



南開大學  
Nankai University

计算机学院和网络空间安全学院  
软件工程实验报告

基于 Axure 的界面原型设计

姓名：魏伯繁

学号：2011395

专业：信息安全

2023 年 4 月 23 日

# 目录

<b>1 案例分析</b>	<b>2</b>
1.1 划分区域	2
1.2 分解元素	3
1.2.1 页头部分	3
1.2.2 核心搜索区域	3
1.2.3 热搜区域	4
1.2.4 推荐文章区域	4
1.2.5 版权标识区域	5
1.3 识别交互	6
1.3.1 鼠标悬停交互	6
1.3.2 下拉提示框展示	6
1.4 构建元素来源	7
<b>2 案例制作</b>	<b>7</b>
2.1 界面设计	7
2.2 交互设计	9
2.2.1 搜索提示框	9
2.2.2 搜索外框变色	9
2.2.3 带图片的提示框展开	10
2.2.4 移入变色	10
2.3 总结	10
<b>3 第一层标题</b>	<b>10</b>
3.1 第二层标题	10
3.1.1 第三层标题	11
<b>4 代码</b>	<b>11</b>
4.1 伪代码	11
4.2 代码	11
<b>5 图和表格</b>	<b>12</b>
5.1 图	12
<b>6 其他会用到</b>	<b>13</b>
6.1 枚举	13
6.2 公式	13
6.3 链接	13

# 1 案例分析

在绘制自己的网络原型图之前，我们先对百度的原型页面进行分析，对其网页区域进行划分，分解每个区域的不同元素，识别同一区域不同元素的交互，或不同区域之间的交互，并且尝试对其中某些代表性的元素进行构建尝试。

