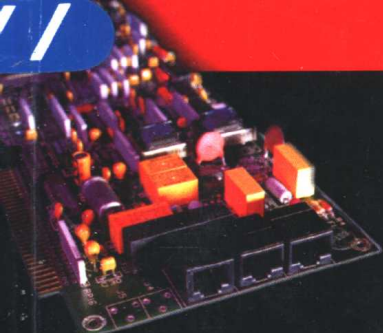


数据库丛书



Oracle9i

Programming with XML

编程手册

陈刚 刘晋东 译

学苑出版社

数据库丛书

TP311.132.3
16

Oracle9i

Programming with XML

编程手册

陈刚 刘晋东 译

学苑出版社

(京)新登字 151 号

内 容 简 介

本书是用 XML 开发 Oracle9i 的参考手册。

全书由五部分组成。第一部分包括第 1、2、3、4 章,介绍了 Oracle.xml.parser.v2 包、oracle.xml.classgen 类生成器、XSQL Servlet 以及 Transviewer Beans;第二部分包括第 5 章,介绍了基于 C 语言的 XML 解析器;第三部分包括第 6 章,介绍了基于 C 语言的 XML 解析器;第四部分包括第 7、8 介绍了基于 C++的 XML 解析器与 Oracle XML 类生成器;第五部分包括第 9、10 章,介绍了 Oracle 中 XSU 的 Java API 函数以及 PL/SQL API 函数。

本书内容新、丰富,注重基础知识与实际应用相结合,实用性强,是使用 XML 进行 Oracle 数据库开发的高级开发人员和编程人员的参考手册,同时也可作为高校计算机应用专业师生的教学与自学参考用书。

需要本书的用户可直接与北京中关村 083 信箱市场部联系,电话:62541992,邮编:100080。

数据库丛书

Oracle9i Programming with XML 编程手册

译者:陈刚 刘晋东

审校:周风明

责任编辑:甄国宪

排版:希望图书输出中心 杜海燕

出版发行:学苑出版社 邮政编码:100036

社址:北京市海淀区万寿路西街 11 号

印刷:北京双青印刷厂

开本:787×1092 1/16

印张:34.25 字数:621.5 千字

印数:5000~7000 册

版次:2001 年 9 月第 1 版第 2 次印刷

ISBN 7-5077-0886-1/TP·28

本册定价:55.00 元

学苑版图书印、装错误可随时退换

序 言

随着Internet技术的日新月异,网络的发展为我们带来了许多难以想象的便利,在这个发展过程中,HTML语言作出了不可磨灭的贡献,尽管如此,HTML语言却越来越难以满足网络发展的要求,为了解决这种矛盾,新一代的标记语言XML应运而生,它在许多方面为网络带来了革命性的变革。

XML是“Extensible Markup Language”的缩写,即可扩展标记语言。它是Internet环境中跨平台的、依赖于内容的技术,是处理分布式结构信息的有力工具。在W3C组织的领导下,XML技术日益成熟。与HTML一样,XML也源自SGML(Standard Generalize Markup Language),可以说是SGML的一个简化版本,它有着HTML语言所欠缺的巨大的伸缩性与灵活性,突破了HTML固定标记集合的约束,用户可以定义自己的标记来描述文件中的任何数据元素,使文件的内容更丰富更复杂并组成一个完整的信息体系。

XML的主要特点有:良好的数据存储格式、可扩展性、高度结构化、易于网络传输等。通过使用XML语言,可以方便的实现不同工业标准之间的数据交换,因此,Oracle、IBM以及Microsoft公司都积极地投入了大量人力与物力研发XML相关软件,这无疑确定了XML在IT产业的美好前景,本书正是介绍XML语言在Oracle 9i中的使用规范。

通过Oracle发布XML组件,用户可以访问Oracle 8i及Oracle 9i数据库里的数据,并可以将XML文档转换为HTML文档或其他文件格式,总的来说,XML语言主要应用于以下几个方面:

1. 在不同的数据库之间的进行数据传递,解决了数据的统一接口问题,但是,XML并没有定义数据文件中数据表达的具体规范,而是使用标签来表达数据的逻辑结构和含义。

2. 在客户端进行分布式数据处理,服务器只需发出一个XML文档,在客户端就可以根据不同的需要生成不同的应用程序进行数据处理。XML的自解释性使客户端在收到数据的同时也理解数据的逻辑结构与含义,从而使广泛、通用的分布式计算成为可能。

