Documentation for class XML

public class XML - Class for accessing XML data through the XPath language.

This class offers methods for accessing the nodes of a XML document using the XPath language. You can add or remove nodes, set or modify their content and their attributes. No additional PHP extensions like DOM XML or something similar are required to use these features.

Links:	**Latest release of this class**, **W3C XPath**
Copyright:	Copyright (c) 2001 Michael P. Mehl. All rights reserved.
Author(s):	Michael P. Mehl <**mpm@phpxml.org**>
Version:	1.0 (2001-03-08)

Public Methods

void **XML ([string $file])**
Constructor of the class.

void **load_file (string $file)**
Reads a file and parses the XML data.

string **get_file ([array $highlight, string $root, integer**
Generates a XML file with the content of the current document.

string **add_node (string $context, string $name)**
Adds a new node to the XML document.

void **remove_node (string $node)**
Removes a node from the XML document.

void **add_content (string $path, string $value)**
Add content to a node.

void **set_content (string $path, string $value)**
Set the content of a node.

string **get_content (string $path)**
Retrieves the content of a node.

void **add_attributes (string $path, array $attributes)**
Add attributes to a node.

void **set_attributes (string $path, array $attributes)**
Sets the attributes of a node.

array **get_attributes (string $path)**
Retrieves a list of all attributes of a node.

string **get_name (string $path)**
Retrieves the name of a document node.

array **evaluate (string $path, [string $context])**

Evaluates an XPath expression.

Private Methods

void	**handle_start_element (int $parser, string $name, array $attributes)** Handles opening XML tags while parsing.
void	**handle_end_element (int $parser, string $name)** Handles closing XML tags while parsing.
void	**handle_character_data (int $parser, string $text)** Handles character data while parsing.
array	**split_paths (string $expression)** Splits an XPath expression into its different expressions.
array	**split_steps (string $expression)** Splits an XPath expression into its different steps.
array	**get_axis (string $step, string $node)** Retrieves axis information from an XPath expression step.
int	**search_string (string $term, string $expression)** Looks for a string within another string.
boolean	**is_function (string $expression)** Checks for a valid function name.
array	**evaluate_step (string $context, array $steps)** Evaluates a step of an XPath expression.
mixed	**evaluate_function (string $function, string $arguments, string** Evaluates an XPath function
mixed	**evaluate_predicate (string $node, string $predicate)** Evaluates a predicate on a node.
array	**check_predicates (array $nodes, array $predicates)** Checks whether a node matches predicates.
boolean	**check_node_test (string $context, string $node_test)** Checks whether a node matches a node-test.
array	**handle_axis_child (array $axis, string $context)** Handles the XPath child axis.
array	**handle_axis_parent (array $axis, string $context)** Handles the XPath parent axis.
array	**handle_axis_attribute (array $axis, string $context)** Handles the XPath attribute axis.
array	**handle_axis_self (array $axis, string $context)** Handles the XPath self axis.

array	**handle_axis_descendant (array $axis, string $context)** Handles the XPath descendant axis.
array	**handle_axis_ancestor (array $axis, string $context)** Handles the XPath ancestor axis.
array	**handle_axis_namespace (array $axis, string $context)** Handles the XPath namespace axis.
array	**handle_axis_following (array $axis, string $context)** Handles the XPath following axis.
array	**handle_axis_preceding (array $axis, string $context)** Handles the XPath preceding axis.
array	**handle_axis_following_sibling (array $axis, string $context)** Handles the XPath following-sibling axis.
array	**handle_axis_preceding_sibling (array $axis, string $context)** Handles the XPath preceding-sibling axis.
array	**handle_axis_descendant_or_self (array $axis, string $context)** Handles the XPath descendant-or-self axis.
array	**handle_axis_ancestor_or_self (array $axis, string $context)** Handles the XPath ancestor-or-self axis.
mixed	**handle_function_last (string $node, string $arguments)** Handles the XPath function last.
mixed	**handle_function_position (string $node, string $arguments)** Handles the XPath function position.
mixed	**handle_function_count (string $node, string $arguments)** Handles the XPath function count.
mixed	**handle_function_id (string $node, string $arguments)** Handles the XPath function id.
mixed	**handle_function_name (string $node, string $arguments)** Handles the XPath function name.
mixed	**handle_function_string (string $node, string $arguments)** Handles the XPath function string.
mixed	**handle_function_concat (string $node, string $arguments)** Handles the XPath function concat.
mixed	**handle_function_starts_with (string $node, string $arguments)** Handles the XPath function starts-with.
mixed	**handle_function_contains (string $node, string $arguments)** Handles the XPath function contains.
mixed	**handle_function_substring_before (string $node, string $arguments)** Handles the XPath function substring-before.
mixed	**handle_function_substring_after (string $node, string $arguments)**

