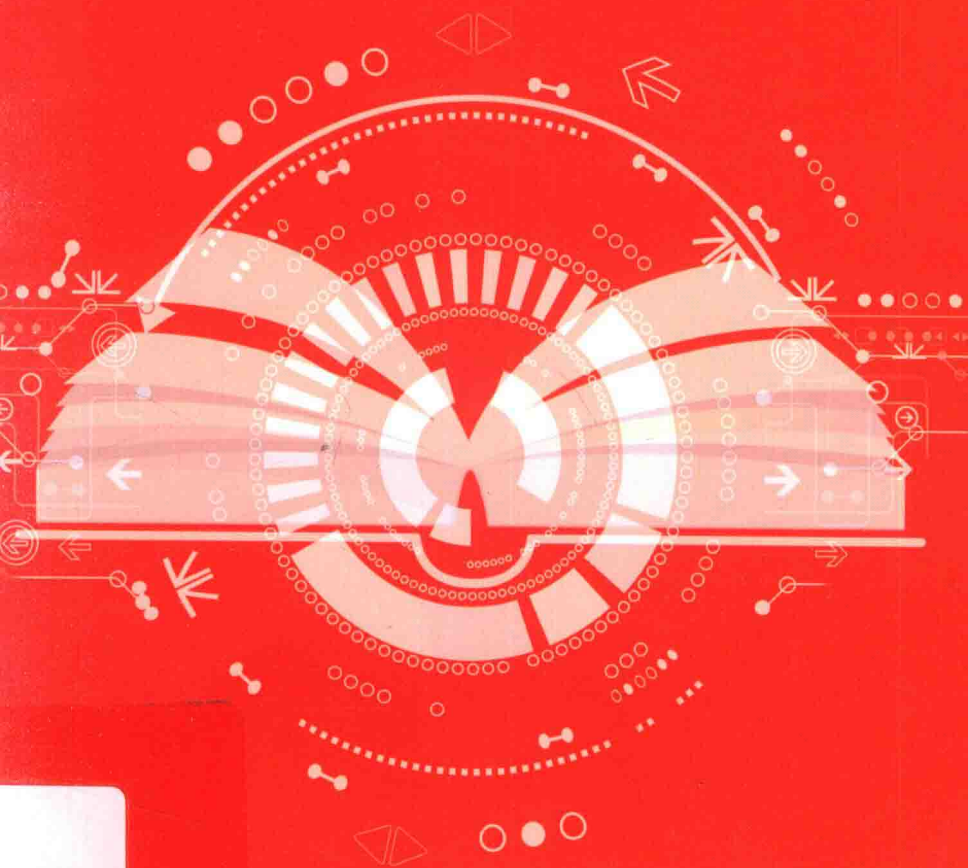


网络信息检索小百科

吴晓明 著



光明日报出版社

网络信息检索小百科

吴晓明 著

光明日报出版社

图书在版编目(CIP)数据

网络信息检索小百科 / 吴晓明著. —北京: 光明日报出版社, 2016.8

ISBN 978-7-5194-1379-8

I. ①网… II. ①吴… III. ①网络检索 IV. ①G354.4

中国版本图书馆 CIP 数据核字(2016)第 170266 号

网络信息检索小百科

著 者: 吴晓明

责任编辑: 李 娟 封面设计: 久仰文化

责任校对: 邓 贝 责任印制: 曹 净

出版发行: 光明日报出版社

地 址: 北京市东城区珠市口东大街 5 号, 100062

电 话: 010-6702219(咨询), 67078870(发行), 67078235(邮购)

传 真: 010-67078227, 67078255

网 址: <http://book.gmw.cn>

E-mail: gmcbs@gmw.cn, cnlijuan@gmw.cn

法律顾问: 北京德恒律师事务所龚柳方律师

印 刷: 三河市明华印务有限公司

装 订: 三河市明华印务有限公司

本书如有破损、缺页、装订错误, 请与本社联系调换

开 本: 880 × 1230 1/32

字 数: 100 千字 印 张: 7.5

版 次: 2017 年 5 月第 1 版 印 次: 2017 年 5 月第 1 次印刷

书 号: ISBN 978-7-5194-1379-8

定 价: 40.00 元

版权所有 翻印必究

前 言

随着互联网的产生，信息的传播速度有了质的飞跃，随后全球一体化的进程加速，信息数量的增速也不断加剧，在如今的信息时代，互联网已经充斥了各种各样的信息，如何才能从信息的海洋中找到自己需要的？信息检索就像一条小船，掌握了信息检索的技巧就能在信息的海洋里乘风破浪，懂得利用网络信息检索对学习与工作都有着重要的影响。

