

文章
姚鑫 · 六月 15, 2021 阅读大约需分钟

第八章 Other Options of the Writer

第八章 Other Options of the Writer

Other Options of the Writer

Canonicalize()

方法的作用是：以规范的形式编写XML节点。此方法具有以下签名：

```
method Canonicalize(node As %XML.Node, ByRef PrefixList, formatXML As %Boolean = 0) as %Status
```

- node是文档的一个子树，作为%XML.Node的实例。
- PrefixList是以下其中之一：
 - 对于包含式规范，将PrefixList指定为“c14n”。
在本例中，输出的形式是XML Canonicalization Version 1.0，由<https://www.w3.org/TR/xml-c14n>。
 - 对于排他规范，将PrefixList指定为具有以下节点维数组：

Node	Value
PrefixList(前缀)，其中前缀是名称空间前缀	与此名称空间前缀一起使用的名称空间

- FormatXML控制格式。如果format XML为true，则编写器使用为编写器实例指定的格式，而不是XML规范指定的格式。因此，输出不是规范的XML，但是已经对规范的XML进行了命名空间处理。此选项对于输出XML文档的片段(如Web服务的ProcessBodyNode()回调中的SOAP主体)很有用，同时仍对格式有一定的控制。

隐藏属性

编写器实例的隐藏属性影响具有对象值的属性输出。此类属性强制所有此类输出为隐藏输出，也就是说，强制输出包含被引用对象的ID，而不是对象的详细信息。此属性XMLDEFAULTREFERENCEONS和XMLREFERENCIONS(在启用XML的对象属性交互，如表所示。表显示了每种情况的结果输出：

Effect of Shallow = 1	
XMLREFERENCE和XMLDEFAULTREFERENCE的值	Output if Shallow=1
属性XMLREFERENCE是“SUMMARY”或“COMPLETE”	此属性生成输出
属性XMLREFERENCE为“ID”、“OID”或“GUID”	该属性输出，其类型为ID、OID或GUID
属性XMLREFERENCE没有设置，但是类参数XMLDEFAULTREFERENCE是“SUMMARY”或“COMPLETE”	此属性生成输出
属性XMLREFERENCE没有设置，但类参数XMLDEFAULTREFERENCE是“ID”、“OID”或“GUID”	该属性输出，其类型为ID、OID或GUID
属性XMLREFERENCE和类参数XMLDEFAULTREFERENCE都没有设置	此属性生成输出
Shallow属性影响其值为串行对象的属性具有非对象值的属性	

Summary Property

编写器实例的Summary属性控制是导出整个启用XML的对象，还是只导出其摘要；它可以是下列值之一：

- 值0会导出整个对象；这是默认设置。
- 值1仅导出作为摘要列出的属性

如何将对象映射到XML中所述，对象的摘要由其XMLSUMMARY在类参数中指定；它是逗号分隔的属性列表。例如，为启用了XML的Person类生成输出，并且默认输出如所示：

```
<Persons>
  <Person>
    <Name>Xenia, Yan T.</Name>
    <DOB>1986-10-21</DOB>
  </Person>
  <Person>
    <Name>Vivaldi, Ashley K.</Name>
    <DOB>1981-01-25</DOB>
  </Person>
</Persons>
```

将Person类的XMLSUMMARY设置为“name”。在这种情况下，如果编写器将Summary属性置为1，则输出将如所示：

```
<Persons>
  <Person>
    <Name>Xenia, Yan T.</Name>
  </Person>
  <Person>
    <Name>Vivaldi, Ashley K.</Name>
  </Person>
</Persons>
```

Base64LineBreaks属性

可以%Binary类型的属性自动换行符，也可以%xsd.base64Binary类型的属性自动换行符。为此，请将编写器实例的Base64LineBreaks属性置为1。在这种情况下，编写器在每76个字符之后插入一个自动换行符回车符。此属性默认值为0。