Handles the XPath function substring-after.

mixed **handle_function_substring (string $node, string $arguments)**
Handles the XPath function substring.

mixed **handle_function_string_length (string $node, string $arguments)**
Handles the XPath function string-length.

mixed **handle_function_translate (string $node, string $arguments)**
Handles the XPath function translate.

mixed **handle_function_boolean (string $node, string $arguments)**
Handles the XPath function boolean.

mixed **handle_function_not (string $node, string $arguments)**
Handles the XPath function not.

mixed **handle_function_true (string $node, string $arguments)**
Handles the XPath function true.

mixed **handle_function_false (string $node, string $arguments)**
Handles the XPath function false.

mixed **handle_function_lang (string $node, string $arguments)**
Handles the XPath function lang.

mixed **handle_function_number (string $node, string $arguments)**
Handles the XPath function number.

mixed **handle_function_sum (string $node, string $arguments)**
Handles the XPath function sum.

mixed **handle_function_floor (string $node, string $arguments)**
Handles the XPath function floor.

mixed **handle_function_ceiling (string $node, string $arguments)**
Handles the XPath function ceiling.

mixed **handle_function_round (string $node, string $arguments)**
Handles the XPath function round.

mixed **handle_function_text (string $node, string $arguments)**
Handles the XPath function text.

string **prestr (string $string, string $delimiter)**
Retrieves a substring before a delimiter.

string **afterstr (string $string, string $delimiter)**
Retrieves a substring after a delimiter.

void **display_error (string $message)**
Displays an error message.

Private Fields

array	**$nodes** List of all document nodes.
array	**$ids** List of document node IDs.
string	**$path** Current document path.
integer	**$position** Current document position.
string	**$root** Path of the document root.
string	**$xpath** Current XPath expression.
array	**$entities** List of entities to be converted.
array	**$axes** List of supported XPath axes.
array	**$functions** List of supported XPath functions.
array	**$operators** List of supported XPath

Method Details: XML

public void XML ([string $file])

This constructor initializes the class and, when a filename is given, tries to read and parse the given file.

Parameter:

string $file Path and name of the file to read and

Return value:

void

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

load_file()

Method Details: load_file

public void load_file (string $file)

This method reads the content of a XML file, tries to parse its content and upon success stores the information retrieved from the file into an array.

Parameter:

string	$file	Path and name of the file to be read and

Return value:

void

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

handle_start_element(), **handle_end_element()**, **handle_character_data()**

Method Details: get_file

public string get_file ([array $highlight, string $root, integer $level])

This method creates a string containing the XML data being read and modified by this class before. This string can be used to save a modified document back to a file or doing other nice things with it.

Parameter:

array	$highlight	Array containing a list of full document paths of nodes to be highlighted by <font>...</font> tags in the generated XML string.
string	$root	While doing a recursion with this method, this parameter is used for internal purpose.
integer	$level	While doing a recursion with this method, this parameter is used for internal purpose.

Return value:

string	The returned string contains well-formed XML data representing the content of this document.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

load_file(), **evaluate()**, **get_content()**

Method Details: add_node

public string add_node (string $context, string $name)

This method adds a new node to the tree of nodes of the XML document being handled by this class. The new node is created according to the parameters passed to this method.

Parameter:

string $context	Full path of the parent, to which the new node should be added as a child.
string $name	Name of the new node.

Return value:

string	The string returned by this method will contain the full document path of the created node.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

remove_node(), **evaluate()**

Method Details: remove_node

public void remove_node (string $node)

This method removes a node from the tree of nodes of the XML document. If the node is a document node, all children of the node and its character data will be removed. If the node is an attribute node, only this attribute will be removed, the node to which the attribute belongs as well as its children will remain unmodified.

Parameter:

string $node	Full path of the node to be

Return value:

void

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

add_node(), **evaluate()**

Method Details: add_content

public void add_content (string $path, string $value)

This method adds content to a node. If it's an attribute node, then the value of the attribute will be set, otherwise the character data of the node will be set. The content is appended to existing content, so nothing will be overwritten.

Parameter:

string $path	Full document path of the node.
string $value	String containing the content to be

Return value:

void

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

get_content(), **evaluate()**

Method Details: set_content

public void set_content (string $path, string $value)

This method sets the content of a node. If it's an attribute node, then the value of the attribute will be set, otherwise the character data of the node will be set. Existing content will be overwritten.

