文章
姚 鑫 · 五月 27, 2021 阅读大约需 5 分钟

第八章 处理收到的电子邮件

第八章 处理收到的电子邮件

处理收到的电子邮件

本节介绍如何处理通过%Net.POP3检索到的电子邮件(%Net.MailMessage)。

Message Basics

检索电子邮件(%Net.MailMessage)后，通常首先确定它是哪种类型的邮件以及如何阅读它；也就是说，它是否是多部分邮件以及各部分是否是二进制的。在此步骤中，您可以使用ContentType属性。或者，可以使用IsBinary、IsHTML和IsMultiPart属性，它们间接提供与contentType相同的信息。

如果消息是多部分消息，则每个部分都是%Net.MailMessagePart的一个实例。

Message Headers

消息本身和消息的每个部分都有一组标头。

%Net.MailMessage和%Net.MailMessagePart类提供的属性使可以轻松访问最常用的标头。例如，%Net.MailMessage提供收件人、发件人、主题和日期等属性。Headers数组属性允许访问任何自定义标题.

此外，如果已通过%Net.POP3检索到消息，则可以使用GetAttribute()方法。在给定标头名称和属性的情况下，此方法返回该属性的值。

Message Contents

了解常规消息结构后，请使用以下技术检索内容：

- 对于多部分消息，请使用Parts属性，该属性是部分的数组。Parts.Count()给出部件的数量。每个部件的键都是一个整数，从1开始。使用GetAt()方法检索给定的部件。消息部分是%Net.MailMessagePart的实例。
- 对于二进制消息(或消息部分)，请使用BinaryData属性。
- 对于文本消息(或消息部分)，请使用TextData属性。
    - 如果IsHTML为0，则TextData属性为普通文本字符串。
    - 如果IsHTML为1，则TextData属性为HTML文本字符串。

请注意，发送邮件的电子邮件客户端确定邮件中的任何包装。邮件服务器无法控制这一点,

其他消息信息

MessageSize属性表示邮件的总长度(不包括任何附加的电子邮件)。

以下方法提供有关消息的其他信息：

GetLocalDateTime()

返回检索消息的日期和时间，并转换为$HOROLOG格式的本地时间。

GetUTCDateTime()

返回检索消息的日期和时间，并以$HOROLOG格式转换为UTC。

GetUTCSeconds()

返回自1840年12月31日以来检索消息的日期和时间(秒)。

以下类方法也可用于时间/日期转换：

HToSeconds()

将$HOROLOG格式的日期/时间转换为自1840年12月31日以来的秒的类方法。

SecondsToH()

将自1840年12月31日以来的秒数转换为$HOROLOG格式的日期/时间的类方法。

示例1：ShowMsgInfo()

```
ClassMethod ShowMsgInfo(msg as %Net.MailMessage)
{
    Write "Message details *****",!
    Write "To (count): ", msg.To.Count(),!
    Write "From: ", msg.From,!
    Write "Cc (count): ", msg.Cc.Count(),!
    Write "Bcc (count): ", msg.Bcc.Count(),!
    Write "Date: ", msg.Date,!
    Write "Subject: ", msg.Subject,!
    Write "Sender: ", msg.Sender,!
    Write "IsMultipart: ", msg.IsMultiPart,!
    Write "Number of parts: ", msg.Parts.Count(),!
    Write "Number of headers: ", msg.Headers.Count(),!
    Write "IsBinary: ", msg.IsBinary,!
    Write "IsHTML: ", msg.IsHTML,!
    Write "TextData: ", msg.TextData.Read(),!
    Write "BinaryData: ", msg.BinaryData.Read(),!
}
```

此方法产生类似于以下内容的输出：

```
Message details *****
To (count): 1
From: "XXX XXX" <XXX@XXX.com>
Cc (count): 0
Bcc (count): 0
Date: Fri, 16 Nov 2007 11:57:46 -0500
Subject: test 5
```

```
Sender:
IsMultipart: 0
Number of parts: 0
Number of headers: 16
IsBinary: 0
IsHTML: 0
TextData: This is test number 5, which is plain text.
BinaryData:
```