3. 用户可以根据自己的需要对数据源进行加工和重组,以适合不同方面的数据需求。

本书从下面几个方面详细介绍了在Oracle9i中使用XML的规范:

- 基于JAVA包的XML解析器,其中介绍了Oracle.xml.parser.v2包、oracle.xml.classgen类生成器、XSQL Servlet以及Transviewer Beans中的接口、类和异常的语法定义和继承关系,以及它们所包括的字段和方法。
- 基于C语言包的XDK,介绍了基于C语言的XML解析器,其中包括Parser API函数、W3 Sax API函数、W3 DOM API函数以及名字空间API函数。
- 基于C++包的XDK,介绍了基于C++的XML解析器与Oracle XML类生成器。
- XML-SQL通用包,介绍了Oracle中XSU的Java API函数以及PL/SQL API函数。

本书主要由陈刚和刘晋东翻译,其中第二、三、四、六、七章是由陈刚翻译;第一、五、八、九、十章由刘晋东翻译。此外,还有以下人员参与了翻译和校对工作:邹伟,顾农,赵天姝,唐建波,梁章敏,韩江,张家澄,张豫宁,张典臣,陈宝鹏,张春利。由于作者水平有限,书中难免出现缺点和不足,殷切希望广大读者批评指正。

译者

JS8 16 / 11

目 录

第一部分 基本 Java 包的 XML 解析器

第 1 章 基于 JAVA 的 XML 解析器	3	1.29 XSLProcessor.....	116
1.1 说明	3	1.30 XSLStylesheet.....	121
1.2 AttrDecl.....	4	第 2 章 基于 JAVA 的 XML 类生成器	124
1.3 DefaultXMLDocumentHandler	8	2.1 说明	124
1.4 DOMParser.....	13	2.2 oracle.xml.classgen	124
1.5 DTD.....	18	2.3 oracle.xml.classgen	125
1.6 ElementDecl	25	2.4 ClassGenerator(类生成器)	125
1.7 NodeFactory	30	2.5 InvalidContentException	127
1.8 NSName	32	第 3 章 XSQL Servlet	129
1.9 NDRResolver	34	3.1 Res.....	129
1.10 oraxsl	34	3.2 XSQLActionHandler	138
1.11 SAXAttrList	36	3.3 XSQLActionHandleImpl	139
1.12 SAXParser	40	3.4 XSQLCommandLine	140
1.13 XMLAttr.....	43	3.5 XSQLDiagnostic	140
1.14 XMLCDATA	49	3.6 XSQLHttpUtil	141
1.15 XMLComment	52	3.7 XSQLPageRequest.....	142
1.16 XMLDocument	54	3.8 XSQLPageRequestImpl.....	147
1.17 XMLDocumentFragment	66	3.9 XSQLParserHelpe.....	153
1.18 XMLDocumentHandler	68	3.10 XSQLRequest	155
1.19 XMLElement.....	72	3.11 XSQLServlet	158
1.20 XMLEntityReference	81	3.12 XSQLServletPageRequest	159
1.21 XMLNode	83	3.13 XSQLStylesheetProcessor	163
1.22 XMLParseException	93	3.14 XSQLUtil	164
1.23 XMLParser	96	第 4 章 TransViewer Beans	167
1.24 XMLPI.....	101	4.1 oracle.xml.async 包说明.....	167
1.25 XMLText	104	4.2 DOMBuilder	167
1.26 XMLToken.....	107	4.3 DOMBuilderBeanInfo	176
1.27 XMLTokenizer.....	111	4.4 DOMBuilderErrorEvent.....	177
1.28 XSLException	116	4.5 DOMBuilderErrorListener.....	179

