| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/LSParserFilter.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/org/w3c/dom/ls/LSParser.html)   [**NEXT CLASS**](http://docs.google.com/org/w3c/dom/ls/LSProgressEvent.html) | [**FRAMES**](http://docs.google.com/index.html?org/w3c/dom/ls/LSParserFilter.html)    [**NO FRAMES**](http://docs.google.com/LSParserFilter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | CONSTR | [METHOD](#2et92p0) | DETAIL: [FIELD](#tyjcwt) | CONSTR | [METHOD](#17dp8vu) |

## **org.w3c.dom.ls**

Interface LSParserFilter

public interface **LSParserFilter**

LSParserFilters provide applications the ability to examine nodes as they are being constructed while parsing. As each node is examined, it may be modified or removed, or the entire parse may be terminated early.

At the time any of the filter methods are called by the parser, the owner Document and DOMImplementation objects exist and are accessible. The document element is never passed to the LSParserFilter methods, i.e. it is not possible to filter out the document element. Document, DocumentType, Notation, Entity, and Attr nodes are never passed to the acceptNode method on the filter. The child nodes of an EntityReference node are passed to the filter if the parameter " [entities](http://www.w3.org/TR/DOM-Level-3-Core/core.html#parameter-entities)" is set to false. Note that, as described by the parameter " [entities](http://www.w3.org/TR/DOM-Level-3-Core/core.html#parameter-entities)", unexpanded entity reference nodes are never discarded and are always passed to the filter.

All validity checking while parsing a document occurs on the source document as it appears on the input stream, not on the DOM document as it is built in memory. With filters, the document in memory may be a subset of the document on the stream, and its validity may have been affected by the filtering.

All default attributes must be present on elements when the elements are passed to the filter methods. All other default content must be passed to the filter methods.

DOM applications must not raise exceptions in a filter. The effect of throwing exceptions from a filter is DOM implementation dependent.

See also the [Document Object Model (DOM) Level 3 Load and Save Specification](http://www.w3.org/TR/2004/REC-DOM-Level-3-LS-20040407).

| **Field Summary** | |
| --- | --- |
| static short | [**FILTER\_ACCEPT**](http://docs.google.com/org/w3c/dom/ls/LSParserFilter.html#FILTER_ACCEPT)            Accept the node. |
| static short | [**FILTER\_INTERRUPT**](http://docs.google.com/org/w3c/dom/ls/LSParserFilter.html#FILTER_INTERRUPT)            Interrupt the normal processing of the document. |
| static short | [**FILTER\_REJECT**](http://docs.google.com/org/w3c/dom/ls/LSParserFilter.html#FILTER_REJECT)            Reject the node and its children. |
| static short | [**FILTER\_SKIP**](http://docs.google.com/org/w3c/dom/ls/LSParserFilter.html#FILTER_SKIP)            Skip this single node. |

| **Method Summary** | |
| --- | --- |
| short | [**acceptNode**](http://docs.google.com/org/w3c/dom/ls/LSParserFilter.html#acceptNode(org.w3c.dom.Node))([Node](http://docs.google.com/org/w3c/dom/Node.html) nodeArg)            This method will be called by the parser at the completion of the parsing of each node. |
| int | [**getWhatToShow**](http://docs.google.com/org/w3c/dom/ls/LSParserFilter.html#getWhatToShow())()            Tells the LSParser what types of nodes to show to the method LSParserFilter.acceptNode. |
| short | [**startElement**](http://docs.google.com/org/w3c/dom/ls/LSParserFilter.html#startElement(org.w3c.dom.Element))([Element](http://docs.google.com/org/w3c/dom/Element.html) elementArg)            The parser will call this method after each Element start tag has been scanned, but before the remainder of the Element is processed. |

| **Field Detail** |
| --- |

### FILTER\_ACCEPT

static final short **FILTER\_ACCEPT**

Accept the node.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#org.w3c.dom.ls.LSParserFilter.FILTER_ACCEPT)

### FILTER\_REJECT

static final short **FILTER\_REJECT**

Reject the node and its children.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#org.w3c.dom.ls.LSParserFilter.FILTER_REJECT)

### FILTER\_SKIP

static final short **FILTER\_SKIP**

Skip this single node. The children of this node will still be considered.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#org.w3c.dom.ls.LSParserFilter.FILTER_SKIP)

### FILTER\_INTERRUPT

static final short **FILTER\_INTERRUPT**

Interrupt the normal processing of the document.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#org.w3c.dom.ls.LSParserFilter.FILTER_INTERRUPT)

| **Method Detail** |
| --- |

### startElement

short **startElement**([Element](http://docs.google.com/org/w3c/dom/Element.html) elementArg)

The parser will call this method after each Element start tag has been scanned, but before the remainder of the Element is processed. The intent is to allow the element, including any children, to be efficiently skipped. Note that only element nodes are passed to the startElement function.