## 1.1 划分区域

打开百度的官方网址，可以将页面分解为以下五个区域，分解的标准主要依基于所实现的功能不同以及在页面中所处的位置不同。

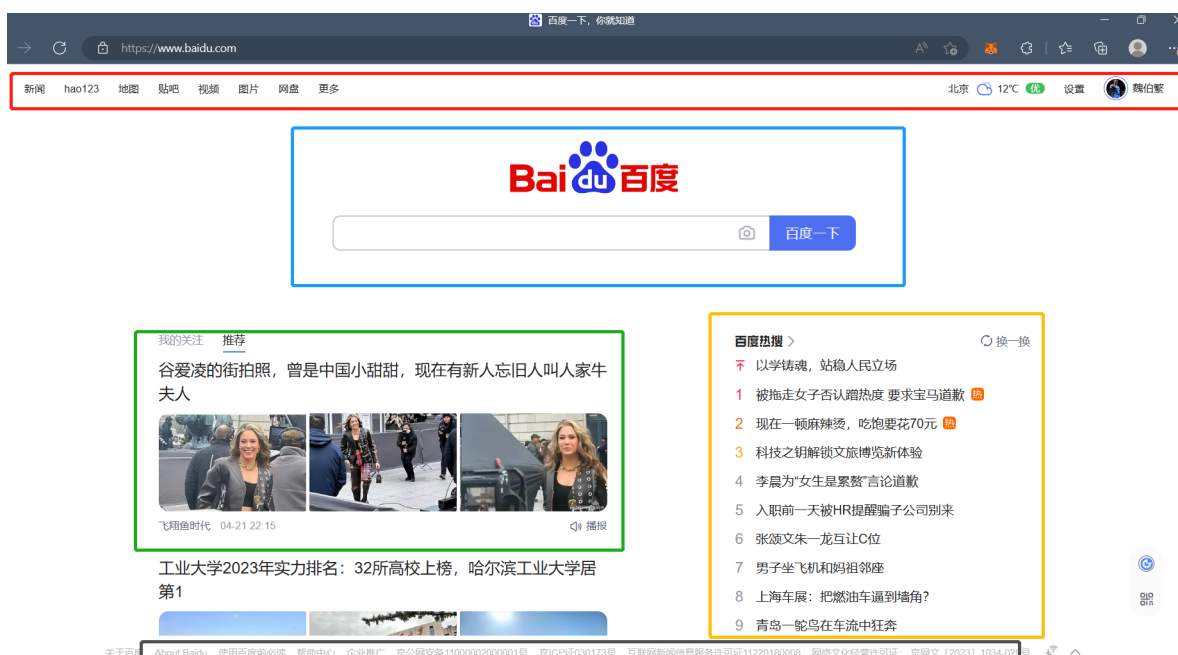


图 1.1: 百度搜索分解图

百度搜索界面的页面布局可以分为以下几个部分：

1. 页头部分，主要包括百度旗下一些子功能的跳转按钮以及个人账户的相关操作
2. 核心搜索区域：核心搜索区域包括百度的标志性 LOGO，放置在页面的正中间，方便用户识别和定位。核心搜索区域还包括搜索框以及搜索按钮，搜索框通常是用户进行搜索操作的入口，百度搜索框放置在页面中央位置，醒目且易于用户发现和使用。
3. 热搜榜：热门搜索展示在页面的右侧，用户可以快速了解当前最热门的搜索关键词，并在需要的时候点击进入相应的搜索结果页面。
4. 推荐内容：推荐的内容展示在页面左下方，用户可以更为便捷的了解相关信息，并执行相关操作。
5. 版权标志，在搜索页面的最下端包括一些百度网站的版权标志，包括 ICP 号、网络经营许可证等信息，保证网站的可靠性和合法性。

整个百度搜索界面的布局经过反复优化与设计，最终采用了这种布局的原因如下：

1. 显著的 LOGO 和搜索框：百度的 LOGO 和搜索框是整个页面最重要也是最核心的部分，应该放置在较为显著的位置，以便用户快速找到并进行搜索操作。

2. 醒目的功能：热搜榜功能和推荐内容是百度搜索的突出功能，放置在页面的右下方和左下方，分别引导用户浏览更多热门信息和优质内容，并提升百度的交互性和用户粘性。
3. 信息优化：百度搜索的页面布局通过信息的优化，使得搜索过程变得更为高效、便捷和直观。例如搜索工具栏的设计，让用户可以快速筛选出更加精准的搜索结果。
4. 美学设计：百度整个页面设计简洁、大方，色彩搭配和谐、统一，符合当下的审美趋势，也增加了用户的使用愉悦感。
5. 通过对页面的布局设计和优化，百度搜索为用户带来了快捷、高效、优质的搜索服务体验，也提高了用户粘性及百度的搜索引擎使用率。

## 1.2 分解元素

### 1.2.1 页头部分

上图中红色区域的部分代表了百度网站的页头区域。该区域主要可以分为百度的其他子功能的跳转按钮（左上角）以及所处位置天气展示以及个人账户的相关功能，包括切换账户、退出登录等功能（右上角）。这些功能与百度搜索本身的功能相关程度不大，相当于百度的一些附加功能，所以放在相对边缘的位置。但是有些用户可能对这些功能的需求很大，比如查看天气、查看地图等，所以放置在最容易被察觉的边缘，也就是最上端。



图 1.2: 页头分解

### 1.2.2 核心搜索区域

核心搜索区域由百度 logo、搜索框、搜索按钮三个主要元素组成，其中还包括使用图片搜索的照相机按钮。该区域是百度页面的核心区域，所以放置在页面的正中间。

百度的搜索框在整个页面中的位置是非常重要的，这个位置的选择与摆放需要考虑到用户的使用习惯、搜索习惯和网页布局等因素，包括以下几点：

1. 易识别性：搜索框应该在页面的明显位置，让用户第一时间注意到其存在，这样可以提高用户使用搜索引擎的效率。
2. 置顶优先：在整个搜索页面中，搜索框应该尽量放在页面的上方，这样可以减少页面的滚动操作，提高用户的使用体验。
3. 简洁明了：搜索框的设计应该简洁明了，方便用户快速进行输入和搜索操作，同时还要具有一定美感，和主题风格相一致。
4. 偏于默认：搜索框的输入框和搜索按钮可以使用默认颜色和样式，这样可以让用户对搜索框的操作更为自然，不会产生困惑和抵触。
5. 工具栏搭配：百度搜索框上面往往会配备搜索工具栏，包括搜索类型、热门搜索等，可以提供更为便捷的搜索体验，同时也可以展现百度搜索所具备的全面性。