4.6	DOMBuilderEvent	179	4.37	XMLTreeView.....	240
4.7	DOMBuilderListener	180	4.38	XMLTreeViewBeanInfo.....	244
4.8	ResourceManager.....	181	4.39	oracle.xml.srcviewer Package.....	245
4.9	XSLTransformer.....	183	4.40	Oracle.xml.srcviewer.XMLSourceView ...	245
4.10	XSLTransformerBeanInfo.....	186	4.41	XMLSourceViewBeanInfo	258
4.11	XSLTransformerErrorEvent	188	4.42	oracle.xml.async 包.....	259
4.12	XSLTransformerErrorListener	189	4.43	DOMBuilder	260
4.13	XSLTransformerEvent.....	189	4.44	DOMBuilderBeanInfo	268
4.14	XSLTransformerListener.....	191	4.45	DOMBuilderErrorEvent.....	270
4.15	oracle.xml.srcviewer Package	192	4.46	DOMBuilderErrorListener.....	271
4.16	XMLSourceViewBeanInfo.....	204	4.47	DOMBuilderEvent	272
4.17	oracle.xml.async 包.....	206	4.48	DOMBuilderListener	273
4.18	DOMBuilder.....	206	4.49	ResourceManager.....	274
4.19	DOMBuilderBeanInfo.....	215	4.50	XSLTransformer.....	275
4.20	DOMBuilderErrorEvent.....	217	4.51	XSLTransformerBeanInfo.....	279
4.21	DOMBuilderErrorListener	218	4.52	XSLTransformerErrorEvent.....	280
4.22	DOMBuilderEvent	219	4.53	XSLTransformerErrorListener	282
4.23	DOMBuilderListener	220	4.54	XSLTransformerEvent	282
4.24	ResourceManager.....	221	4.55	XSLTransformerListener	283
4.25	XSLTransformer	222	4.56	oracle.xml.transviewer 包.....	284
4.26	XSLTransformerBeanInfo	226	4.57	DBAccess.....	285
4.27	XSLTransformerErrorEvent	227	4.58	XMLTransformPanel	289
4.28	XSLTransformerErrorListener	228	4.59	XMLTransformPanelBeanInfo.....	292
4.29	XSLTransformerEvent.....	229	4.60	XMLTransViewer.....	293
4.30	XSLTransformerListener.....	230	4.61	oracle.xml.treeviewer 包.....	294
4.31	oracle.xml.transviewer Package	231	4.62	XMLTreeView.....	294
4.32	DBAccess	231	4.63	XMLTreeViewBeanInfo.....	297
4.33	XMLTransformPanel.....	235	4.64	说明.....	299
4.34	XMLTransformPanelBeanInfo.....	238	4.65	XMLTreeView.....	299
4.35	XMLTransViewer.....	239	4.66	XMLTreeViewBeanInfo.....	302
4.36	oracle.xml.treeviewer 包.....	240			

第二部分 XDK for PL/SQL Packages

第5章	基于 PL/SQL 的 XML 解析器	307
-----	---------------------------	-----

5.1	PL/SQL 解析器 API 函数.....	307
-----	------------------------	-----

5.2 类型和方法 (Types and Functions)	307	的方法	352
5.3 解析器接口类型描述	309	5.29 关于文档碎片 (Document Fragment)	
5.4 函数原型(Function Prototypes)	309	的方法	352
5.5 扩展样式单语言 (Extensible Stylesheet		5.30 关于文档类型 (Document Type) 的	
Language, XSL) 包处理器 API 函数	315	方法	353
5.6 W3C 文档对象模型 (W3C Document		5.31 关于元素 (Element) 的方法	357
Object Model ,DOM) API 函数	320	5.32 关于实体 (Entity) 的方法	361
5.7 类型	321	5.33 关于实体引用 (Entity Reference) 的	
5.8 方法	323	方法	361
5.9 关于命名节点映射表(Named Node		5.34 关于记法 (Notation) 的方法	362
Map)的方法	327	5.35 关于处理指令 (Processing Instruction)	
5.10 关于节点列表(Node List)的方法	327	的方法	363
5.11 关于属性 (Attr) 的方法	328	5.36 关于文本 (Text) 的方法	363
5.12 关于 C Data Section 的方法	329	5.37 关于文档 (Document) 的方法	364
5.13 关于字符型数据的方法	329	5.38 org.w3c.dom.Attr 接口	371
5.14 关于元素 (Element) 的方法	333	5.39 org.w3c.dom.DATASEction 接口	372
5.15 关于实体 (Entity) 的方法	335	5.40 org.w3c.dom.CharacterData 接口	373
5.16 关于实体引用 (Entity Reference)		5.41 org.w3c.dom.Comment 接口	376
的方法	335	5.42 org.w3c.dom.Document 接口	376
5.17 关于记法 (Notation) 的方法	336	5.43 org.w3c.dom.DocumentFragment 接口	380
5.18 关于处理指令 (Processing Instruction)		5.44 org.w3c.dom.DocumentType 接口	381
的方法	336	5.45 org.w3c.dom.DOMException 类	382
5.19 关于文本 (Text) 的方法	337	5.46 org.w3c.dom.DOMImplementation	383
5.20 关于文档 (Document) 的方法	337	5.47 org.w3c.dom.Element 接口	384
5.21 关于节点 (Node) 的方法	340	5.48 org.w3c.dom.Entity 接口	388
5.22 关于命名节点映射表(Named Node		5.49 org.w3c.dom.EntityReference 接口	389
Map)的方法	347	5.50 org.w3c.dom.NamedNodeMap 接口	389
5.23 关于节点列表(Node List)的方法	347	5.51 org.w3c.dom.Node 接口	391
5.24 关于属性 (Attr) 的方法	348	5.52 org.w3c.dom.NodeList 接口	397
5.25 关于 C Data Section 的方法	349	5.53 org.w3c.dom.Notation	398
5.26 关于字符型数据的方法	350	5.54 org.w3c.dom.ProcessingInstruction	
5.27 关于注释 (Comment) 的方法	351	接口	399
5.28 关于 DOM 实现 (DOM Implementation)		5.55 org.w3c.dom.Text 接口	399