CycleCheck属性

编写器实例的CycleCheck属性控制编写器是否检查引用对象内可能导致错误的任何循环(死循环)。默认值为1，这意味着编写器确实会检查日期。

如果确定没有日期，请将CycleCheck设置为0以略微提高性能。

其他示例：可选择设置的编写器

对于 %XML.Writer的属性，以下方法可能很有用。它接受一个输入参数，该参数是一个将编写器命名为“version”的字符串。每个编写器版本对应于编写器实例的属性特定设置。

```
Class Util.XmlUtils Extends %RegisteredObject
{

/// ???wname????????????????????
ClassMethod CreateWriter(wname) As %XML.Writer
{
    set w=##class(%XML.Writer).%New()
    set w.Indent=1
    set w.IndentChars="  "
    if wname="DefaultWriter" {
        set w.Indent=0 ; set back to default
    }
    elseif wname="EncodedWriter" {
        set w.Format="encoded"
    }
    elseif wname="EncodedWriterRefInline" {
        set w.Format="encoded"
        set w.ReferencesInline=1
    }
    elseif wname="AttQualWriter" {
        set w.AttributeQualified=1
    }
    elseif wname="AttUnqualWriter" {
        set w.AttributeQualified=0 ; default
    }
    elseif wname="ElQualWriter" {
        set w.ElementQualified=1 ; default
    }
    elseif wname="ElUnqualWriter" {
        set w.ElementQualified=0
    }
    elseif wname="ShallowWriter" {
        set w.Shallow=1
    }
    elseif wname="SOAPWriter1.1" {
        set w.Format="encoded"
        set w.ReferencesInline=1
    }
    elseif wname="SOAPWriter1.2" {
        set w.Format="encoded12"
        set w.ReferencesInline=1
    }
    elseif wname="SummaryWriter" {
        set w.Summary=1
    }
    elseif wname="WriterNoXmlDecl" {
        set w.NoXMLDeclaration=1
    }
    elseif wname="WriterRefInline" {
        set w.ReferencesInline=1
    }
    elseif wname="WriterRuntimeIgnoreNull" {
        set w.RuntimeIgnoreNull=1
    }
    elseif wname="WriterSuppressXmlns" {
        set w.SuppressXmlns=1
    }
    elseif wname="WriterUTF16" {
        set w.Charset="UTF-16"
    }
}
```

```

}
elseif wname="WriterWithDefNS" {
    set w.DefaultNamespace="www.Def.org"
}
elseif wname="WriterWithDefNSSuppressXmlns" {
    set w.DefaultNamespace="www.Def.org"
    set w.SuppressXmlns=1
}
elseif wname="WriterWithDtdSettings" {
    set w.DocType ="MyDocType"
    set w.SystemID = "http://www.mysite.com/mydoc.dtd"
    set w.PublicID = "-//W3C//DTD XHTML 1.0 Transitional//EN"
    set w.InternalSubset = ""
}
elseif wname="WriterXsiTypes" {
    set w.OutputTypeAttribute=1
}
quit w
}
}

```

该片段显示了如何使用此方法帮助生成当示例：

```

/// method to write one to a file
ClassMethod WriteOne(myfile, cls, element, wname, ns, local, rootns)
{
    set writer=##class(Util.XmlUtils).CreateWriter(wname)
    set mydir="e:/temp/"
    set comment="Output for the class: "_cls
    set comment2="Writer settings: "_wname

    if $extract(mydir,$length(mydir))'="/" {set mydir=mydir_"/"}
    set file=mydir_myfile
    set status=writer.OutputToFile(file)
    if $$$ISERR(status) { do $System.Status.DisplayError(status) quit }

    set status=writer.WriteComment(comment)
    if $$$ISERR(status) { do $System.Status.DisplayError(status) quit }

    set status=writer.WriteComment(comment2)
}

```

请注意，输出将包括两个注释行。一个表示文件中显示的启用XML的类的名称。另一个指示用于生成文件的编辑器设置的名称。输出目录是集中控制的(通过参数)，这个泛型方法包括传递给RootElement()方法和Object()方法的参数。

[#Caché #InterSystems IRIS](#)

源 URL: <https://cn.community.intersystems.com/post/%E7%AC%AC%E5%85%AB%E7%AB%A0-other-options-writer>