为了获取知识和信息，古人崇尚“皓首穷经”，赞赏“博闻强记”。而如今，当信息急剧增长，出现如钱学森在《现代科学技术的发展》一文中所说的景况，“现在光浏览一下世界上一年内发表的有关化学的论文和著作，一个化学家如果他每周看40个小时，也要读48年”。个人的阅读能力与信息总量的反差，已成几何级数的增长。因此，如何有效地获取与选择信息，已成为人们学习、工作和生活的一项基本功。人们必须掌握信息检索知识。

本书论述了信息检索的有关概念与基本知识，以及Internet重要的中外文搜索引擎与检索技巧，介绍了网上各类型电子信息 and 重要专题信息的查询技巧，阐述了网上电子图书，电子期刊，电子报纸与其他传媒的内容、特点及其使用方法，对网络信息的选择、评价，开发与利用等问题与对策。

与其他网络信息检索方面的书相比，该书兼备理论性与可操作性，在向读者提供信息检索基本知识与技术的基础上，注重实践经验的总结，并精选出几百个网址。本书不仅可作为各高校信息检索课的教材，而且可以指导广大网络用户的网络查询实践，还可供各类型图书馆，情报机构的工作人员学习工作时参考。

目 录

第一章 网页的发展史.....	1
第一节 原始的网页	1
第二节 基于表格的网页时代	3
第三节 JavaScript 的到来	5
第四节 基于 flash 的网页设计	10
第五节 CSS 的黄金应用	13
第六节 响应式网页	18
第七节 纵观 web1.0~web6.0	21
第二章 网络数据库的使用方法.....	24
第一节 运用百度百科了解未知信息	24
第二节 通过新浪微盘搜索文献资料	39
第三节 利用网易公开课学习各名校课程	42
第四节 各类影音娱乐网站搜索资源简介	48

第三章 信息教育和文献检索	63
第一节 信息社会与信息素质教育	63
第二节 文献检索基础知识	80
第三节 文献信息检索系统	88
第四节 检索效果评价	96
第四章 文献信息检索技术	101
第一节 文献信息检索方式	101
第二节 检索途径和检索语言	104
第三节 计算机信息检索技术	111
第四节 检索步骤	120
第五章 文献传递系统	122
第一节 中国高等教育文献保障系统	122
第二节 开世览文	123
第三节 国家科技图书文献中心	126
第四节 读秀学术搜索: 区域数字图书馆搜索平台 ...	129
第六章 电子图书检索	132
第一节 电子图书基础知识	132
第二节 中文电子图书	156
第三节 国外电子图书	162
第四节 网上免费电子图书	165

第七章 中外综合性检索平台	169
第一节 中国知网	169
第二节 万方数据知识服务平台	174
第三节 国道专题数据库超市系统	181
第四节 DIALOG 系统	183
第五节 STN 系统	185
第六节 ISI Web of Knowledge	186
第七节 工程索引	195
第八节 化学文摘	196
第九节 生物学文摘	200
第十节 INSPEC	201
第十一节 OCLC FirstSearch	204

第八章 网络搜索引擎	216
第一节 搜索引擎的检索	216
第二节 学术搜索引擎	219
第三节 其他搜索引擎	225

第一章 网页的发展史

第一节 原始的网页

自从第一个网站在上个世纪90年代初诞生以来，设计师们尝试了各种网页的视觉效果。早期的网页完全由文本构成，除了一些小图片和毫无布局可言的标题与段落。时代在不断进步，接下来出现了表格布局，然后是 Flash，最后是基于 CSS 的网页设计。

在互联网真正开始之时，黑色的显示屏仅能显示单色的像素。可以说，当互联网天地初开之时，Web Design 仅仅意味着字符和空格的排列组合。虽然图形化的界面早在80年代初就有了，但在此时普及率并不高。直到90年代，图形化界面才真正进入千家万户，而那时候，才是属于互联网的狂野西部。

如今，各类精致绚丽的网站泛滥成灾，即使不懂技术也能做



出一个漂亮的网站。然而在25年前，世界上第一个网页的诞生确实极其不易，且页面的设计亦是十分粗糙。好奇第一个网页长啥样吗？

据悉，最早的网络构想可追溯到1980年。那时，英国计算机科学家蒂姆·伯纳斯·李构建了一个名为“ENQUIRE”、类似维基百科的项目，并在1990年11月12日提出了一个更正式的“万维网”建议，且在同年11月13日在一台Next工作站上写出了第一个网页。

