



REC-DOM-Level-1-19981001

Document Object Model (DOM) Level 1 Specification

Version 1.0

W3C Recommendation 1 October, 1998

This version

<http://www.w3.org/TR/1998/REC-DOM-Level-1-19981001>
<http://www.w3.org/TR/1998/REC-DOM-Level-1-19981001/DOM.ps>
<http://www.w3.org/TR/1998/REC-DOM-Level-1-19981001/DOM.pdf>
<http://www.w3.org/TR/1998/REC-DOM-Level-1-19981001/DOM.tgz>
<http://www.w3.org/TR/1998/REC-DOM-Level-1-19981001/DOM.zip>
<http://www.w3.org/TR/1998/REC-DOM-Level-1-19981001/DOM.txt>

Latest version

<http://www.w3.org/TR/REC-DOM-Level-1>

Previous versions

<http://www.w3.org/TR/1998/PR-DOM-Level-1-19980818>
<http://www.w3.org/TR/1998/WD-DOM-19980720>
<http://www.w3.org/TR/1998/WD-DOM-19980416>
<http://www.w3.org/TR/WD-DOM-19980318>
<http://www.w3.org/TR/WD-DOM-971209>
<http://www.w3.org/TR/WD-DOM-971009>

WG Chair

Lauren Wood, *SoftQuad, Inc.*

Editors

Vidur Apparao, *Netscape*
Steve Byrne, *Sun*
Mike Champion, *ArborText*
Scott Isaacs, *Microsoft*
Ian Jacobs, *W3C*
Arnaud Le Hors, *W3C*
Gavin Nicol, *Inso EPS*
Jonathan Robie, *Texcel Research*
Robert Sutor, *IBM*
Chris Wilson, *Microsoft*
Lauren Wood, *SoftQuad, Inc.*

Principal Contributors

Vidur Apparao, *Netscape*
Steve Byrne, *Sun (until November 1997)*
Mike Champion, *ArborText, Inc.*

Scott Isaacs, *Microsoft (until January, 1998)*
Arnaud Le Hors, *W3C*
Gavin Nicol, *Inso EPS*
Jonathan Robie, *Texcel Research*
Peter Sharpe, *SoftQuad, Inc.*
Bill Smith, *Sun (after November 1997)*
Jared Sorensen, *Novell*
Robert Sutor, *IBM*
Ray Whitmer, *iMall*
Chris Wilson, *Microsoft (after January, 1998)*

Status of this document

This document has been reviewed by W3C Members and other interested parties and has been endorsed by the Director as a W3C Recommendation. It is a stable document and may be used as reference material or cited as a normative reference from another document. W3C's role in making the Recommendation is to draw attention to the specification and to promote its widespread deployment. This enhances the functionality and interoperability of the Web.

The authors of this document are the DOM Working Group members, different chapters may have different editors.

Comments on this document should be sent to the public mailing list www-dom@w3.org.

A list of current W3C Recommendations and other technical documents can be found at <http://www.w3.org/TR>.

Errata

The list of known errors in this document is found at <http://www.w3.org/DOM/updates/REC-DOM-Level-1-19981001-errata.html>.

Available Languages

The English version of this specification is the only normative version. However, for translations in other languages see <http://www.w3.org/DOM/updates/REC-DOM-Level-1-translations.html>.

Abstract

This specification defines the Document Object Model Level 1, a platform- and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure and style of documents. The Document Object Model provides a standard set of objects for representing HTML and XML documents, a standard model of how these objects can be combined, and a standard interface for accessing and manipulating them. Vendors can support the DOM as an interface to their proprietary data structures and APIs, and content authors can write to the standard DOM interfaces rather than product-specific APIs, thus increasing interoperability on the Web.

The goal of the DOM specification is to define a programmatic interface for XML and HTML. The DOM Level 1 specification is separated into two parts: Core and HTML. The Core DOM Level 1 section provides a low-level set of fundamental interfaces that can represent any structured document, as well as defining extended interfaces for representing an XML document. These extended XML interfaces need not be implemented by a DOM implementation that only provides access to HTML documents; all of the fundamental interfaces in the Core section must be implemented. A compliant DOM implementation that implements the extended XML interfaces is required to also implement the fundamental Core interfaces, but not the HTML interfaces. The HTML Level 1 section provides additional, higher-level interfaces that are used with the fundamental interfaces defined in the Core Level 1 section to provide a more convenient view of an HTML document. A compliant implementation of the HTML DOM implements all of the fundamental Core interfaces as well as the HTML interfaces.

Table of contents

● Expanded Table of Contents5
● Copyright Notice7
● What is the Document Object Model?9
● Chapter 1: Document Object Model (Core) Level 1	15
● Chapter 2: Document Object Model (HTML) Level 1	49
● Appendix A: Contributors	95
● Appendix B: Glossary	97
● Appendix C: IDL Definitions	103
● Appendix D: Java Language Binding	117
● Appendix E: ECMA Script Language Binding	135
● References	161
● Index	163
● Production Notes (Non-Normative)	167

Table of contents

Expanded Table of Contents

● Expanded Table of Contents5
● Copyright Notice7
● What is the Document Object Model?9
○ Introduction	10
○ What the Document Object Model is	10
○ What the Document Object Model is not	12
○ Where the Document Object Model came from	12
○ Entities and the DOM Core	12
○ DOM Interfaces and DOM Implementations	13
○ Limitations of Level 1	14
● Chapter 1: Document Object Model (Core) Level 1	15
○ 1.1. Overview of the DOM Core Interfaces	16
● 1.1.1. The DOM Structure Model	16
● 1.1.2. Memory Management	16
● 1.1.3. Naming Conventions	17
● 1.1.4. Inheritance vs Flattened Views of the API	17
● 1.1.5. The DOMString type	18
● 1.1.6. Case sensitivity in the DOM	18
○ 1.2. Fundamental Interfaces	19
○ 1.3. Extended Interfaces	43
● Chapter 2: Document Object Model (HTML) Level 1	49
○ 2.1. Introduction	50
○ 2.2. HTML Application of Core DOM	50
● 2.2.1. Naming Conventions	50
○ 2.3. Miscellaneous Object Definitions	51
○ 2.4. Objects related to HTML documents	52
○ 2.5. HTML Elements	55
● 2.5.1. Property Attributes	55
● 2.5.2. Naming Exceptions	55
● 2.5.3. Exposing Element Type Names (tagName)	56
● 2.5.4. The HTMLInputElement interface	56
● 2.5.5. Object definitions	57
● Appendix A: Contributors	95
● Appendix B: Glossary	97
● Appendix C: IDL Definitions	103
○ C.1. Document Object Model Level 1 Core	103
○ C.2. Document Object Model Level 1 HTML	106
● Appendix D: Java Language Binding	117
○ D.1. Document Object Model Level 1 Core	117

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.