链滴

Dom4j 解析含前缀与名字空间的 XML

作者：linyiheng

原文链接：https://ld246.com/article/1503243041097

来源网站：链滴

Dom4j解析含前缀名字空间与默认名字空间的XML

示例XML

```
<s:Envelope xmlns:a="http://www.w3.org/2005/08/addressing"
   xmlns:s="http://www.w3.org/2003/05/soap-envelope">
   <s:Header>
      <a:Action s:mustUnderstand="1">http://schemas.microsoft.com/windows/management
2012/01/enrollment/IDiscoveryService/Discover</a:Action>
      <a:MessageID>urn:uuid:748132ec-a575-4329-b01b-6171a9cf8478</a:MessageID>
      <a:ReplyTo>
         <a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
      </a:ReplyTo>
      <a:To s:mustUnderstand="1">https://host.linyiheng.cn:8004/EnrollmentServer/Discovery
svc</a:To>
   </s:Header>
   <s:Body>
      <Discover
         xmlns="http://schemas.microsoft.com/windows/management/2012/01/enrollment">
         <request xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
            <EmailAddress>admin@aa</EmailAddress>
            <RequestVersion>1.0</RequestVersion>
         </request>
      </Discover>
   </s:Body>
</s:Envelope>
```

基础知识

1. 含前缀名字空间，和默认名字空间存在的意义在于它们用来标识XML元素所属ID及赋予该ID标识素相关约束等功能。

2. 默认名字空间所属元素下所有子元素，若未被其他前缀定义则默认识别默认名字空间为其ID。

3. XPath语法："//前缀:标签"，满足前缀:标签的元素即选出。"/前缀:标签/..."即从文档根路径搜索符条件元素。详情可见下表：

4. 当搜索的标签属于默认名字空间，则前缀，域HashMap中以xmlns作为其key

表达式	描述
nodename	选取此节点的所有子节点
/	从根节点选取
//	从匹配选择的当前节点选择文档中的节点，而考虑它们的位置
.	选取当前节点
..	选取当前节点的父节点
@	选取属性

示例代码

以下代码用于从上面XML中取出EmailAddress标签中的值

```
package cn.linyiheng.mde;

import java.io.File;
import java.io.IOException;
import java.util.HashMap;
import java.util.Map;

import org.apache.commons.io.FileUtils;
import org.apache.commons.io.IOUtils;
import org.dom4j.DocumentException;
import org.dom4j.DocumentHelper;

public class SoapXML {
    @org.junit.Test
    public void generateSoapXML() {

    }

    @org.junit.Test
    public void resolveSoapXML() {
        String requestContent="";
        try {
            requestContent = FileUtils.readFileToString(new File("mde/discoveryRequest.xml"), "U
F-8");
        } catch (IOException e2) {
            e2.printStackTrace();
        }

        org.dom4j.Document doc = null;
        try {
            doc = DocumentHelper.parseText(requestContent);
        } catch (DocumentException e1) {
            e1.printStackTrace();
        }
        Map<String, String> nsmap = new HashMap<String, String>();
        System.out.println(doc.getRootElement().getNamespacePrefix());
        System.out.println(doc.getRootElement().getNamespaceURI());
        nsmap.put(doc.getRootElement().getNamespacePrefix(), doc.getRootElement().getName
paceURI());
        nsmap.put("xmlns", "http://schemas.microsoft.com/windows/management/2012/01/enro
lment");
        org.dom4j.XPath xBody=doc.createXPath("/s:Envelope/s:Body");
        org.dom4j.XPath xEmailAddress=doc.createXPath("/s:Envelope/s:Body/xmlns:Discover/x
lns:request/xmlns:EmailAddress");
        xBody.setNamespaceURIs(nsmap);
        xEmailAddress.setNamespaceURIs(nsmap);
        //System.out.println(xBody.selectSingleNode(doc).getStringValue());
        System.out.println(xEmailAddress.selectSingleNode(doc).getStringValue());
        // String nsURI = doc.getRootElement().getNamespaceURI();
        /*
         * nsmap.put("xmlns", nsURI); org.dom4j.XPath
         * xMsgID=doc.createXPath("//xmlns:SyncML/xmlns:SyncHdr/xmlns:MsgID");
```

原文链接：Dom4j 解析含前缀与名字空间的 XML

```
        * xMsgID.setNamespaceURIs(nsmap); String
        * strMsgID=xMsgID.selectSingleNode(doc).getStringValue();
        */
    }
}
```