Atitit 标记语言ML(Markup Language)

目录

[1. 标记语言ML Markup Language 1](#_Toc29335)

[1.1. 简介 1](#_Toc12597)

[2. 置标语言置标语言通常可以分为三类：标识性的、过程性的以及描述性的。 2](#_Toc7165)

[2.1.1. 折叠标识性的置标语言（类似Markdown  2](#_Toc9201)

[2.1.2. 折叠过程性标识（html ？？） 2](#_Toc16032)

[2.1.3. 折叠描述性标识 2](#_Toc29854)

[3. 常用标记语言 3](#_Toc10094)

[4. 应用 3](#_Toc7123)

[5. 语言历史 3](#_Toc13974)

[6. Ml代数 5](#_Toc25383)

[6.1. 5第三代置标语言HTML 5](#_Toc30930)

[6.2. 6建立在XML之上的应用（第四代置标语言）rdf svg 5](#_Toc28652)

# 标记语言ML Markup Language

标记语言，是一种将[文本](https://baike.so.com/doc/6023706-6236703.html" \t "https://baike.so.com/doc/_blank)以及文本相关的其他信息结合起来，展现出关于文档结构和[数据处理](https://baike.so.com/doc/6144874-6358051.html" \t "https://baike.so.com/doc/_blank)细节的电脑文字编码。与[文本](https://baike.so.com/doc/6023706-6236703.html" \t "https://baike.so.com/doc/_blank)相关的其他信息(包括文本的结构和表示信息等)与原来的文本结合在一起，但是使用[标记](https://baike.so.com/doc/5997615-6210588.html" \t "https://baike.so.com/doc/_blank)进行标识。

标记语言不仅仅是一种语言。就像许多语言一样，它需要一个运行时环境，使其有用。提供运行时环境的元素称为用户代理

## 简介

标记语言，也称置标语言，是一种将文本（Text）以及文本相关的其他信息结合起来，展现出关于文档结构和数据处理细节的电脑文字编码。与文本相关的其他信息（包括例如文本的结构和表示信息等）与原来的文本结合在一起，但是使用标记（markup）进行标识。当今广泛使用的置标语言是超文本置标语言（HyperText Markup Language，HTML）和可扩展置标语言 （eXtensible Markup Language，XML）。置标语言广泛应用于网页和网络[应用程序](https://baike.so.com/doc/3417785-3597266.html" \t "https://baike.so.com/doc/_blank)。标记最早用于出版业，是作者、编辑以及出版商之间用于描述出版作品的排版格式所使用的。

标记语言（markup language），用一系列约定好的标记来对电子文档进行标记，以实现对电子文档的语义、结构、及格式的定义。这些标记必须很容易的和内容区分，并且易于识别

# 置标语言置标语言通常可以分为三类：标识性的、过程性的以及描述性的。

### [折叠](https://baike.so.com/doc/6932363-7154686.html)标识性的置标语言（类似**Markdown**

标识性的置标语言 （Presentational markup）是在编码过程中，标记文档的结构信息。例如，在文本文件中, 文件的标题可能需要用特定的格式表示（居中，放大等），这样我们就需要标记文件的标题。字处理以及桌面出版产品有时候能够自动推断出这类的结构信息，但是绝大多数的，像Wiki这样的纯文本[编辑器](https://baike.so.com/doc/6781941-6998356.html" \t "https://baike.so.com/doc/_blank)还不能解决这个问题。

### [折叠](https://baike.so.com/doc/6932363-7154686.html)过程性标识（html ？？）

过程性置标语言（Procedural markup） 一般都专门于文字的表达，但通常对于文本编辑者可见,并且能够被软件依其出现顺序依次解读 。 为了格式化一个标题，在标题文本之前，会紧接着插入一系列的格式标识符，以指示计算机转换到居中的显示模式，同时加大及加粗字体。在标题文本之后，会紧接缀上格式中止标识; 对于更高级的系统[宏命令](https://baike.so.com/doc/6520373-6734103.html" \t "https://baike.so.com/doc/_blank)或者堆栈模式会让这一过程的实现方式更加丰富 。大多数情况下， 过程性置标能力包含有一个Turing-complete编程语言。 过程性置标语言的范例有：nroff, troff, TeX, Lout 以及 PostScript. 过程性置标语言被广泛应用在专业出版领域, 专业的出版商会根据要求使用不同的指标语言已达到出版要求.

### [折叠](https://baike.so.com/doc/6932363-7154686.html)描述性标识

历史

“置标（markup）”这个词来源自传统出版业的“标记”一个手稿，也就是在原稿的边缘加注一些符号来指示打印上的要求。 长久以来，这个工作都是由专门的人（"markup men" ）以及校对人来进行，对原稿标志出使用什么样的字体，字型以及字号，然后再将原稿交给其他人进行手工的排版工作。

# 常用标记语言 jstl xml h5

JSTL（JavaServer Pages Standard Tag Library，JSP标准标签库

# 应用

XAML（Extensible Application Markup Language），基于XML语言，在微软WPF（Windows Presentation Foundation）中使用。

# 语言历史

“置标”或“标记”一词来源于传统的出版业，是作者、编辑以及出版商之间用于描述出版作品的排版格式所使用的。起初一位作者如果想把他的作品拿到出版社出版，他不得不在正文的边缘地带加注一些符号来表示打印上的需求。长引以往，出版社就有了专人专门识别这些加注符号，当时称这些人为“Markup men”，由他们识别出来后按要求打印出版。我们知道当一项服务变得越来越大众化以后，人们就会对它提出更多更高的要求，出版也不例外，于是就有了一些人设计出一些人工语言专门用于置标，我们称这种语言为置标语言。

