# 03. 索引与爬取设置：与谷歌高效沟通 — 详细扩展

目标：把课程里的“Index/NoIndex、robots.txt、sitemap”从概念提升为可落地执行的操作手册与检查清单。

## 目录

1. Index 与 NoIndex（原理、常见误区、实操示例）
2. robots.txt（语法、进阶用法、部署与测试）
3. Sitemap（生成、拆分、提交与监控）
4. 从0到1的实操步骤（把理论变成你网站上可运行的方案）
5. 常见问题与排查清单
6. 指标与工具清单

## 1. Index 与 NoIndex

### 1.1 基本原理（一句话）

* **index**：告知搜索引擎“把这个页面放到检索结果里”。
* **noindex**：告知搜索引擎“不要把这个页面放到检索结果里（但是可能仍能爬取）”。

### 1.2 如何实际告诉搜索引擎

* HTML meta：

<meta name="robots" content="index, follow">  
<meta name="robots" content="noindex, follow">

* HTTP Header（适用于 PDF、图片、API 响应或你不能修改 HTML 的场景）：

X-Robots-Tag: noindex, nofollow

### 1.3 与 robots.txt 的区别（关键误区）

* **robots.txt->Disallow** 是阻止**抓取**（crawling），但并不绝对阻止**索引**。如果外部有链接，Google 有可能仍然把 URL 索引（只是不展示页面内容）。
* **noindex** 需要搜索引擎能**抓取到页面**才能看到并执行 noindex 指令——因此如果你在 robots.txt 里 Disallow 了一个页面，Google 无法抓取到页面内的 meta noindex，那就不能用这种方式来可靠地删除索引。

**原则**：想要页面不被索引，同时仍要让搜索引擎知道不索引时，使用 noindex（允许抓取），而不是仅靠 robots.txt 的 Disallow。

### 1.4 常见用例与建议

* 感谢页 / 私有后台 / 测试页面：建议使用身份认证（最佳）或 noindex, nofollow。
* 重复内容（标签、筛选结果）：优先用 canonical 合并权重；无法合并时可用 noindex, follow。
* 资源文件（PDF）：若不希望被检索，使用 X-Robots-Tag: noindex。

### 1.5 实操小清单（决定 index/noindex）

* 这页是否为用户寻找的“目标页面”？是 → index；否 → 考虑 noindex。
* 是否存在更优的合并目标（canonical）？有 → 设置 canonical。
* 页面是否需要传递链接权重但不展示？ → noindex, follow。

## 2. robots.txt 深度解析

### 2.1 基本格式

User-agent: \*  
Disallow: /private/  
Allow: /private/ok-file.js  
Sitemap: https://example.com/sitemap.xml

* User-agent: 针对哪个爬虫（Google 用 Googlebot）。
* Disallow: 不允许抓取的路径。
* Allow（Google 支持）：允许抓取的子路径。
* 在文件末尾放 Sitemap: 可以帮助爬虫快速发现站点地图。

### 2.2 支持的模式（常用）

* \* 通配符（匹配任意字符串）。
* $ 表示结尾（匹配 URL 末尾）。

### 2.3 常见指令误区

* 不要用 robots.txt 阻止 CSS/JS：阻止后，Googlebot 无法渲染页面，会影响移动可用性和索引质量。
* 不要阻止 sitemap 自身或站点重要资源。

### 2.4 进阶用法

* 针对特定爬虫设置不同规则：

User-agent: Googlebot  
Disallow: /test/  
  
User-agent: \*  
Disallow:

* 使用 Crawl-delay：多数主流搜索引擎（Google）不支持该指令；针对其他爬虫可用。

### 2.5 部署与测试

1. 放到网站根目录（https://example.com/robots.txt）。
2. 在 Google Search Console 使用 **Robots.txt Tester**（检验规则是否生效）。
3. 监控抓取错误（Search Console -> Coverage / Crawl Stats）。

## 3. Sitemap（站点地图）实战

### 3.1 Sitemap 类型

* **XML Sitemap**（最常见，结构化）。
* **Sitemap Index**（当你有多个 sitemap 时，用索引文件指向它们）。
* **Text sitemap**（每行一个 URL，简洁但少用）。

### 3.2 基本 XML 结构示例

<?xml version="1.0" encoding="UTF-8"?>  
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">  
 <url>  
 <loc>https://example.com/page</loc>  
 <lastmod>2025-08-06</lastmod>  
 <changefreq>daily</changefreq>  
 <priority>0.8</priority>  
 </url>  
</urlset>

