



# Extensible Markup Language (XML) 1.0

W3C Recommendation 10-Feb-98

## This version

<http://www.w3.org/TR/1998/REC-xml-19980210>  
<http://www.w3.org/TR/1998/REC-xml-19980210.xml>  
<http://www.w3.org/TR/1998/REC-xml-19980210.html>  
<http://www.w3.org/TR/1998/REC-xml-19980210.pdf>  
<http://www.w3.org/TR/1998/REC-xml-19980210.ps>

## Latest version

<http://www.w3.org/TR/REC-xml>

## Previous version

<http://www.w3.org/TR/PR-xml-971208>

## Editors

Tim Bray, Textuality and Netscape ([tbray@textuality.com](mailto:tbray@textuality.com))  
Jean Paoli, Microsoft ([jeanpa@microsoft.com](mailto:jeanpa@microsoft.com))  
C. M. Sperberg-McQueen, University of Illinois at Chicago ([cmsmcq@uic.edu](mailto:cmsmcq@uic.edu))

## Abstract

The Extensible Markup Language (XML) is a subset of SGML that is completely described in this document. Its goal is to enable generic SGML to be served, received, and processed on the Web in the way that is now possible with HTML. XML has been designed for ease of implementation and for interoperability with both SGML and HTML.

## 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.

This document specifies a syntax created by subsetting an existing, widely used international text processing standard (Standard Generalized Markup Language, ISO 8879:1986(E) as amended and corrected) for use on the World Wide Web. It is a product of the W3C XML Activity, details of which can be found at <http://www.w3.org/XML>. A list of current W3C Recommendations and other technical documents can be found at <http://www.w3.org/TR>.

This specification uses the term URI, which is defined by [Berners-Lee], a work in progress expected to update [RFC1738] and [RFC1808].

The list of known errors in this specification is available at <http://www.w3.org/XML/xml-19980210-errata>.

Please report errors in this document to [xml-editor@w3.org](mailto:xml-editor@w3.org).

This page intentionally left blank.

# Table of Contents

<b>1. Introduction</b> .....	<b>1</b>
1.1 Origin and Goals.....	1
1.2 Terminology .....	1
<b>2. Documents</b> .....	<b>2</b>
2.1 Well-Formed XML Documents.....	3
2.2 Characters.....	3
2.3 Common Syntactic Constructs.....	3
2.4 Character Data and Markup.....	4
2.5 Comments .....	5
2.6 Processing Instructions.....	5
2.7 CDATA Sections .....	5
2.8 Prolog and Document Type Declaration .....	5
2.9 Standalone Document Declaration.....	7
2.10 White Space Handling.....	8
2.11 End-of-Line Handling .....	8
2.12 Language Identification.....	8
<b>3. Logical Structures</b> .....	<b>10</b>
3.1 Start-Tags, End-Tags, and Empty-Element Tags.....	10
3.2 Element Type Declarations .....	11
3.3 Attribute-List Declarations.....	13
3.4 Conditional Sections .....	15
<b>4. Physical Structures</b> .....	<b>16</b>
4.1 Character and Entity References.....	16
4.2 Entity Declarations .....	17
4.3 Parsed Entities .....	19
4.4 XML Processor Treatment of Entities and References.....	20
4.5 Construction of Internal Entity Replacement Text.....	22
4.6 Predefined Entities.....	23
4.7 Notation Declarations .....	23
4.8 Document Entity.....	23
<b>5. Conformance</b> .....	<b>24</b>
5.1 Validating and Non-Validating Processors.....	24
5.2 Using XML Processors .....	24
<b>6. Notation</b> .....	<b>24</b>

## Appendices

<b>A. References</b> .....	<b>26</b>
A.1 Normative References .....	26
A.2 Other References.....	26
<b>B. Character Classes</b> .....	<b>27</b>
<b>C. XML and SGML (Non-Normative)</b> .....	<b>29</b>
<b>D. Expansion of Entity and Character References (Non-Normative)</b> .....	<b>29</b>
<b>E. Deterministic Content Models (Non-Normative)</b> .....	<b>30</b>
<b>F. Autodetection of Character Encodings (Non-Normative)</b> .....	<b>30</b>
<b>G. W3C XML Working Group (Non-Normative)</b> .....	<b>32</b>

This page intentionally left blank.

# 1. Introduction

Extensible Markup Language, abbreviated XML, describes a class of data objects called XML documents and partially describes the behavior of computer programs which process them. XML is an application profile or restricted form of SGML, the Standard Generalized Markup Language [ISO8879]. By construction, XML documents are conforming SGML documents.

XML documents are made up of storage units called entities, which contain either parsed or unparsed data. Parsed data is made up of characters, some of which form character data, and some of which form markup. Markup encodes a description of the document's storage layout and logical structure. XML provides a mechanism to impose constraints on the storage layout and logical structure.

A software module called an *XML processor* is used to read XML documents and provide access to their content and structure. It is assumed that an XML processor is doing its work on behalf of another module, called the *application*. This specification describes the required behavior of an XML processor in terms of how it must read XML data and the information it must provide to the application.

## 1.1 Origin and Goals

XML was developed by an XML Working Group (originally known as the SGML Editorial Review Board) formed under the auspices of the World Wide Web Consortium (W3C) in 1996. It was chaired by Jon Bosak of Sun Microsystems with the active participation of an XML Special Interest Group (previously known as the SGML Working Group) also organized by the W3C. The membership of the XML Working Group is given in an appendix. Dan Connolly served as the WG's contact with the W3C.

The design goals for XML are:

1. XML shall be straightforwardly usable over the Internet.
2. XML shall support a wide variety of applications.
3. XML shall be compatible with SGML.
4. It shall be easy to write programs which process XML documents.
5. The number of optional features in XML is to be kept to the absolute minimum, ideally zero.
6. XML documents should be human-legible and reasonably clear.
7. The XML design should be prepared quickly.
8. The design of XML shall be formal and concise.
9. XML documents shall be easy to create.
10. Terseness in XML markup is of minimal importance.

This specification, together with associated standards (Unicode and ISO/IEC 10646 for characters, Internet RFC 1766 for language identification tags, ISO 639 for language name codes, and ISO 3166 for country name codes), provides all the information necessary to understand XML Version 1.0 and construct computer programs to process it.

This version of the XML specification may be distributed freely, as long as all text and legal notices remain intact.

## 1.2 Terminology

The terminology used to describe XML documents is defined in the body of this specification. The terms defined in the following list are used in building those definitions and in describing the actions of an XML processor:

**may**

Conforming documents and XML processors are permitted to but need not behave as described.

# 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.