



Chapter4-SAX解析

SAX的全称是Simple APIs for XML，也即XML简单应用程序接口。与DOM不同，SAX提供的访问模式是一种顺序模式，这是一种快速读写XML数据的方式。当使用SAX分析器对XML文档进行分析时，会触发一系列事件，并激活相应的事件处理函数，应用程序通过这些事件处理函数实现对XML文档的访问，因而SAX接口也被称作事件驱动接口。

SAX分析器在对XML文档进行分析时，触发了一系列的事件，由于事件触发本身是有序性的，因此，SAX提供的是一种顺序访问机制，对于已经分析过的部分，不能再倒回去重新处理。SAX之所以被叫做“简单”应用程序接口，是因为SAX分析器只做了一些简单的工作，大部分工作还要由应用程序自己去做。也就是说，SAX分析器在实现时，它只是顺序地检查XML文档中的字节流，判断当前字节是XML语法中的哪一部分、是否符合XML语法，然后再触发相应的事件，而事件处理函数本身则要由应用程序自己来实现。同DOM分析器相比，SAX分析器缺乏灵活性。然而，由于SAX分析器实现简单，对内存要求比较低，因此实现效率比较高，对于那些只需要访问XML文档中的数据而不对文档进行更改的应用程序来说，SAX分析器更为合适。

SAX分析器的大体构成框架



南京网博
专业专注 全心服务



- 图中最上方的SAXParserFactory用来生成一个分析器实例。XML文档是从左侧箭头所示处读入，当分析器对文档进行分析时，就会触发在DocumentHandler, ErrorHandler, DTDHandler以及EntityResolver接口中定义的回调方法。

org.xml.sax.helpers.DefaultHandler 类的方法

项目	处理方法
文档开始	startDocument()
<Person>	startElement()
“李元霸”	characters()
</ Person >	endElement()
文档结束	endDocument()

使用SAX解析XML的优点



南京网博
专业专注 全心服务

- 与DOM相比，SAX的优点如下：
 - 内存消耗小
 - 无需为所有节点创建对象
- 适用情形：
 - 需要解析大型XML文档
 - 只需要文档数据的部分信息
- 缺点
 - 不能对文档进行随机遍历
 - 只能遍历文档一次

SAX (Simple APIs for XML, 操作 XML 的简单接口) 与 DOM 操作不同的是, SAX 采用的是一种顺序的模式进行访问, 是一种快速读取 XML 数据的方式。当使用 SAX 解析器进行操作时会触发一系列的事件, 如表 3-7 所示。当扫描到文档 (Document) 开始与结束、元素 (Element) 开始与结束时都会调用相关的处理方法, 并由这些操作方法作出相应的操作, 直至整个文档扫描结束, 如图 3-11 所示。

表 3-7 SAX 主要事件

No.	方 法	类 型	描 述
1	public void startDocument() throws SAXException	普通	文档开始
2	public void endDocument() throws SAXException	普通	文档结束
3	public void startElement(String uri,String localName, String qName,Attributes attributes) throws SAXException	普通	元素开始, 可以取得元素的名称及元素的全部属性
4	public void endElement(String uri,String localName, String qName) throws SAXException	普通	元素结束, 可以取得元素的名称及元素的全部属性
5	public void characters(char[] ch,int start,int length) throws SAXException	普通	元素内容



```
public class SAXParserHandler extends DefaultHandler
{
    public void startDocument() throws SAXException{
    }
    public void endDocument() throws SAXException{
    }
    public void characters(char[] ch,int start,int length)throws SAXException{
    }
    public void startElement(String namespaceURI,String localName,String
qName,Attributes atts)throws SAXException{
    }
    public void endElement(String namespaceURI,String localName,String
qName)throws SAXException{
    }
}
```

- SAXParserFactory

- ☞ 主要用来创建SAXParser对象

- SAXParser

- ☞ `Parse(File f, DefaultHandler sax)`

Sax解析与Dom解析的优缺点



南京网博
专业专注 全心服务

No.	区 别	DOM 解析	SAX 解析
1	操作	将所有文件读取到内存中形成 DOM 树，如果文件量过大，则无法使用	顺序读入所需要的文件内容，不会一次性全部读取，不受文件大小的限制
2	访问限制	DOM 树在内存中形成，可以随意存放或读取文件树的任何部分，没有次数限制	由于采用部分读取，只能对文件按顺序从头到尾解析一遍，不支持对文件的随意存取
3	修改	可以任意修改文件树	只能读取 XML 文件内容，但不能修改
4	复杂度	易于理解，易于开发	开发上比较复杂，需要用户自定义事件处理器
5	对象模型	系统为用户自动建立 DOM 树，XML 对象模型由系统提供	对开发人员更加灵活，可以用 SAX 建立自己的 XML 对象模型

37、XML三种解析方式分别是什么，区别是什么？

由不同的组织提供了三类解析XML的方式：DOM，SAX，STAX，其他

- 1) DOM：官方提供，w3c，所有语言都可以使用DOM对html或xml进行解析操作，Java也提供了此类支持，但解析代码相对比较复杂。
 - a) 优点：兼容性好，可以支持创建XML，只要在内存允许范围内，解析速度都很快
 - b) 缺点：代码烦琐，如果超出内存范围，会导致服务器崩溃。
- 2) SAX：官方提供，SUN，只能Java使用，逐行解析，必须编写一个通用的解析规则，当碰到某个XML文件时，选择套用某一解析规则，自动完成解析操作。
 - a) 优点：对内存无压力
 - b) 缺点：代码很难想象，无法创建XML，对小文件解析速度比较慢。
- 3) 其他：第三方提供，很多种，例如：JDOM，dom4j，综合了以上两种解析的优点，并将DOM解析中的一些不是Java规范的代码，整合到java规范中，使开发更容易
 - a) 优点：很多
 - b) 缺点：不支持其他语言。

88、XML基本语法（背）

XML：可扩展的标记语言

HTML：超文本标记语言

DHTML：就是div + css + javascript + flash，对普通html进行一些页面动态效果的扩展。

XHTML：是w3c对所有动态网页技术的一个整合，所有动态网页技术都可以写成XHTML。

标记语言就是以各种标记（节点）组成的语法。

html中，标记是固定的，不可以任意改变，每个标记都有自己的作用。

xml中，标记是可以任意规定的，可以随意修改

xml组成分为两部分：头信息，主体信息。

头信息用来进行一些配置和导入工作。

设置编码以及版本。

```
<?xml version="1.0" encoding="GBK"?>
```

版本只有1.0，encoding设置是否支持中文，默认编码是ISO-8859-1

头信息中还可以导入css或xsl，用来设置xml的显示格式。

html的作用是用来显示信息给用户浏览器查看的。xml一般是给开发人员用来配置数据和传递数据的。