这个网页上线于当地时间1990年12月20日。但直到1991年8月6日，蒂姆·伯纳斯·李才通过alt.hypertext新闻组正式公开了这个网页。这是第一个简单的，基于文本，包含几个链接的网站。原始网页的副本现在仍然在线。它有十多个链接，试图告诉人们什么是万维网。

随后的网页都比较相似，完全基于文本，单栏设计，有一些链接等等。最初版本的HTML只有最基本的内容结构：标题（<h1>， <h2>...），段落（<p>）和链接（<a>）。随后新版本的HTML开始允许在页面上添加图片（），然后开始支持制作表格（<table>）。

1994年，万维网联盟（W3C）成立，他们将HTML确立为网页的标准标记语言。这一举动阻断了任何独立公司想要开发专利的浏览器和相应的程序语言的野心，因为这会对网络的完整性产生不利的影响。W3C一直致力于确立与维护网页编程语言的标准

(例如 JavaScript)。

第二节 基于表格的网页时代

能够显示图片的浏览器的诞生，是促使网页设计这个行业诞生的重要先决条件。实际上在当时，最接近于信息结构化的概念，是HTML中已有的元素：表格（Table）。所以，David Siegel 在他的网页设计书《Creating Killer Sites》讲述了他设计优秀网站的秘诀：在表格中嵌套表格，将静态的表格和动态的表格以巧妙的方式结合到一起。尽管表格本身是用来承载数据的，用来承载内容和图片有点奇怪，但是在那个时代，这种方法依然显得颇为靠谱，并且大行其道。

网页设计所面临的另外一个问题，就是如何保持网页那脆弱的结构。也正是因为这种需求，切片设计（Slicing Design）逐渐流行了起来。设计师创建出漂亮的网页布局，随后开发者将整个设计稿切片，找出呈现设计的最佳方法。另一方面，表格还有一些炫酷的功能，比如垂直对齐，以像素为单位或者以百分比来控制对齐。在那个时代，表格可是近乎栅格系统一般的灵活的设计神器，也正是因此，那个时代的开发者并不喜欢前端的代码。



表格布局使网页设计师制作网站时有了更多选择。在HTML中表格标签的本意是为了显示表格化的数据，但是设计师很快意识到可以利用表格来构造他们设计的网页，这样就可以制作较以往作品更加复杂的，多栏目的网页。

表格布局就这样流行了起来，融合了背景图片切片技术，常给人以看起来较实际布局简洁得多的结构。

这个时期的网页设计还不太关注语义化和可用性方面的问题，主要还在追求良好的结构美学。

同一时期也是大量应用GIF占位图片控制留白的时期。一些主流的公司甚至训练设计师如何使用GIF占位；如微软的“关于HTML表格中的GIF占位”。第一批主要应用表格布局的“所见即所得”网页设计软件的发展助长了表格的应用。另外，某些软件自动生成的表格如此复杂以至于许多设计师不可能从头手写代码（例如每行只有1px高却包含了几百列的表格）。即使是稍微复杂一点儿的网页（比如多栏目页面），设计师们都要依赖于表格来创建。

第三节 JavaScript 的到来

JavaScript是一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。它的解释器被称为JavaScript引擎，为浏览器的一部分，广泛用于客户端的脚本语言，最早是在HTML（标准通用标记语言下的一个应用）网页上使用，用来给HTML网页增加动态功能。

在1995年时，由Netscape公司的Brendan Eich，在网景导航者浏览器上首次设计实现而成。因为Netscape与Sun合作，Netscape管理层希望它外观看起来像Java，因此取名为JavaScript。但实际上它的语法风格与Self及Scheme较为接近。

为了取得技术优势，微软推出了JScript，CEnvi推出ScriptEase，与JavaScript同样可在浏览器上运行。为了统一规格，因为JavaScript兼容于ECMA标准，因此也称为ECMAScript。

JavaScript的出现补足了尚且原始的HTML。举个例子，如果你想写个弹出窗，或者想动态修改某些对象的顺序？HTML不行，但是JS可以！不过此时JS的主要问题在于，它处于整个网页布局的顶层并且需要单独加载。很多时候它仅仅被懒惰的开发者用作



一个简单的补丁，但如果使用得当，JS可以非常强大。今天，同样的功能如果CSS能实现，我们会尽量避免使用JS。不可否认的是，JS本身确实很强大，前端常用的jQuery，后端的Node.js都是不可多得的好东西。