第三部分 基于 C 语言的 XDK

第 6 章 基于 C 语言的 XML 解析器	403	6.6 函数索引	405
6.1 Parser API 函数	403	6.7 XSLT API 函数(可扩展样式单语言)	410
6.2 调用顺序	404	6.8 W3C SAX API 函数	412
6.3 内存	404	6.9 W3C DOM API 函数	417
6.4 线程安全	404	6.10 名字空间 API 函数	452
6.5 数据类型索引	404		

第四部分 基于 C++包的 XDK

第 7 章 基于 C++的 XML 解析器	459	7.14 类: ProcessingInstruction	481
7.1 类: Attr	459	7.15 类: Text	482
7.2 类: CDATASection	460	7.16 类: XMLParser	483
7.3 类: Comment	460	7.17 C++ SAX API	487
7.4 类: Document	460	7.18 C++ DOM API 函数	491
7.5 类: DocumentType	464	第 8 章 基于 C++的 Oracle XML 类生成器	
7.6 类: DOMImplementation	465	V1.0.0.0.0	492
7.7 类: Element	465	8.1 基于 C++的 XML 类生成器概述	492
7.8 类: Entity	468	8.2 输入	492
7.9 类: EntityReference	469	8.3 输出	493
7.10 类: NameNodeMap	470	8.4 相关的 XML 标准	493
7.11 类: Node	471	8.5 程序举例	493
7.12 类: NodeList	479	8.6 关于 XML 类生成器的类	493
7.13 类: Notation	480	8.7 由类生成器生成的类	494

第五部分 XML-SQL 通用 (XML-SQL Utility, XSU) 包

第 9 章 Oracle 中 XSU 的 Java API 函数	501	第 10 章 XSU 的 PL/SQL API	519
9.1 oracle.xml.sql.dml.OracleXMLSave	501	10.1 DBMS_XMLQuery	519
9.2 oracle.xml.sql.query.OracleXMLQuery	506	10.2 函数和过程索引	520
9.3 oracle.xml.sql.OracleXMLSQLException	515	10.3 函数和过程	523
9.4 oracle.xml.sql.OracleXMLSQLExceptionNo-		10.4 DBMX_XMLSave	530
RowsException	517	10.5 函数和过程	533

第一部分

基于 JAVA 包的 XML 解析器

本部分包含内容如下：

- 第 1 章 “基于 JAVA 的 XML 解析器”
- 第 2 章 “基于 JAVA 的 XML 类生成器”
- 第 3 章 “XSQL Servlet”
- 第 4 章 “Transviewer Beans”