置标语言的概念由William W. Tunnicliffe在1975年的美国出版业高级行政主管会议上首次提出，尽管当时他宁愿称它为“通用码”。1970年，在Tunnicliffe的领导下为出版业开发出通用码标准“GenCode”，然而在那后不久国标标准的委员会开发出第一个广泛使用的描述性置标语言SGML。

标记语言（markup language），用一系列约定好的标记来对电子文档进行标记，以实现对电子文档的语义、结构、及格式的定义。这些标记必须很容易的和内容区分，并且易于识别。标记语言的发展如下：

GML(1969)

|

SGML(1985)

|

XML(1998) 、、、、、、、HTML(1993)

|——————|——|———|——|

MathML、WML、SVG、CML、XHTML

1,为了促进数据交换和操作，在20世纪60年代，通过IBM格公司研究人员的杰出工作，得出了重要的结论：要提高系统的移植性，必须采用一种通用的文档格式，这种文档的格式必须遵守特定的规则。这也就是创建GML （Generalized Markup Language,通用标记语言）的指导原则，从人们所产生的将文件结构化为标准的格式的动机出发，IBM创建了GML。

2，在标记语言的概念达成共识的基础上，IBM公司的研究人员Charles Goldfarb带领的开发团队完善着GML，将其称为SGML(Standard Generalized Markup Language,标准通用标记语言)，SGML成为了IBM内部格式化和维护合法化文件的手段。后来被拓展和修改，作为一种全面的信息标准以适应工业范围的广泛应用，1986年，SGML被国际标准化组织（ISO）所采纳。

他的功能非常强大，但是非常复杂，需要许多昂贵的软件配合运行，因此在很长一段时间内没有被推广。

3，1989年，欧洲粒子物理实验室（CERT）的研究员Tim Berners-Lee和Anders Berglund共同创建了一种基于标记的语言HTML,他可看做SGML的简单应用，开始时仅仅提供一种对静态文本的信息显示的方法，后来越来越多的标签产生，两大浏览器厂商微软和网景格式，甚至创建了自己的产品的兼容标签，使HTML变得臃肿不堪，兼容性不好。

# Ml代数

* [第一代置标语言GML](http://www.zwbk.org/MyLemmaShow.aspx?lid=70678" \l "2)
* 3[第二代置标语言SGML](http://www.zwbk.org/MyLemmaShow.aspx?lid=70678" \l "3)
* 4[第三代置标语言XML](http://www.zwbk.org/MyLemmaShow.aspx?lid=70678" \l "4)
  + 4.1[XML产生的原因](http://www.zwbk.org/MyLemmaShow.aspx?lid=70678" \l "5)
  + 4.2[XML的使用形式](http://www.zwbk.org/MyLemmaShow.aspx?lid=70678" \l "6)

## 5[第三代置标语言HTML](http://www.zwbk.org/MyLemmaShow.aspx?lid=70678" \l "7)

## 6[建立在XML之上的应用（第四代置标语言）](http://www.zwbk.org/MyLemmaShow.aspx?lid=70678" \l "8)rdf svg

　（1）RDF(Resource Description Framework)，是万维网联盟（W3C）提出的一组标记语言的技术标准, 以便对网络资源的内容与结构作更为丰富的描述和表达。

　　（2）XForms，是HTML form的继承者。它是W3C的标准。它采用XML的格式，易于使用。

　　（3）DocBook是当前风行于开放源码世界的一种文档撰写格式，已经成为计算机文档撰写的事实上的规范。DocBook是基于SGML/XML的、面向结构的文档撰写模式，打破了传统的、面向表现的、所见即所得文档撰写模式。Docbook是一种“原始数据”型的格式，完全独立于任何处理软件，可以通过XSLT，或者类似技术自动生成HTML，PDF，WordML（Word 2003的XML格式），RTF，JavaHelp, HtmlHelp(CHM)等等一系列不同格式，相同内容的文档。

　　（4）简单对象访问协议 (SOAP，全写为Simple Object Access Protocol) 是一种标准化的通讯规范，主要用于Web服务(web service)中。SOAP的出现是为了简化网页服务器(Web Server)在从XML数据库中提取资料时，无需花时间去格式化页面，并能够让不同应用程式之间透过HTTP通讯协定，以XML格式互相交换彼此的资料，使其与编程语言、平台和硬件无关。此标准由IBM、Microsoft、UserLand和DevelopMentor在1998年共同提出，并得到IBM，莲花（Lotus），康柏（Compaq）等公司的支持，于2000年提交给万维网联盟（World Wide Web Consortium；W3C），目前 SOAP 1.1 版是业界共同的标准，属于第二代的XML协定（第一代具主要代表性的技术为XML-RPC以及WDDX）。

　　（5）Wireless Markup Language，缩写为WML，是WAP规范指定的基于XML的基本内容格式，使用支持该规范的设备例如移动电话可以浏览WML的页面。

　　（6）可缩放矢量图形（Scalable Vector Graphics，SVG）是基于可扩展标记语言（XML），用于描述二维矢量图形的一种图形格式。SVG由W3C制定，是一个开放标准。

　　（7）可扩展超文本置标语言（eXtensible HyperText Markup Language，XHTML），是一种置标语言，表现方式与超文本置标语言（HTML）类似，不过语法上更加严格。从继承关系上讲，HTML是一种基于标准通用置标语言（SGML）的应用，是一重非常灵活的置标语言，而XHTML则基于可扩展置标语言（XML），XML是SGML的一个子集。XHTML 1.0在2000年1月26日成为W3C的推荐标准。

　　（8）Atom是一对彼此相关的标准。Atom供稿格式（Atom Syndication Format）是用于网站消息来源，基于XML的文档格式；而Atom出版协定（Atom Publishing Protocol，简称AtomPub或APP）是用于新增及修改网络资源，基于HTTP的协议。