总之，在确定搜索框的位置时，最重要的是要考虑到用户的需求和习惯，通过设计合适的位置和功能能够提高用户的满意度，也增加了搜索引擎的使用频率。



图 1.3: 核心搜索区域分解

### 1.2.3 热搜区域

百度将热搜区域放置在了页面的右侧并且根据热搜的排名进行展示，百度还在热搜区域的旁边设置了换一换按钮，可以切换百度热搜的内容。

1. 提高用户搜索效率：热搜区展示了当天最热门的搜索关键词，用户可以通过点击热门关键词快速进入相关的搜索结果，从而提高了搜索效率。
2. 增加搜索质量：热搜区域的热门关键词是根据用户实际搜索行为自动推荐的，可以反映当前大众关注和需求，因此，热搜区提供的搜索结果更符合用户的需求，从而提高了搜索质量。
3. 带动流量：热搜区可以吸引用户的注意力，促使用户进行更多的搜索操作，从而带动网站的流量，并有利于百度的广告营销。
4. 页面整洁性：将热搜区放在页面右侧，可以保持页面整洁简洁，同时不影响搜索框的主要功能，也不会影响用户的搜索体验。

总之，热搜区的设置和放置位置是百度为了提高搜索效率和质量，同时保持页面整洁和吸引用户注意力而做出的权衡和决定。



图 1.4: 热搜区域分解

### 1.2.4 推荐文章区域

百度还详细地推荐了一些新闻文章放置在了页面的左下角，其中每个文章都包括标题、图片、发表文章的媒体、发表日期以及朗读文章的选择项。

百度将推荐文章展示在网站的左下部，是出于以下几个考虑：

1. 提高服务质量：推荐文章一般是一些优质内容，可以帮助用户快速了解相关领域的最新趋势或知识点，提高用户对网站的信任度和忠诚度。

2. 引流：推荐相关内容可以引导用户浏览其他页面，吸引用户停留在网站内，提高网站的粘性并带来更多的流量。
3. 优化页面布局：将推荐文章放在左下部分，可以使网站页面布局更加优化，不会影响用户进行搜索操作，同时提高用户浏览内容的效率。
4. 个性化推荐：根据用户的搜索历史和偏好，推荐相关的文章，可以提高推荐的有效性，更好地满足用户需求。

总之，展示推荐文章并放置在左下部分，是百度为了提高服务质量和网站浏览体验，同时通过引流和个性化推荐来促进网站交互和用户粘性的举措。



图 1.5: 热搜区域分解

### 1.2.5 版权标识区域

百度在页面下方设置一些版权标志主要是为了保护原创内容的版权和知识产权，并提升搜索结果的可信度和权威性。具体原因如下：

1. 保护原创内容版权：网络上存在大量的原创内容，如文学、音乐、影视等，这些内容的创作者应该得到相应的版权保护，这样才能够鼓励原创和创意，提高内容质量。百度在页面下方添加版权标志可以提示用户检索到的搜索结果可能涉及版权或知识产权，提醒用户注意并尊重原创内容。
2. 提高搜索结果的可信度和权威性：在搜索结果页面下方添加版权标志可以彰显百度对知识产权保护的重视，提高搜索结果的可信度和权威性。而一些既存线上版权资质认证标志和线下合作品牌标志均能够辨识垃圾、豆腐渣等质量问题，提高了用户的搜索体验和满意度。
3. 展示百度的社会责任：百度是 ChinaNetCenter 的会员，在网络版权保护方面承担了一定的社会责任。在页面下方展示版权标志也是向社会各界展示百度的社会责任和公众形象。这样不仅可以提高百度的品牌形象，还可以增强公众对百度的信任和认可。

综上所述，百度在页面下方设置版权标志是出于对原创内容的版权保护、提高搜索结果可信度和权威性以及展示百度的社会责任等多方面的考虑。这种举措有助于保护原创内容、提高搜索结果的质量和满意度，在市场和社会上也能提供更好的公信力和美誉度。

图 1.6: 版权标志区域分解

### 1.3 识别交互

在百度检索界面中存在很多和用户交互的过程，我将抽取其中具有代表性的一些功能进行简述。

#### 1.3.1 鼠标悬停交互

百度的鼠标悬停页面交互主要是在鼠标停留在某个链接、图标或其他可交互元素上时触发，为用户提供更详细的信息或操作选项。具体的交互方式包括鼠标悬停在文字上时文字会变蓝，文字底部会增加下划线等。