第 1 章 基于 JAVA 的 XML 解析器

1.1 说 明

类简介	
接口	
NSName	该接口为元素和属性名提供了名字空间支持
NSResolver	该接口为解析名字空间提供了支持
XMLDocumentHandler	该接口扩展了 org.xml.sax.DocumentHandler 接口
XMLToken	XMLToken 的基本接口
类	
AttrDecl	该类保存了在属性表中声明的每种属性的信息, 这些属性存在于文档类型定义 (DTD) 中
Oracle.xml.parser.v2 包	该类实现了 XMLDocumentHandler 接口的缺省行为。
DOMParser	该类按照 W3C (World Wide Web Consortium) 推荐标准实现了可扩展标记语言 (eXtensible Markup Language, XML) 解析器
DTD	该类用于实现 DOM 文档类型接口并且保存了 XML 文档的文档类型定义 (Document Type Definition) 信息
ElementDecl	该类表示了文档类型定义 (DTD) 中的元素声明
NodeFactory	该类指定了创建各种 DOM 树接点的方法, 其中 DOM 树是在解析过程中建立的
oraxsl	Oraxsl 类提供了命令行接口, 从而可以在多 XML 文档中应用样式单
SAXAttrList	该类实现了 SAX(Simple API XML)AttributeList 接口并且能够提供名字空间支持
SAXParser	该类按照 W3C 推荐标准实现了可扩展标记语言 (XML) 1.0 的 SAX 解析器
XMLAttr	该类实现了 DOM 属性接口并且保存了元素的每一种属性信息
XMLCDATA	该类实现了 DOM CDATASection 接口
XMLComment	该类实现了 DOM Comment 接口
XMLDocument	该类实现了 DOM 文档接口, 表示了完整的 XML 文档并且作为文档对象模型 (DOM) 树的根节点而使用
XMLDocumentFragment	该类实现了 DOM DocumentFragment 接口
XMLElement	该类实现了 DOM Element 接口
XMLEntityReference	
XMLNode	该类实现了 DOM Node 的接口, 并且作为完整文档对象模型 (DOM) 的基本数据类型而使用

类简介	
XMLParser	该类作为 DOMParser 类和 SAXParser 类的基类而使用
XMLPI	该类实现了 DOM Processing Instruction 接口
XMLText	该类实现了 DOM Text 接口
XMLTokenizer	该类按照 W3C 推荐标准实现了可扩展标记语言 (XML) 1.0 的 SAX 解析器
XSLProcessor	该类提供了一种转换 XML 文档的方法, 该方法使用先前构造的 XSLStyleSheet 类进行转换操作
XSLStyleSheet	该类保存了诸如模板、关键字、变量和属性集的 XSL 样式单信息
异常	
XMLParseException	该异常表示了一个 XML 文档的过程中发生的解析异常
XSLException	该异常表示了一个在 XSL 转换过程中发生的异常

1.2 AttrDecl

语法定义和继承关系

```
public class AttrDecl extends XMLNode implements oracle.xml.parser.v2.XMLConstants,
java.io.Serializable
    java.lang.Object
    |
    +--XMLNode
        |
        +--oracle.xml.parser.v2.AttrDecl
```

已经实现的所有接口

```
java.lang.Cloneable, org.w3c.dom.Node, java.io.Serializable, oracle.xml.parser.v2.XML
Constants
```

说明:

该类包含了在文档类型定义中所有的属性信息, 这些属性在一个属性表中声明。

成员简介	
字段	
CDATA	属性类型—字符串型—CDATA
DEFAULT	属性表示—Default
ENTITIES	属性类型—记号类型—Entities
ENTITY	属性类型—记号类型—Entity
ENUMERATION	属性类型—枚举型—Enumeration
FIXED	属性表示—Fixed

成员简介

ID	属性类型—记号类型—ID
IDREF	属性类型—记号类型—ID reference
IDREFS	属性类型—记号类型—ID references
IMPLIED	属性表示—Implied
NMTOKEN	属性类型—记号类型—Name token
NMTOKENS	属性类型—记号类型—Name tokens
NOTATION	属性类型—枚举型—Notation
REQUIRED	属性表示—Required
方法	
getAttrPresence()	获取属性表示
getAttrType()	获取属性类型
getDefaultValue()	获取属性缺省值
getEnumerationValues()	获取属性值

继承成员简介

从 XMLNode 类继承的字段:

AMP, ASTERISK, ATTRDECL, cANY, cATTLIST, cCDATA, cCDATAEND, cCDATASTART, cCOMMENTEND, cCOMMENTSTART, cDECCREF, cDECLSTART, cDOCTYPE, cELEMENT, cEMPTY, cEMPTYTAGEND, cENCODING, cENDTAGSTART, cENTITIES, cENTITY, cFIXED, cHEXCREF, cID, cIDREF, cIDREFS, cIGNORE, cIMPLIED, cINCLUDE, cNDATA, cNMTOKEN, cNMTOKENS, cNOTATION, COLON, COMMA, cPIEND, cPISTART, cPUBLIC, cREQUIRED, cSTANDALONE, cSYSTEM, cVERSION, cXML, DOUBLEQUOTE, ELEMENTDECL, EOF, EQ, ERROR, FATAL_ERROR, FDIGIT, FLETTER, FMISCNAME, FSTARTNAME, FWHITESPACE, HASH, ICOUNT, ISTART, LEFTSQB, LPAREN, nameCDATA, nameCOMMENT, nameDOCUMENT, nameDOCUMENTFRAGMENT, name-ENCODING, nameNameSpace, nameSpaceSeparator, nameSTANDALONE, nameTEXT, nameVERSION, nameXML, nameXMLLang, nameXMLNamespace, nameXMLNSNamespace, nameXMLSpace, nameXSLPI, NONVALIDATING, OR, PERCENT, PLUS, QMARK, QUOTE, RIGHTSQB, RPAREN, SEMICOLON, SLASH, TAGEND, TAGSTART, VALIDATING, WARNING

从 oracle.xml.parser.v2.XMLContants 接口继承的字段:

AMP, ASTERISK, cANY, cATTLIST, cCDATA, cCDATAEND, cCDATASTART, cCOMMENTEND, cCOMMENTSTART, cDECCREF, cDECLSTART, cDOCTYPE, cELEMENT, cEMPTY, cEMPTYTAGEND, cENCODING, cENDTAGSTART, cENTITIES, cENTITY, cFIXED, cHEXCREF, cID, cIDREF, cIDREFS, cIGNORE, cIMPLIED, cINCLUDE, cNDATA, cNMTOKEN, cNMTOKENS, cNOTATION, COLON, COMMA, cPIEND, cPISTART, cPUBLIC, cREQUIRED, cSTANDALONE, cSYSTEM, cVERSION, cXML,

继承成员简介

DOUBLEQUOTE, EOF, EQ, ERROR, FATAL_ERROR, FDIGIT, FLETTER, FMISCNAME, FSTARTNAME, FWHITESPACE, HASH, ICOUNT, ISTART, LEFTSQB, LPAREN, nameCDATA, nameCOMMENT, nameDOCUMENT, nameDOCUMENTFRAGMENT, nameENCODING, nameNameSpace, nameSpaceSeparator, nameSTANDALONE, nameTEXT, nameVERSION, nameXML, nameXMLLang, nameXMLNamespace, nameXMLNSNamespace, nameXMLSpace, nameXSLPI, NONVALIDATING, OR, PERCENT, PLUS, QMARK, QUOTE, RIGHTSQB, RPAREN, SEMICOLON, SLASH, TAGEND, TAGSTART, VALIDATING, WARNING

从接口节点继承的字段:

ATTRIBUTE_NODE, CDATA_SECTION_NODE, COMMENT_NODE, DOCUMENT_FRAGMENT_NODE, DOCUMENT_NODE, DOCUMENT_TYPE_NODE, ELEMENT_NODE, ENTITY_NODE, ENTITY_REFERENCE_NODE, NOTATION_NODE, PROCESSING_INSTRUCTION_NODE, TEXT_NODE

从XMLNode 类继承的方法:

appendChild(Node), cloneNode(boolean), getAttributes(), getChildNodes(), getFirstChild(), getLastChild(), getNextSibling(), getNodeName(), getNodeType(), getNodeValue(), getOwnerDocument(), getParenNode(), getPreviousSibling(), hasChildNodes(), insertBefore(Node, Node), print(OutputStream), print(OutputStream, String), print(PrintWriter), removeChild(Node), replaceChild(Node, Node), selectNodes(String, NSResolver), selectSingleNode(String, NSResolver), setNodeValue(String), transformNode(XSLStylesheet), valueOf(String, NSResolver)

从 java.lang.Object 类继承的方法:

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

从 Node 接口继承的方法:

appendChild(Node), cloneNode(boolean), getAttributes(), getChildNodes(), getFirstChild(), getLastChild(), getNextSibling(), getNodeName(), getNodeType(), getNodeValue(), getOwnerDocument(), getParenNode(), getPreviousSibling(), hasChildNodes(), insertBefore(Node, Node), removeChild(Node), replaceChild(Node, Node), setNodeValue(String)

字段

CDATA

public static final int CDATA

属性类型—字符串型—CDATA

DEFAULT

public static final int DEFAULT

属性表示—Default

ENTITIES

`public static final int ENTITIES`

属性类型—记号类型—Entities

ENTITY

`public static final int ENTITY`

属性类型—记号类型—Entity

ENUMERATION

`public static final int ENUMERATION`

属性类型—枚举型—Enumeration

FIXED

`public static final int FIXED`

属性表示—Fixed

ID

`public static final int ID`

属性类型—记号类型—ID

IDREF

`public static final int IDREF`

属性类型—记号类型—ID reference

IDREFS

`public static final int IDREFS`

属性类型—记号类型—ID references

IMPLIED

`public static final int IMPLIED`

属性表示—Implied

NMTOKEN

`public static final int NMTOKEN`

属性类型—记号类型—Name token

NMTOKENS

`public static final int NMTOKENS`

属性类型—记号类型—Name tokens

NOTATION

`public static final int NOTATION`

属性类型—枚举型—Notation

REQUIRED

`public static final int REQUIRED`

属性表示—Required

方法

`getAttrPresence()`

`public int getAttrPresence()`

获取属性表示。

返回值

当前属性的表示。

`getAttrType()`

`public int getAttrType()`

获取属性类型。

返回值

当前属性的类型。

`getDefaultValue()`

`public java.lang.String getDefaultValue()`

获取属性缺省值。

返回值

当前属性的缺省值。

`getEnumerationValues()`

`public java.util.Vector getEnumerationValues()`

获取属性值。

返回值

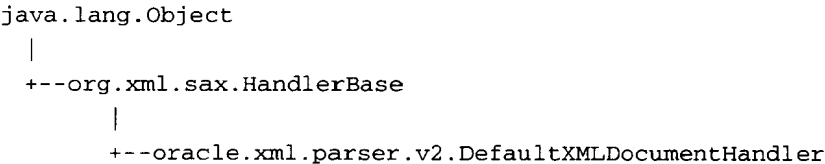
以枚举型方式表示的当前属性的值。

1.3 DefaultXMLDocumentHandler

语法定义和继承关系

`public class DefaultXMLDocumentHandler extends org.xml.sax.HandlerBase implements`

XMLDocumentHandler



直接派生的子类

XMLTokenizer

已经实现的所有接口

org.xml.sax.DocumentHandler, org.xml.sax.DTDHandler, org.xml.sax.EntityResolver,
org.xml.sax.ErrorHandler, XMLDocumentHandler

说明

该类实现了 XMLDocumentHandler 接口的缺省行为。

如果用户只需要实现该接口的部分功能，可以通过扩展该类达到目的。

成员简介

构造函数

DefaultXMLDocumentHandler() 构造一个缺省的文档句柄

方法

cDATASection(char[], int, int) 接收一个 CDATA 部分（CDATA Section）的通知

comment(String) 接收一个注释的通知

endDoctype() 接收 DTD 尾部的通知

endElement(NSName) 接收一个元素尾部的通知

setDoctype(DTD) 接收 DTD 通知

setTextDecl(String, String) 接收一个文本 XML 声明（Text XML Declaration）的通知

setXMLDecl(String, String, String) 接收一个 XML 声明(XML Declaration)的通知

startElement(NSName, SAXAttrList) 接收一个元素头部的通知

继承成员简介

从 HandlerBase 类继承的方法：

characters(char[], int, int), endDocument(), endElement(String), error(SAXParseException), fatalError-
(SAXParseException), ignorableWhitespace(char[], int, int), notationDecl(String, String, String), processing-
Instruction(String, String), resolveEntity(Sting, String), serDocumentLocator(Locator), startDocument(),
startElement(String, AttributeList), unparsedEntityDecl(String, String, String, String), warning(SAXParse-
Exception)
