Seleniet XPATH Locator QuickRef

Author(s)	Thomas Eitzenberger
Version	0.2
Status	Ready for review

Content

Selecting Nodes

XPath uses path expressions to select nodes in an XML document. The node is selected by following a path or steps. The most useful path expressions are listed below:

Expression	Description
nodename	Selects all nodes with the name "*nodename*"
/	Selects from the root node
//	Selects nodes in the document from the current node that match the selection no matter where they are
.	Selects the current node
..	Selects the parent of the current node
@	Selects attributes

In the table below we have listed some path expressions and the result of the expressions:

Path Expression	Result
bookstore	Selects all nodes with the name "bookstore"
/bookstore	Selects the root element bookstore **Note:** If the path starts with a slash (/) it always represents an absolute path to an element!
bookstore/book	Selects all book elements that are children of bookstore
//book	Selects all book elements no matter where they are in the document
bookstore//book	Selects all book elements that are descendant of the bookstore element, no matter where they are under the bookstore element
//@lang	Selects all attributes that are named lang

Predicates

Predicates are used to find a specific node or a node that contains a specific value.

Predicates are always embedded in square brackets.

In the table below we have listed some path expressions with predicates and the result of the expressions:

Path Expression	Result
/bookstore/book[1]	Selects the first book element that is the child of the bookstore element. **Note:** In IE 5,6,7,8,9 first node is[0], but according to W3C, it is [1]. To solve this problem in IE, set the SelectionLanguage to XPath: *In JavaScript:* *xml*.setProperty("SelectionLanguage","XPath");
/bookstore/book[last()]	Selects the last book element that is the child of the bookstore element
/bookstore/book[last()-1]	Selects the last but one book element that is the child of the bookstore element
/bookstore/book[position()<3]	Selects the first two book elements that are children of the bookstore element
//title[@lang]	Selects all the title elements that have an attribute named lang
//title[@lang='en']	Selects all the title elements that have a "lang" attribute with a value of "en"
/bookstore/book[price>35.00]	Selects all the book elements of the bookstore element that have a price element with a value greater than 35.00
/bookstore/book[price>35.00]/title	Selects all the title elements of the book elements of the bookstore element that have a price element with a value greater than 35.00

Selecting Unknown Nodes

XPath wildcards can be used to select unknown XML nodes.

Wildcard	Description
*	Matches any element node
@*	Matches any attribute node
node()	Matches any node of any kind

In the table below we have listed some path expressions and the result of the expressions:

Path Expression	Result
/bookstore/*	Selects all the child element nodes of the bookstore element
//*	Selects all elements in the document
//title[@*]	Selects all title elements which have at least one attribute of any kind

Selecting Several Paths

By using the | operator in an XPath expression you can select several paths.

In the table below we have listed some path expressions and the result of the expressions:

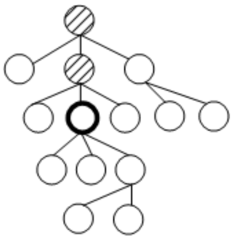
Path Expression	Result
//book/title | //book/price	Selects all the title AND price elements of all book elements
//title | //price	Selects all the title AND price elements in the document
/bookstore/book/title | //price	Selects all the title elements of the book element of the bookstore element AND all the price elements in the document

XPath Operators

Below is a list of the operators that can be used in XPath expressions:

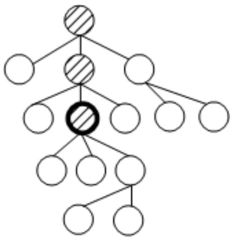
Operator	Description	Example
|	Computes two node-sets	//book | //cd
+	Addition	6 + 4
-	Subtraction	6 - 4
*	Multiplication	6 * 4
div	Division	8 div 4
=	Equal	price=9.80
!=	Not equal	price!=9.80
<	Less than	price<9.80
<=	Less than or equal to	price<=9.80
>	Greater than	price>9.80
>=	Greater than or equal to	price>=9.80
or	or	price=9.80 or price=9.70
and	and	price>9.00 and price<9.90
mod	Modulus (division remainder)	5 mod 2

XPath Axis Specifiers



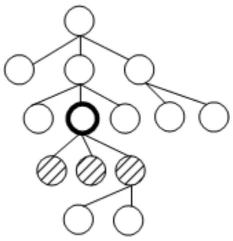
ancestor axis

selects all ancestor nodes from the active context node on on the path to the root node.
Using inverse order.



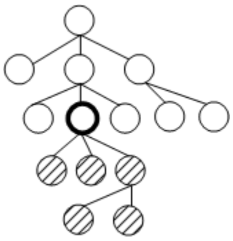
ancestor-or-self axis

selects all ancestor nodes from the active context node on on the shortest path to the root node and the context node itself.
Using inverse order.