视频

图 1.7: 鼠标悬停交互

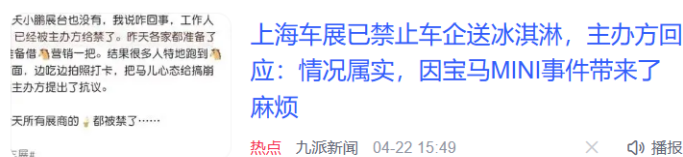


图 1.8: 鼠标悬停交互

- 3 建设美丽家园我们作了这些努力
- 4 警方通报网传水果摊主强买强卖
- 5 男子一口气买8公斤金条藏厕所楼道

图 1.9: 鼠标悬停交互

#### 1.3.2 下拉提示框展示

下拉提示框在一些多功能按钮被鼠标悬停或点击时展开，其中包含一些百度推荐的搜索内容以及子功能模块的跳转链接。



图 1.10: 下拉提示框交互



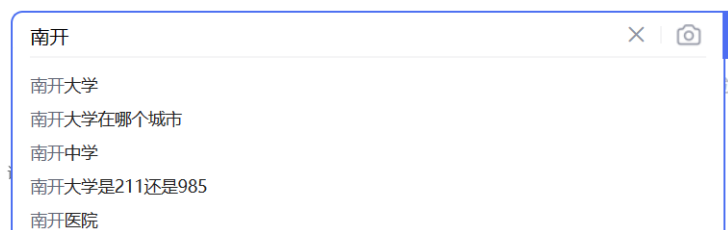


图 1.11: 下拉提示框交互

## 1.4 构建元素来源

百度搜索页面构建元素的来源主要包括以下几个方面：

1. HTML/CSS/JS: HTML 是百度搜索页面构建的基础，而 CSS 和 JS 则是页面样式和交互的关键元素。这些代码都是由百度前端工程师编写的，包括页面布局、颜色、字体、图片等各种构建元素。
2. 原型设计工具: 百度前端工程师在构建搜索页面之前通常会使用原型设计工具进行页面的设计和预览，例如 Axure RP、Sketch 等。这些工具可以协助设计师创建和验证页面的布局和交互，从而减少开发过程中出现的误差和漏洞。
3. 后端数据: 百度搜索页面的一些元素，例如搜索结果、商户信息、广告信息等，以及用户行为数据，例如用户搜索关键词和点击链接等，都是从后端服务器获取的。这些数据通常由后端工程师编写 API 并传送给前端工程师，前端工程师再将这些数据渲染到搜索页面上。
4. 开源库和框架: 百度前端工程师在构建搜索页面时通常使用各种开源库和框架，例如 jQuery、Bootstrap、Vue.js 等。这些开源库和框架提供了丰富的页面元素、交互和样式，大大加快了页面开发的速度和效率。

## 2 案例制作

在本次页面中，我设计了三个网页，第一个网页是搜索前的网页，第二个网页是检索后的网页，第三个网页是一张空白网页，所有具有跳转功能的链接最后都会跳转到第三章网页。

### 2.1 界面设计

最终的界面设计效果如下图所示：首先是搜索前的页面，以百度的原型网站为基础，我设计的搜索前的页面页头部分、核心搜索区域、热搜区域以及版权标识区域，其中页头区域位于页面的正上方边缘处，页头包括百度的一些子功能连接（左上部分）、以及当前位置天气显示以及个人账户信息（右上部分），核心搜索区域包括百度的 logo、搜索框、图片搜索按钮以及搜索按键。百度热搜部分就是调整到了核心搜索位置的正下方，版权标识区则依然位于整个页面的正下方边缘处。





图 2.12: 网站设计 page1

搜索后的页面我制作了南开大学校长陈雨露先生的百度搜索首页，效果图如下所示。设计的依托依然是原版百度后的搜索结果，其中最上方的依然是百度的 logo，但是由于这是搜索后的界面，所以搜索框已经不是核心内容了，最重要的内容应该是搜索之后的结果。所以搜索框和搜索按钮同样被调整到了最上端，同样位于最上端的还有百度的返回首页功能按钮以及个人的账户信息等跳转链接。