The element node passed to startElement for filtering will include all of the Element's attributes, but none of the children nodes. The Element may not yet be in place in the document being constructed (it may not have a parent node.)

A startElement filter function may access or change the attributes for the Element. Changing Namespace declarations will have no effect on namespace resolution by the parser.

For efficiency, the Element node passed to the filter may not be the same one as is actually placed in the tree if the node is accepted. And the actual node (node object identity) may be reused during the process of reading in and filtering a document.

**Parameters:**elementArg - The newly encountered element. At the time this method is called, the element is incomplete - it will have its attributes, but no children. **Returns:**

* FILTER\_ACCEPT if the Element should be included in the DOM document being built.
* FILTER\_REJECT if the Element and all of its children should be rejected.
* FILTER\_SKIP if the Element should be skipped. All of its children are inserted in place of the skipped Element node.
* FILTER\_INTERRUPT if the filter wants to stop the processing of the document. Interrupting the processing of the document does no longer guarantee that the resulting DOM tree is XML well-formed. The Element is rejected.

Returning any other values will result in unspecified behavior.

### acceptNode

short **acceptNode**([Node](http://docs.google.com/org/w3c/dom/Node.html) nodeArg)

This method will be called by the parser at the completion of the parsing of each node. The node and all of its descendants will exist and be complete. The parent node will also exist, although it may be incomplete, i.e. it may have additional children that have not yet been parsed. Attribute nodes are never passed to this function.

From within this method, the new node may be freely modified - children may be added or removed, text nodes modified, etc. The state of the rest of the document outside this node is not defined, and the affect of any attempt to navigate to, or to modify any other part of the document is undefined.

For validating parsers, the checks are made on the original document, before any modification by the filter. No validity checks are made on any document modifications made by the filter.

If this new node is rejected, the parser might reuse the new node and any of its descendants.

**Parameters:**nodeArg - The newly constructed element. At the time this method is called, the element is complete - it has all of its children (and their children, recursively) and attributes, and is attached as a child to its parent. **Returns:**

* FILTER\_ACCEPT if this Node should be included in the DOM document being built.
* FILTER\_REJECT if the Node and all of its children should be rejected.
* FILTER\_SKIP if the Node should be skipped and the Node should be replaced by all the children of the Node.
* FILTER\_INTERRUPT if the filter wants to stop the processing of the document. Interrupting the processing of the document does no longer guarantee that the resulting DOM tree is XML well-formed. The Node is accepted and will be the last completely parsed node.

### getWhatToShow

int **getWhatToShow**()

Tells the LSParser what types of nodes to show to the method LSParserFilter.acceptNode. If a node is not shown to the filter using this attribute, it is automatically included in the DOM document being built. See NodeFilter for definition of the constants. The constants SHOW\_ATTRIBUTE , SHOW\_DOCUMENT, SHOW\_DOCUMENT\_TYPE, SHOW\_NOTATION, SHOW\_ENTITY, and SHOW\_DOCUMENT\_FRAGMENT are meaningless here. Those nodes will never be passed to LSParserFilter.acceptNode.

The constants used here are defined in [[DOM Level 2 Traversal and Range](http://www.w3.org/TR/2000/REC-DOM-Level-2-Traversal-Range-20001113)] .

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/LSParserFilter.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/org/w3c/dom/ls/LSParser.html)   [**NEXT CLASS**](http://docs.google.com/org/w3c/dom/ls/LSProgressEvent.html) | [**FRAMES**](http://docs.google.com/index.html?org/w3c/dom/ls/LSParserFilter.html)    [**NO FRAMES**](http://docs.google.com/LSParserFilter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | CONSTR | [METHOD](#2et92p0) | DETAIL: [FIELD](#tyjcwt) | CONSTR | [METHOD](#17dp8vu) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).