XML::LibXML Quick Reference Card

Revision 0.1 for XML::LibXML version 1.61

Andrew Ford

The XML::LibXML set of Perl modules provides an interface to the Gnome libxml2 DOM and SAX parser and the DOM tree. The interface is extensive. This quick reference card provides the synopsis of the methods provided. These are fully documented in the manual pages.

XML::LibXML

```
use XML::LibXML qw(tag ...)
```

The ‘libxml’ tag imports the node types: XML_ELEMENT_NODE, XML_ATTRIBUTE_NODE, XML_TEXT_NODE, XML_CDATA_SECTION_NODE, XML_ENTITY_REF_NODE, XML_ENTITY_NODE, XML_PI_NODE, XML_COMMENT_NODE, XML_DOCUMENT_NODE, XML_DOCUMENT_TYPE_NODE, XML_DOCUMENT_FRAG_NODE, XML_NOTATION_NODE, XML_HTML_DOCUMENT_NODE, XML_DTD_NODE, XML_ELEMENT_DECL, XML_ATTRIBUTE_DECL, XML_ENTITY_DECL, XML_NAMESPACE_DECL, XML_XINCLUDE_END, XML_XINCLUDE_START.
The ‘encoding’ tag imports encodeToUTF8 and decodeFromUTF8.
The ‘ns’ tag imports XML_XMLNS_NS and XML_XML_NS.

```
$version_str = XML::LibXML::LIBXML_DOTTED_VERSION
$version_id  = XML::LibXML::LIBXML_VERSION
$DLL_Version = XML::LibXML::LIBXML_RUNTIME_VERSION
$libxmlnode  = XML::LibXML->import_GDOME($node,
$deep)
$gdomenode   = XML::LibXML->export_GDOME($node,
$deep)
```

XML::LibXML::Parser – XML Data Parsing

```
$parser = XML::LibXML->new()
$doc    = $parser->parse_file($xmlfilename)
$doc    = $parser->parse_fh($io_fh)
$doc    = $parser->parse_string($xmlstring)
$doc    = $parser->parse_html_file($htmlfile)
$doc    = $parser->parse_html_fh($io_fh)
$doc    = $parser->parse_html_string($htmlstring)
$frag   = $parser->parse_balanced_chunk($string)
$frag   = $parser->parse_xml_chunk($string)
$parser->process_xincludes($doc)
$parser->processXIncludes($doc)
$parser->parse_chunk($string, $terminate)
$parser->start_push()
$parser->push(@data)
$doc    = $parser->finish_push($recover)
$parser->validation(1)
$parser->recover(1)
$parser->recover_silently(1)
$parser->expand_entities(0)
$parser->keep_blanks(0)
$parser->pedantic_parser(1)
$parser->line_numbers(1)
$parser->load_ext_dtd(1)
$parser->complete_attributes(1)
$parser->expand_xinclude(1)
$parser->load_catalog($catalog_file)
$parser->base_uri($your_base_uri)
$parser->gdome_dom(1)
$parser->clean_namespaces(1)
```

XML::LibXML::Document – DOM Document Class

```
$dom = XML::LibXML::Document->new($version, $encod-
ing)
$dom = XML::LibXML::Document->createDocument($version,
                                              $encoding)
$string  = $doc->encoding()
           $doc->setEncoding($new_encoding)
$string  = $doc->version()
           $doc->standalone
           $doc->setStandalone($numvalue)
$compr   = $doc->compression
           $doc->setCompression($ziplevel)
$string  = $dom->toString($format)
$c14nstr = $doc->toStringC14N($comment_flag,$xpath)
$str     = $doc->serialize($format)
$c14nstr = $doc->serialize_c14n($comment_flag,$xpath)
$state   = $doc->toFile($filename, $format)
$state   = $doc->toFH($fh, $format)
$string  = $doc->toStringHTML()
$string  = $doc->serialize_html()
$bool    = $dom->is_valid()
           $dom->validate()
$root    = $dom->documentElement()
           $dom->setDocumentElement($root)
$element = $dom->createElement($nodename)
$element = $dom->createElementNS($namespaceURI,
$qname)
$text     = $dom->createTextNode($content_text)
$comment  = $dom->createComment($comment_text)
$attrnode = $doc->createAttribute($name [,$value])
$attrnode = $doc->createAttributeNS(namespaceURI,
$name [,$value])
$fragment = $doc->createDocumentFragment()
$cdata    = $dom->create($cdata_content)
$pi       = $doc->createProcessingInstruction($target,
$data)
$entref   = $doc->createEntityReference($refname)
$dtd      = $doc->createInternalSubset($root, $pub,
$sys)
$dtd      = $doc->createExternalSubset($root, $pub,
$sys)
           $doc->importNode($node)
           $doc->adoptNode($node)
$dtd      = $doc->externalSubset
```

```
$dtd      = $doc->internalSubset
            $doc->setExternalSubset($dtd)
            $doc->setInternalSubset($dtd)
$dtd      = $doc->removeExternalSubset()
$dtd      = $doc->removeInternalSubset()
@nodelist = $doc->getElementsByTagName($tagname)
@nodelist = $doc->getElementsByTagName($nsURI,$tagname)
@nodelist = $doc->getElementsByLocalName($localname)
$node     = $doc->getElementById($id)
            $dom->indexElements()
```