Parameter:

string	$path	Full document path of the node.
string	$value	String containing the content to be

Return value:

void

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

get_content(), **evaluate()**

Method Details: get_content

public string get_content (string $path)

This method retrieves the content of a node. If it's an attribute node, then the value of the attribute will be retrieved, otherwise it'll be the character data of the node.

Parameter:

string	$path	Full document path of the node, from which the content should be retrieved.

Return value:

string	The returned string contains either the value or the character data of the node.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

set_content(), **evaluate()**

Method Details: add_attributes

public void add_attributes (string $path, array $attributes)

This method adds attributes to a node. Existing attributes will not be overwritten.

Parameter:

string	$path	Full document path of the node, the attributes should be added
array	$attributes	Associative array containing the new attributes for the node.

Return value:

void

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

set_content(), **get_content()**

Method Details: set_attributes

public void set_attributes (string $path, array $attributes)

This method sets the attributes of a node and overwrites all existing attributes by doing this.

Parameter:

string	$path	Full document path of the node, the attributes of which should be set.
array	$attributes	Associative array containing the new attributes for the node.

Return value:

void

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

set_content(), **get_content()**

Method Details: get_attributes

public array get_attributes (string $path)

This method retrieves a list of all attributes of the node specified in the argument.

Parameter:

string	$path	Full document path of the node, from which the list of attributes should be retrieved.

Return value:

array	The returned associative array contains the all attributes of the specified node.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

get_content(), **$nodes**, **$ids**

Method Details: get_name

public string get_name (string $path)

This method retrieves the name of document node specified in the argument.

Parameter:

string $path	Full document path of the node, from which the name should be retrieved.

Return value:

string	The returned array contains the name of the specified

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

get_content(), **$nodes**, **$ids**

Method Details: evaluate

public array evaluate (string $path, [string $context])

This method tries to evaluate an XPath expression by parsing it. A XML document has to be read before this method is able to work.

Parameter:

string $path	XPath expression to be evaluated.
string $context	Full path of a document node, starting from which the XPath expression should be evaluated.

Return value:

array	The returned array contains a list of the full document paths of all nodes match the evaluated XPath expression.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

$nodes, **$ids**

Method Details: handle_start_element

private void handle_start_element (int $parser, string $name, array $attributes)

While parsing a XML document for each opening tag this method is called. It'll add the tag found to the tree of document nodes.

Parameter:

int	$parser	Handler for accessing the current XML parser.
string	$name	Name of the opening tag found in the document.
array	$attributes	Associative array containing a list of all attributes of the tag found in the document.

Return value:

void

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

handle_end_element(), **handle_character_data()**, **$nodes**, **$ids**

Method Details: handle_end_element

private void handle_end_element (int $parser, string $name)

While parsing a XML document for each closing tag this method is called.

Parameter:

int	$parser	Handler for accessing the current XML parser.
string	$name	Name of the closing tag found in the

Return value:

void

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

handle_start_element(), **handle_character_data()**, **$nodes**, **$ids**

Method Details: handle_character_data

private void handle_character_data (int $parser, string $text)

While parsing a XML document for each character data this method is called. It'll add the character data to the document tree.

Parameter:

int	$parser	Handler for accessing the current XML
string	$text	Character data found in the document.

Return value:

void

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

handle_start_element(), **handle_end_element()**, **$nodes**, **$ids**

Method Details: split_paths

private array split_paths (string $expression)

This method splits an XPath expression. Each expression can consists of list of expression being separated from each other by a | character.

Parameter:

string	$expression	The complete expression to be splitted into its different expressions.

Return value:

array	The array returned from this method contains a list of all expressions found in the expression passed to this method as a parameter.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

Method Details: split_steps

private array split_steps (string $expression)

This method splits an XPath expression. Each expression can consists of list of being separated from each other by a / character.

Parameter:

string	$expression	The complete expression to be splitted into its different

Return value:

array	The array returned from this method contains a list of all steps found in the expression passed to this method as a parameter.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

Method Details: get_axis

private array get_axis (string $step, string $node)

This method tries to extract the name of the axis and its node-test from a given step of an XPath expression at a given node.

Parameter:

string	$step	String containing a step of an XPath expression.
string	$node	Full document path of the node on which the step is

Return value:

array	This method returns an array containing information about the axis found in the step.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate_step()

Method Details: search_string

private int search_string (string $term, string $expression)

This method looks for a string within another string. Brackets in the string the method is looking through will be respected, which means that only if the string the method is looking for is located outside of brackets, the search will be successful.