示例2：ShowMsgPartInfo()

以下方法写入有关消息一部分的信息：

```
ClassMethod ShowMsgPartInfo(msg as %Net.MailMessage, partno as %Integer)
{
    Set part=msg.Parts.GetAt(partno)
    Write "Message part details *****",!
    Write "Message part: ", partno,!
    Write "IsMultipart: ", part.IsMultiPart,!
    Write "Number of parts: ", part.Parts.Count(),!
    Write "Number of headers: ", part.Headers.Count(),!
    Write "IsBinary: ", part.IsBinary,!
    Write "IsHTML: ", part.IsHTML,!
    Write "TextData: ", part.TextData.Read(),!
    Write "BinaryData: ", part.BinaryData.Read(),!
}
```

这将产生与以下内容类似的输出(给定的消息与前面显示的不同)：

```
Message part details *****
Message part: 1
IsMultipart: 0
Number of parts: 0
Number of headers: 2
IsBinary: 0
IsHTML: 0
TextData: 1 test string


BinaryData:
```

示例3：ShowMsgHeaders()

下面的方法写入有关消息头的信息；可以编写一个类似的方法，对消息部分执行相同的操作。

```
ClassMethod ShowMsgHeaders(msg as %Net.MailMessage)
{
    Set headers=msg.Headers
    Write "Number of headers: ", headers.Count(),!

    //iterate through the headers
    Set key=""
    For {
```

```
        Set value=headers.GetNext(.key)
        Quit:key=""
        Write "Header:",key,!
        Write "Value: ",value,!!
    }
}
```

这将产生类似于以下内容的输出：

```
Number of headers: 16
Header: content-class
Value: urn:content-classes:message

Header: content-type
Value: multipart/alternative; boundary="----_=_NextPart_001_01C8286D.D9A7F3B1"

Header: date
Value: Fri, 16 Nov 2007 11:29:24 -0500

Header: from
Value: "XXX XXX" <XXX@XXX.com>

Header: message-id
Value: <895A9EF10DBA1F46A2DDB3AAF061ECD501801E86@Exchange1_backup>

Header: mime-version
Value: 1.0

...
```

自动编码和字符翻译

电子邮件部分包含有关使用的字符集和使用的内容传输编码(如果有的话)的信息。作为参考，本节介绍如何使用此信息。

外发电子邮件

%Net.SMTP检查每个部分的字符集属性，然后应用适当的转换表。

如果未指定给定部件的字符集属性，InterSystems IRIS将使用UTF-8。

%Net.SMTP还检查ContentTransferEncoding属性。如果此属性为 "base64"或"quoted-printable"，则在创建消息时，%Net.SMTP会根据需要对正文进行编码。(如果内容传输编码为 "7bit" 或 "7bit"，则不需要编码。)

重要提示：请注意，如果内容为 “ Base64 ” 编码，则不能包含任何Unicode字符。如果要发送的内容包括Unicode字符，请确保使用$ZCONVERT将内容转换为UTF-8。

传入电子邮件

%Net.POP3检查每个邮件部分的Content-Transfer-Encoding标头，并根据需要对正文进行解码。

然后%Net.POP3检查每个邮件部分的Content-

Type标头。这会影响消息部分的字符集属性，还会控制在InterSystems IRIS中创建消息部分时使用的转换表。

#SQL #Caché #InterSystems IRIS #InterSystems IRIS for Health

源URL: https://cn.community.intersystems.com/post/%E7%AC%AC%E5%85%AB%E7%AB%A0-%E5%A4%84%E7%90%86%E6%94%B6%E5%88%B0%E7%9A%84%E7%94%B5%E5%AD%90%E9%82%AE%E4%BB%B6