refcards.com™

XML::LibXML::Node – abstract base class for nodes

```
$name     = $node->nodeName
            $node->setNodeName($newName)
$bool     = $node->isSameNode($other_node)
$bool     = $node->isEqual($other_node)
$content  = $node->nodeValue
$content  = $node->textContent
$type     = $node->nodeType
            $node->unbindNode()
$child    = $node->removeChild($childnode)
$oldnode  = $node->replaceChild($newNode, $oldNode)
            $node->replaceNode($newNode)
$child    = $node->appendChild($childnode)
$child    = $node->addChild($chilnode)
$node     = $parent->addNewChild($nsURI, $name)
            $node->addSibling($newNode)
$newnode  = $node->cloneNode($deep)
$parent   = $node->parentNode
$next     = $node->nextSibling()
$prev     = $node->previousSibling()
$bool     = $node->hasChildNodes()
$child    = $node->firstChild
$child    = $node->lastChild
$docnode  = $node->ownerDocument
$node     = $node->getOwner
            $node->setOwnerDocument($doc)
            $node->insertBefore($newNode, $refNode)
            $node->insertAfter($newNode, $refNode)
@nodes    = $node->findnodes($xpath_expression)
$result   = $node->find($xpath)
$string   = $node->findvalue($xpath)
@nodes    = $node->childNodes
$xmlstr   = $node->toString($format,$docencoding)
$c14nstr  = $node->toString($with_comments,
$xpath_expression)
$string   = $doc->serialze($format)
$c14nstr  = $doc->serialize_c14n($comment_flag,$xpath)
$localname = $node->localname
$nameprefix = $node->prefix
$uri      = $node->namespaceURI()
$bool     = $node->hasAttributes()
@attrs    = $node->attributes()
$uri      = $node->lookupNamespaceURI($prefix)
$prefix   = $node->lookupNamespacePrefix($URI)
$iter     = $node->iterator
            $node->normalize
```

```
@nslist  = $node->getNamespaces
           $node->removeChildNodes()
           $node->nodePath()
$line_no = $node->line_number()
```

XML::LibXML::Element – element node class

```
$node    = XML::LibXML::Element->new($name)
           $node->setAttribute($aname, $avalue)
           $node->setAttributeNS($nsURI, $aname,
$avalue)
$avalue  = $node->getAttribute($aname)
$avalue  = $node->setAttributeNS($nsURI, $aname)
$node    = $node->getAttributeNode($aname)
$node    = $node->getAttributeNodeNS($namespaceURI,
$aname)
           $node->removeAttribute($aname)
           $node->removeAttributeNS($nsURI, $aname)
$bool    = $node->hasAttribute($aname)
$bool    = $node->hasAttributeNS($nsURI, $aname)
@nodes   = $node->getChildrenByTagName($tagname)
@nodes   = $node->getChildrenByTagNameNS($nsURI,$tagname)
@nodes   = $node->getChildrenByLocalName($localname)
@nodes   = $node->getElementsByTagName($tagname)
@nodes   = $node->getElementsByTagNameNS($nsURI,$localname)
@nodes   = $node->getElementsByLocalName($localname)
$node->appendWellBalancedChunk($chunk)
$node->appendText($PCDATA)
$node->appendTextNode($PCDATA)
$node->appendTextChild($childname , $PCDATA)
$node->setNamespace($nsURI , $nsPrefix, $activate)
$node->setNamespaceDeclURI($nsPrefix, $newURI)
$node->setNamespaceDeclPrefix($oldPrefix, $newPre-
fix)
```

XML::LibXML::Attr – attribute class

```
$attr = XML::LibXML::Attr->new($name [,$value])
$string = $attr->getValue()
$value = $attr->value
$attr->setValue($string)
$node = $attr->getOwnerElement()
$attr->setNamespace($nsURI, $prefix)
$bool = $attr->isId
```

Text, Comment and CDATA Section nodes

```
$node = XML::LibXML::Text->new($content)
$node = XML::LibXML::Comment->new($content)
$node = XML::LibXML::CDATASection-new($content)
$nodedata = $text->data
$text->setData($text_content)
$text->substringData($offset, $length)
$text->appendData($somedata)
$text->insertData($offset, $string)
$text->deleteData($offset, $length)
$text->deleteDataString($remstring, $all)
$text->replaceData($offset, $length, $string)
$text->replaceDataString($old, $new, $flag)
$text->replaceDataRegEx($search_cond, $replace_cond,
$reflags)
```

XML::LibXML::Namespace – namespace implementation

```
$ns     = XML::LibXML::Namespace->new($nsURI)
$string = $ns->getName()
$string = $ns->name()
$string = $ns->prefix()
$lname  = $ns->getLocalName()
$string = $ns->getData()
$string = $ns->getValue()
$string = $ns->value()
$string = $ns->uri()
$uri    = $ns->getNamespaceURI()
$prefix = $ns->getPrefix()
```

Processing Instruction handling

```
$pinode->setData($data_string )
$pinode->setData(name => $value [...])
```

DTD Handling

```
$dtd = XML::LibXML::Dtd->new($pubid, $sysid)
$dtd = XML::LibXML::Dtd->parse_string($dtdstr)
$publicId = $dtd->getName()
$publicId = $dtd->publicId()
$systemId = $dtd->systemId()
```

XML::LibXML Quick Reference Card
Revision 0.1 for XML::LibXML version 1.61 [October 2006]
A refcards.com™ quick reference card.

refcards.com is a trademark of Ford & Mason Ltd.
Published by Ford & Mason Ltd.