### 3.3 实务规则（必须记住）

* 单个 sitemap 最多 **50,000 URLs**，且未压缩文件不得超过 **50MB**。超过则需使用 sitemap index 拆分并压缩（.xml.gz）。
* lastmod 建议使用 ISO 8601（YYYY-MM-DD），且只在内容实际变更时更新。
* changefreq 和 priority 是提示性信息：合理填写即可，但不要滥用。

### 3.4 动态站点对策（电商、新闻）

* 按日期/分类拆分 sitemap（例如 sitemap-products-202509.xml）。
* 对于频繁更新内容，采用增量 sitemap（只把新/变更的 URL 放入新 sitemap）。
* 使用 Sitemap index 来聚合多个小 sitemap，提交 sitemap\_index.xml 到 Search Console。

### 3.5 提交与监控

* 在 robots.txt 添加 Sitemap: https://.../sitemap\_index.xml。
* 在 Google Search Console –> Sitemaps 提交 sitemap URL。
* 定期查看：提交状态、已读 URL 数、错误/警告列表。

## 4. 从 0 到 1 的实操步骤（实战流程）

**目标**：在你的网站上建立可被 Google 高效抓取和索引的体系。

1. **基础审查**：访问 https://yourdomain/robots.txt 和 https://yourdomain/sitemap.xml，把当前文件备份。
2. **检查误封**：用 Search Console 的 URL 检查工具检测关键页面是否被 noindex 或 robots 屏蔽。
3. **修复资源阻塞**：确保 CSS/JS 不被 robots.txt 阻止，保证页面可渲染。
4. **生成 sitemap**：对小站点可用静态生成脚本；大站点用增量/分片策略并压缩输出；把 sitemap URL 加入 robots.txt。
5. **提交 sitemap**：在 GSC 提交并请求索引关键页面（URL Inspection -> Request Indexing）。
6. **监控与迭代**：每天/周查看 Coverage、Crawl Stats、Search Performance，解决 4XX/5XX/重定向链等问题。

## 5. 常见问题与排查清单

**页面没有被收录**：检查顺序 1. robots.txt 是否 Disallow 了？ 2. 页面是否有 noindex meta？ 3. 页面是否被 canonical 指向别处？ 4. 是否有大量重复内容导致被归类为低质量？ 5. 服务器是否返回 5xx/4xx？

**想要快速移除页面**： - 最稳妥：页面允许抓取并返回 noindex（立即可在下次抓取后生效）。 - 临时：使用 Search Console 的“Remove URL”工具（注意：这是临时移除）。

**索引但内容不对/显示外部标题**：通常因为页面被 robots 禁止抓取，Google 只能依靠外部链接信息来展示该 URL。解决：解除阻止或添加 noindex。

## 6. 监控指标 & 推荐工具

### 关键指标

* 已提交 / 已索引 URL 数
* Index Coverage 报告中的错误（404、server errors）
* Crawl Stats（抓取频率、抓取成功率）
* Core Web Vitals / 页面速度

### 推荐工具（按用途）

* 官方：**Google Search Console**, **Bing Webmaster Tools**
* 抓取 & 审计：Screaming Frog, Sitebulb
* 日志分析：自建 ELK / AWStats / GoAccess，或用 Screaming Frog 配合服务器日志
* 测试：curl, robots.txt tester, URL Inspection

## 附录：常用命令 & 模板

### robots.txt 通用模板（允许全部抓取，但隐藏 /admin）

User-agent: \*  
Disallow: /admin/  
Allow: /  
Sitemap: https://example.com/sitemap\_index.xml

### meta noindex 示例

<!-- 永久不被索引，但允许爬虫跟随链接 -->  
<meta name="robots" content="noindex, follow">

### 使用 curl 检查 X-Robots-Tag

curl -I https://example.com/file.pdf | grep -i X-Robots-Tag

## 快速行动清单（复制到你的任务系统）

* 检查 robots.txt（是否意外阻止资源）
* 确认 sitemap 存在并在 robots.txt 中声明
* 在 GSC 中提交 sitemap 并监控索引数变化
* 对重要页面做 URL Inspection 并提交索引请求
* 对频繁变更内容建立自动化 sitemap 生成流程

如果你想： - 我可以把 robots.txt / sitemap 的示例文件打包成下载用的纯文本； - 或把“从0到1”的实操步骤转换为一个可运行的脚本（例如：静态站点的 sitemap 生成脚本或 cron 配置）。

告诉我你想先拿到哪一项，我就把对应的文件或脚本在画布里补上。