一、基本特点

JavaScript是一种属于网络的脚本语言，已经被广泛用于Web应用开发，常用来为网页添加各式各样的动态功能，为用户提供更流畅美观的浏览效果。通常JavaScript脚本是通过嵌入在HTML中来实现自身的功能的。

（一）是一种解释性脚本语言（代码不进行预编译）。

（二）主要用来向HTML（标准通用标记语言下的一个应用）页面添加交互行为。

（三）可以直接嵌入HTML页面，但写成单独的js文件有利于结构和行为的分离。

（四）跨平台特性，在绝大多数浏览器的支持下，可以在多种平台下运行（如Windows、Linux、Mac、Android、iOS等）。

Javascript脚本语言同其他语言一样，有它自身的基本数据类型，表达式和算术运算符及程序的基本程序框架。Javascript提供了四种基本的数据类型和两种特殊数据类型用来处理数据和文字。而变量提供存放信息的地方，表达式则可以完成较复杂的信息处理。

二、日常用途

- (一) 嵌入动态文本于HTML页面。
- (二) 对浏览器事件做出响应。
- (三) 读写HTML元素。
- (四) 在数据被提交到服务器之前验证数据。
- (五) 检测访客的浏览器信息。
- (六) 控制cookies, 包括创建和修改等。
- (七) 基于Node.js技术进行服务器端编程。

三、历史

它最初由Netscape的Brendan Eich设计。JavaScript是甲骨文公司的注册商标。Ecma国际以JavaScript为基础制定了ECMAScript标准。JavaScript也可以用于其他场合, 如服务器端编程。完整的JavaScript实现包含三个部分: ECMAScript, 文档对象模型, 浏览器对象模型。

Netscape在最初将其脚本语言命名为LiveScript, 后来Netscape在与Sun合作之后将其改名为JavaScript。JavaScript最初是受Java启发而开始设计的, 目的之一就是“看上去像Java”, 因此语法上有类似之处, 一些名称和命名规范也借自Java。但JavaScript的主要设计原则源自Self和Scheme。JavaScript与Java名称上的近似, 是当时Netscape为了营销考虑与Sun微系统达成协议的结果。为了取得技术优势, 微软推出了JScript来迎战JavaScript的脚本语言。为了互用性, Ecma国际(前身为欧洲



计算机制造商协会)创建了ECMA-262标准(ECMAScript)。两者都属于ECMAScript的实现。尽管JavaScript作为给非程序人员的脚本语言,而非给程序人员的脚本语言来推广和宣传,但是JavaScript具有非常丰富的特性。

发展初期,JavaScript的标准并未确定,同期有Netscape的JavaScript,微软的JScript和CEnvi的ScriptEase三足鼎立。1997年,在ECMA(欧洲计算机制造商协会)的协调下,由Netscape、Sun、微软、Borland组成的工作组确定统一标准:ECMA-262。

四、特性

JavaScript脚本语言具有以下特点:

(1) 脚本语言。JavaScript是一种解释型的脚本语言,C、C++等语言先编译后执行,而JavaScript是在程序的运行过程中逐行进行解释。

(2) 基于对象。JavaScript是一种基于对象的脚本语言,它不仅可以创建对象,也能使用现有的对象。

(3) 简单。JavaScript语言中采用的是弱类型的变量类型,对使用的数据类型未做出严格的要求,是基于Java基本语句和控制的脚本语言,其设计简单紧凑。

(4) 动态性。JavaScript是一种采用事件驱动的脚本语言,它不需要经过Web服务器就可以对用户的输入做出响应。在访问一个网页时,鼠标在网页中进行鼠标点击或上下移、窗口移

动等操作，JavaScript都可直接对这些事件给出相应的响应。

(5) 跨平台性。JavaScript脚本语言不依赖于操作系统，仅需要浏览器的支持。因此一个JavaScript脚本在编写后可以带到任意机器上使用，前提是机器上的浏览器支持JavaScript脚本语言，目前JavaScript已被大多数的浏览器所支持。

不同于服务器端脚本语言，例如PHP与ASP，JavaScript主要被作为客户端脚本语言在用户的浏览器上运行，不需要服务器的支持。所以在早期程序员比较青睐于JavaScript以减少对服务器的负担，而与此同时也带来另一个问题：安全性。

而随着服务器的强壮，虽然程序员更喜欢运行于服务端的脚本以保证安全，但JavaScript仍然以其跨平台、容易上手等优势大行其道。同时，有些特殊功能（如AJAX）必须依赖JavaScript在客户端进行支持。随着引擎如V8和框架如Node.js的发展，事件驱动及异步IO等特性，JavaScript逐渐被用来编写服务器端程序。