Parameter:

string	$term	String in which the search shall take
string	$expression	String that should be searched.

Return value:

int	This method returns -1 if no string was found, otherwise the offset at which the string was found.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate_step()

Method Details: is_function

private boolean is_function (string $expression)

This method check whether an expression contains a valid name of an XPath function.

Parameter:

string $expression Name of the function to be

Return value:

boolean This method returns true if the given name is a valid XPath function name, otherwise false.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: evaluate_step

private array evaluate_step (string $context, array $steps)

This method tries to evaluate a step from an XPath expression at a specific context.

Parameter:

string $context Full document path of the context from which starting the step should be evaluated.

array $steps Array containing the remaining steps of the current XPath expression.

Return value:

array This method returns an array containing all nodes that are the result of evaluating the given XPath step.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: evaluate_function

private mixed evaluate_function (string $function, string $arguments, string $node)

This method evaluates a given XPath function with its arguments on a specific node of the document.

Parameter:

string $function Name of the function to be evaluated.

string $arguments String containing the arguments being passed to the function.

string $node Full path to the document node on which the function should be evaluated.

Return value:

mixed This method returns the result of the evaluation of the function. Depending on the function the type of the return value can be different.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: evaluate_predicate

private mixed evaluate_predicate (string $node, string $predicate)

This method tries to evaluate a predicate on a given node.

Parameter:

string $node	Full path of the node on which the predicate should be evaluated.
string $predicate	String containing the predicate expression to be evaluated.

Return value:

mixed	This method is called recursively. The first call should return a boolean value, whether the node matches the predicate or not. Any call to the method being made during the recursion may also return other types for further processing.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: check_predicates

private array check_predicates (array $nodes, array $predicates)

This method checks whether a list of nodes passed to this method match a given list of predicates.

Parameter:

array $nodes	Array of full paths of all nodes to be
array $predicates	Array of predicates to use.

Return value:

array	The array returned by this method contains a list of all nodes matching the given predicates.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate_step()

Method Details: check_node_test

private boolean check_node_test (string $context, string $node_test)

This method checks whether a node in the document matches a given node-test.

Parameter:

string	$context	Full path of the node, which should be tested for matching the node-test.
string	$node_test	String containing the node-test for the node.

Return value:

boolean	This method returns true if the node matches the node-test, otherwise false.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_child

private array handle_axis_child (array $axis, string $context)

This method handles the XPath child axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array	This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_parent

private array handle_axis_parent (array $axis, string $context)

This method handles the XPath parent axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array	This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_attribute

private array handle_axis_attribute (array $axis, string $context)

This method handles the XPath attribute axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array	This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_self

private array handle_axis_self (array $axis, string $context)

This method handles the XPath self axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array	This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_descendant

private array handle_axis_descendant (array $axis, string $context)

This method handles the XPath descendant axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_ancestor

private array handle_axis_ancestor (array $axis, string $context)

This method handles the XPath ancestor axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_namespace

private array handle_axis_namespace (array $axis, string $context)

This method handles the XPath namespace axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array	This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_following

private array handle_axis_following (array $axis, string $context)

This method handles the XPath following axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array	This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_preceding

private array handle_axis_preceding (array $axis, string $context)

This method handles the XPath preceding axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array	This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_following_sibling

private array handle_axis_following_sibling (array $axis, string $context)

This method handles the XPath following-sibling axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array	This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_preceding_sibling

private array handle_axis_preceding_sibling (array $axis, string $context)

This method handles the XPath preceding-sibling axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array	This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_descendant_or_self

private array handle_axis_descendant_or_self (array $axis, string $context)

This method handles the XPath descendant-or-self axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array	This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_axis_ancestor_or_self

private array handle_axis_ancestor_or_self (array $axis, string $context)

This method handles the XPath ancestor-or-self axis.

Parameter:

array	$axis	Array containing information about the axis.
string	$context	Node from which starting the axis should be

Return value:

array	This method returns an array containing all nodes that were found during the evaluation of the given axis.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_last

private mixed handle_function_last (string $node, string $arguments)

This method handles the XPath function last.

Parameter:

string	$node	Full path of the node on which the function should be processed.
string	$arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_position

private mixed handle_function_position (string $node, string $arguments)

This method handles the XPath function position.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_count

private mixed handle_function_count (string $node, string $arguments)

This method handles the XPath function count.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_id

private mixed handle_function_id (string $node, string $arguments)

This method handles the XPath function id.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_name

private mixed handle_function_name (string $node, string $arguments)

This method handles the XPath function name.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_string

private mixed handle_function_string (string $node, string $arguments)

This method handles the XPath function string.