首先页面的做大半部分搜索结构是陈雨露先生的个人简介，其次是陈雨露校长近期的新闻，主要包括和其他领导共同参加工作会谈的新闻，再下面则是陈雨露校长的一些讲话视频，视频格式也按照百度的传统格式：视频截图 + 文字说明 + 视频来源。

页面的右一半包括了相关搜索，和陈雨露校长的相关搜索是南开大学的一些学院信息以及一些全国顶尖高效和学院的信息，再下方则是一些和天津、南开有关的百度热搜，便于用户了解一些被搜索者的周边信息。而百度的版权信息同样位于整个页面的最下方边缘处。



图 2.13: 网站设计 page2

## 2.2 交互设计

为了尽可能的吸收百度做的十分优秀的交互设计，我也在我的网页设计中融入了很多交互设计。

### 2.2.1 搜索提示框

搜索提示框的设计模仿百度进行设计，只要输入 nk，就可以触发搜索提示框展开，当搜索框内内容被清零时搜索提示框则隐藏

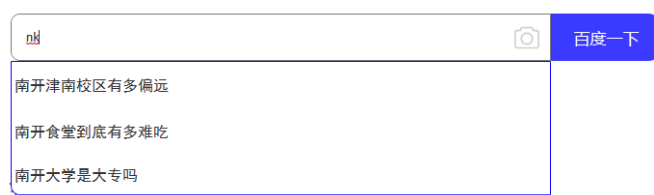


图 2.14: 搜索提示框

### 2.2.2 搜索外框变色

搜索外框变色，和讲解视频中 ppt 要求的一致，我也设计了搜索外框变色，实现原理与前者类似。

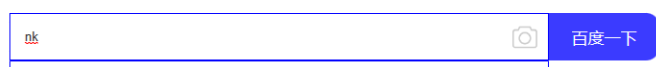


图 2.15: 输入框变色

### 2.2.3 带图片的提示框展开

鼠标移入“更多”可以展开提示框，里面包括很多百度的子功能，都可以点击进行跳转。



图 2.16: 带图片的提示框展开

### 2.2.4 移入变色

搜索后的很多文本框也采取了移入变色的设计思路：



图 2.17: 移入变色



图 2.18: 移入变色

## 2.3 总结

本次实验以百度搜索为原型,我实现了两个界面的复现和改进工作,通过本次学习,我掌握了 axure 软件的使用,了解了一些前端设计的思路。对于网页中的交互设计思路有了一些了解,相信这些知识在以后的开发、设计环节都会对我产生很多积极的影响。

## 3 第一层标题

### 3.1 第二层标题

第二层标题比第一层标题小。asdgasgasdkljgasjd asdgasdgasdg  
asdgadga

### 3.1.1 第三层标题

第三层标题比第二层标题小。

## 4 代码

### 4.1 伪代码

伪代码

---

**Algorithm 1** 初始化 obj 文件信息——对应 MeshSimplify 类中 readfile 函数, Face 类 calMatrix 函数

---

**Input:** obj 文件, 顶点、边、面列表

**Output:** 是否读取成功

```

1: function CALMATRIX(Face)
2:   normal  $\leftarrow e1 \times e2$ 
3:   normal  $\leftarrow normal / normal.length$ 
4:   temp[]  $\leftarrow normal.x, normal.y, normal.z, normal \cdot Face.v1$ 
5:   Matrix[i][j] = temp[i] * temp[j]
6:   return Matrix
7: end function
8: 根据 obj 的 v 和 f 区分点面信息, 读取并加入列表
9: scale  $\leftarrow$  记录点坐标中距离原点最远的分量, 以便后续 OpenGL 进行显示
10: ori  $\leftarrow$  记录中心点, 便于 OpenGL 显示在中心位置, 避免有的 obj 偏移原点较多
11: 根据三角面片信息, 计算一个面的三条边
12: 计算每个面的矩阵  $\leftarrow calMatrix$ 
13: 将每个面的矩阵加到各点, 由点维护
14: return True

```

---

### 4.2 代码

代码样式 1

---

```

1
2  int main() {
3      cout << "Hello World!" << endl;
4      return 0;
5  }