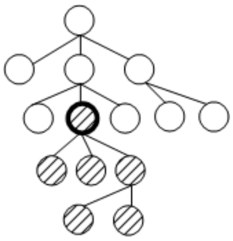
child axis

selects all child nodes of the active context node.
Selected nodes are in document order.



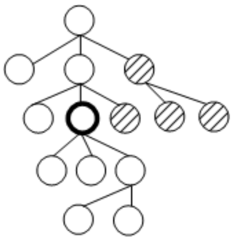
descendant axis

selects all descendant nodes of the active context node.
Selected nodes are in document order.



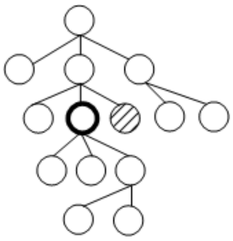
descendant-or-self axis

selects all descendant nodes of the active context node and the context node itself..
Selected nodes are in document order.



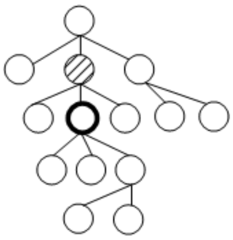
following axis

selects all nodes following the active context node excluding the nodes on the descendant axis.
Selected nodes are in document order.



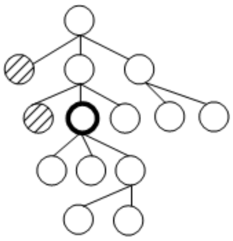
following-sibling axis

selects all sibling nodes following the active context node.
Selected nodes are in document order.



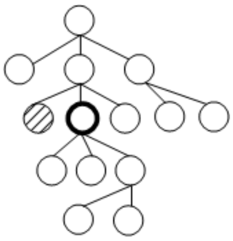
parent axis

selects the parent node of the active context node.



preceding axis

selects all nodes preceding the active context node excluding the nodes on the ancestor axis.
Selected nodes are in inverse order.



preceding-sibling axis

selects all sibling nodes preceding the active context node.
Selected nodes are in inverse order.

XPath 1.0 Functions

Node-Set Functions

Function	Description
number last()	Returns the position of the last node in the context list.
number position()	Returns the position of the current context node.
number count(node-set)	Returns the number of nodes in the node-set.
node-set id(object)	Retruns the element specified by it's unique id.
string local-name(node-set)	Returns the local-name for the first node in the node-set.
string local-name()	Returns the local-name for the context node.
string namespace-uri(node-set)	Returns the namespace-uri for the first node in the node-set.
string namespace-uri()	Returns the namespace-uri for the context node.
string name(node-set)	Returns the name for the first node in the node-set.
string name()	Returns the name for the context node.

String Functions

Function	Description
string string()	Returns a string value representation of the context node.
string string(object)	Returns the string representation of the object argument.
string concat(string, string, string*)	Returns the concatenation of its arguments.
boolean starts-with(string, string)	Returns true if the first argument string starts with the second argument string.
boolean contains(string, string)	Returns true if the first argument string contains the second argument string.
string substring-before(string, string)	Returns the substring of the first argument string that comes before the first occurrence of the second argument.
string substring-after(string, string)	Returns the substring of the first argument string that comes after the first occurrence of the second argument.
string substring(string, number,	Returns the substring of the first argument starting at the

number)	position specified by the second argument and the length specified by the third argument.
string substring(string, number)	Returns the substring of the first argument from the position specified by the second argument.
number string-length(string)	Returns the length of the string specified by the argument.
number string-length()	Returns the length of the string specified by the context node.
string normalize-space(string)	Returns a white-space normalized string specified by the argument.
string normalize-space()	Returns a white-space normalized string specified by the context-node.
string translate(string, string, string)	Replaces characters in the string specified by the second argument with characters specified by the third argument.

Boolean Functions

Function	Description
boolean boolean(object)	Returns the boolean representation of the object argument.
boolean not(boolean)	Returns a boolean with the opposite value of its argument.
boolean true()	Returns a boolean with the value of true.
boolean false()	Returns a boolean with the value of false.
boolean lang(string)	Returns true if the language of the context node is the same as the languag specified by the argument.

Number Functions

Function	Description
number number(object)	Returns the number representation of the object argument.
number number()	Returns the number representation of the context node.
number sum(node-set)	Returns the sum of all nodes in the node-set.
number floor(number)	Returns the largest integer value not greater than the argument.
number ceiling(number)	Returns the smallest integer value not less than the argument.

number round(number)	Returns the integer value closest to the argument.