc2045.html.

[3] Freed N, Borenstein N. Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types[S/OL]. 1996. http://rfc.net/rfc2046.html.

[4] 曹建文,黄志平,魏新莉. 基于 MIME 的电子邮件发送程序的设计和实现[J]. 中南林学院学报,2004,24(5): 108 - 112.

[5] 王淑蓉,沈　虹. 分析 MIME 邮件组成结构及构建邮件收发系统[J]. 现代电子技术,2004(23):15 - 17.

[6] 陈训逊,方滨兴,李　蕾. MIME 解码算法优化问题研究[J]. 计算机应用,2003,23(12):263 - 265.