```

---

代码样式 2

逐列访问平凡算法

<pre> 1  void ord() 2  { 3      double head, tail, freq, head1, tail1, times=0; // timers 4      init(N); </pre>
--

```

5   QueryPerformanceFrequency ((LARGE_INTEGER *)&freq );
6   QueryPerformanceCounter ((LARGE_INTEGER *)&head);
7   for (int i=0; i<NN; i++)
8       for (int j=0; j<NN; j++)
9           col_sum[i] += (b[j][i]*a[j]);
10  QueryPerformanceCounter ((LARGE_INTEGER *)& tail) ;
11  cout << "\nordCol:" <<(tail-head)*1000.0 / freq<< "ms" << endl;
12  }

```

## 5 图和表格

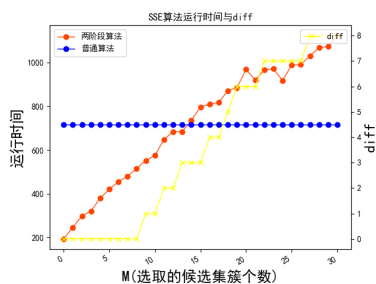
### 5.1 图

如图5.19所示

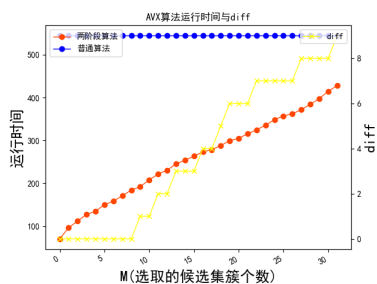


图 5.19: Caption

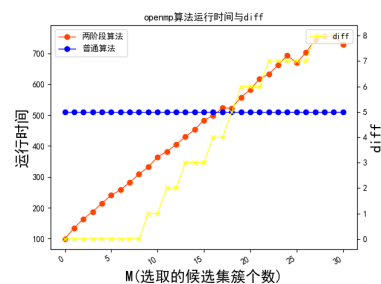
如图5.20所示。如图20(a)所示。



(a) SSE 的执行时间和准确率.



(b) AVX 的执行时间和准确率.



(c) OpenMP 的执行时间和准确率.

图 5.20: 不同并行优化算法的执行时间与准确率对比

表, 如表1所示。

N/n\Algo	naive-conv	naive-pool	omp-conv	omp-pool
64/2	0.0167	0.01255	0.04142	0.03799
64/4	0.03599	0.0394	0.0458	0.0421

表 1: 性能测试结果 (4 线程)(单位:ms)

带单元格表格, 如表2所示。

推荐你们使用这个网站, 将数据填进去, 设置好格式之后就可以自动生成表格的 latex 代码。

[TableGenerator](#)

<i>Cost</i>		To				
		<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>
From	<i>B</i>	7	0	1	3	8
	<i>C</i>	8	1	0	2	7
	<i>D</i>	8	3	2	0	5

表 2: 结点 C 距离向量表 (无毒性逆转)

## 6 其他会用到

### 6.1 枚举

带标号枚举

1. 1

2. 2

不带标号枚举

- 1

- 2

### 6.2 公式

多行公式

$$a + b = a + b \quad (1)$$

$$\frac{a + b}{a - b} \quad (2)$$

$$a + b = a + b \frac{a + b}{a - b} \quad (3)$$

单行公式：

$$\sum_{i=1}^N$$

行内公式： $\sum_{i=1}^N$

推荐你们使用 *Mathpix Snip*。是一个公式识别软件，可以自动生成公式对应的 latex 代码。

### 6.3 链接

参考文献 [3][1][2] [超链接](#) [YouTube](#)

## 参考文献

- [1] Gene H Golub and James M Ortega. *Scientific computing: an introduction with parallel computing*. Elsevier, 2014.
- [2] Stephen Bassi Joseph, Emmanuel Gbenga Dada, Sanjay Misra, and Samuel Ajoka. Parallel faces recognition attendance system with anti-spoofing using convolutional neural network. In *Illumination of Artificial Intelligence in Cybersecurity and Forensics*, pages 123–137. Springer, 2022.
- [3] Michael J Quinn. *Parallel computing theory and practice*. McGraw-Hill, Inc., 1994.