Parameter:

string $node	Full path of the node on which the function should be processed.

string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_concat

private mixed handle_function_concat (string $node, string $arguments)

This method handles the XPath function concat.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_starts_with

private mixed handle_function_starts_with (string $node, string $arguments)

This method handles the XPath function starts-with.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_contains

private mixed handle_function_contains (string $node, string $arguments)

This method handles the XPath function contains.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_substring_before

private mixed handle_function_substring_before (string $node, string $arguments)

This method handles the XPath function substring-before.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_substring_after

private mixed handle_function_substring_after (string $node, string $arguments)

This method handles the XPath function substring-after.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_substring

private mixed handle_function_substring (string $node, string $arguments)

This method handles the XPath function substring.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_string_length

private mixed handle_function_string_length (string $node, string $arguments)

This method handles the XPath function string-length.

Parameter:

string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_translate

private mixed handle_function_translate (string $node, string $arguments)

This method handles the XPath function translate.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_boolean

private mixed handle_function_boolean (string $node, string $arguments)

This method handles the XPath function boolean.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_not

private mixed handle_function_not (string $node, string $arguments)

This method handles the XPath function not.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_true

private mixed handle_function_true (string $node, string $arguments)

This method handles the XPath function true.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_false

private mixed handle_function_false (string $node, string $arguments)

This method handles the XPath function false.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_lang

private mixed handle_function_lang (string $node, string $arguments)

This method handles the XPath function lang.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_number

private mixed handle_function_number (string $node, string $arguments)

This method handles the XPath function number.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_sum

private mixed handle_function_sum (string $node, string $arguments)

This method handles the XPath function sum.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_floor

private mixed handle_function_floor (string $node, string $arguments)

This method handles the XPath function floor.

Parameter:

string $node	Full path of the node on which the function should be processed.

string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_ceiling

private mixed handle_function_ceiling (string $node, string $arguments)

This method handles the XPath function ceiling.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_round

private mixed handle_function_round (string $node, string $arguments)

This method handles the XPath function round.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: handle_function_text

private mixed handle_function_text (string $node, string $arguments)

This method handles the XPath function text.

Parameter:

string $node	Full path of the node on which the function should be processed.
string $arguments	String containing the arguments that were passed to the function.

Return value:

mixed	Depending on the type of function being processed this method returns different types.

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

evaluate()

Method Details: prestr

private string prestr (string $string, string $delimiter)

This method retrieves everything from a string before a given delimiter, not including the delimiter.

Parameter:

string $string	String, from which the substring should be
string $delimiter	String containing the delimiter to use.

Return value:

string	Substring from the original string before the

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

afterstr()

Method Details: afterstr

private string afterstr (string $string, string $delimiter)

This method retrieves everything from a string after a given delimiter, not including the delimiter.

Parameter:

string $string	String, from which the substring should be
string $delimiter	String containing the delimiter to use.

Return value:

string Substring from the original string after the

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

See also:

prestr()

Method Details: display_error

private void display_error (string $message)

This method displays an error messages and stops the execution of the script. This method is called exactly in the same way as the printf function. The first argument contains the message and additional arguments of various types may be passed to this method to be inserted into the message.

Parameter:

string $message Error message to be

Return value:

void

Author(s):

Michael P. Mehl <**mpm@phpxml.org**>

Field Details: $nodes

private array $nodes

This array contains a list of all document nodes saved as an associative array.

Field Details: $ids

private array $ids

This array contains a list of all IDs of all document nodes that are used for counting when adding a new node.

Field Details: $path

private string $path

This variable saves the current path while parsing a XML file and adding the nodes being read from the file.

Field Details: $position

private integer $position

This variable counts the current document position while parsing a XML file and adding the nodes being read from the file.

Field Details: $root

private string $root

This string contains the full path to the node that acts as the root node of the whole document.

Field Details: $xpath

private string $xpath

This string contains the full XPath expression being parsed currently.

Field Details: $entities

private array $entities

This array contains a list of entities to be converted when an XPath expression is evaluated.

Field Details: $axes

private array $axes

This array contains a list of all valid axes that can be evaluated in an XPath expression.

Field Details: $functions

private array $functions

This array contains a list of all valid functions that can be evaluated in an XPath expression.

Field Details: $operators

private array $operators

This array contains a list of all valid operators that can be evaluated in a predicate of an XPath expression. The list is ordered by the precedence of the operators (lowest precedence first).