我在写技术文章时的一般手法与风格

作者：Hackyle；

更新时间：Wed Sep 27 16:28:16 CST 2023

本文主要说明我在写技术文章时，常用的写作方式与技巧。旨在告诉读者，本博客内的所有技术文章，都是在什么样的指导思想下写出来的。

这么做的目的是，希望读者尽可能快速、高效地获取本篇文章的核心内容，节省时间与精力，去做更多有意义的事情！

一篇文章的主要结构：

1. 标题
2. 摘要
3. 开头
4. 中间
5. 结尾

语言表达的一般原则：追求逻辑严密

- 表达观点：观点概括 + 解释 + 论证 + 论据（证明） + 例子 + 总结观点
- 做法解释：因为原理、性质，所以需要这么做
- 阐述问题：问题的现象 + 产生的背景 + 如何触发（如何复现）
- 解决问题：问题的现象 + 问题出现的原因 + 解决的具体方案 + 以后如何避免此类问题的发生

本文内容导览

文章标题

一般陈述，例如：我在写技术文章时的一般手法与风格

完整的起因和结果，或者只给起因或者结果，例如：

- 辞职考公的人，后来怎么样了？
- 这个东西不能碰，否则后悔莫及！

判断依据，例如：如何判断一个公司的好坏

如何做到，例如：教你1分钟学会取标题

读者的欲求，例如：想要提高XX能力；想要指导XX技术；想要做出XX成果

文章开头

- 本文的**主要内容**：想要告诉读者什么
- **前置知识**：读者需要指导什么背景知识，才能更加流畅地阅读本篇文章
- 为什么要写本篇文章，写这篇文章是基于什么**背景**
- 内容**导览**：一般是整篇文章的目录结构

- 事物解释
- 观点表达

解释某种事物：名词 --> 定义 --> 描述 --> 解释 --> 举例子 --> 总结

1. 名词：用名词指代要解释说明的事物
2. 定义：高度概括事物的基本属性和本质特征
3. 描述：通过各种语言手法对事物进行形象化的阐述
4. 解释：说明事物变化的原因，事物之间的联系，或者是事物发展的规律
5. 举例子：代表性的、恰当的事例来体现名词的指代，帮助读者构建事物画像
6. 总结：精确文本化事物的属性与含义，极力避免理解歧义

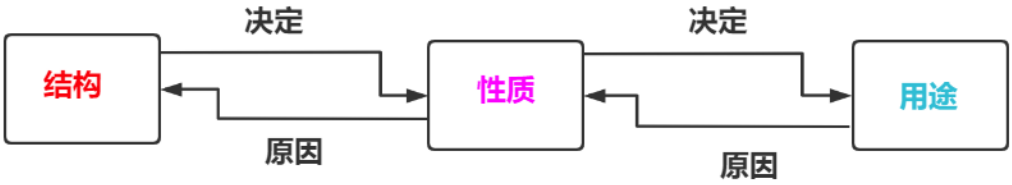
表达某种观点：主题/观点 --> 解释 --> 论证、论据 --> 举例子 --> 总结 --> 解决方案

1. 主题/观点
2. 解释
3. 论证、论据：理论、事实、引用；推理、归纳；对比、类比；因果、演绎；
4. 举例子
5. 总结
6. 解决方案

什么是技术

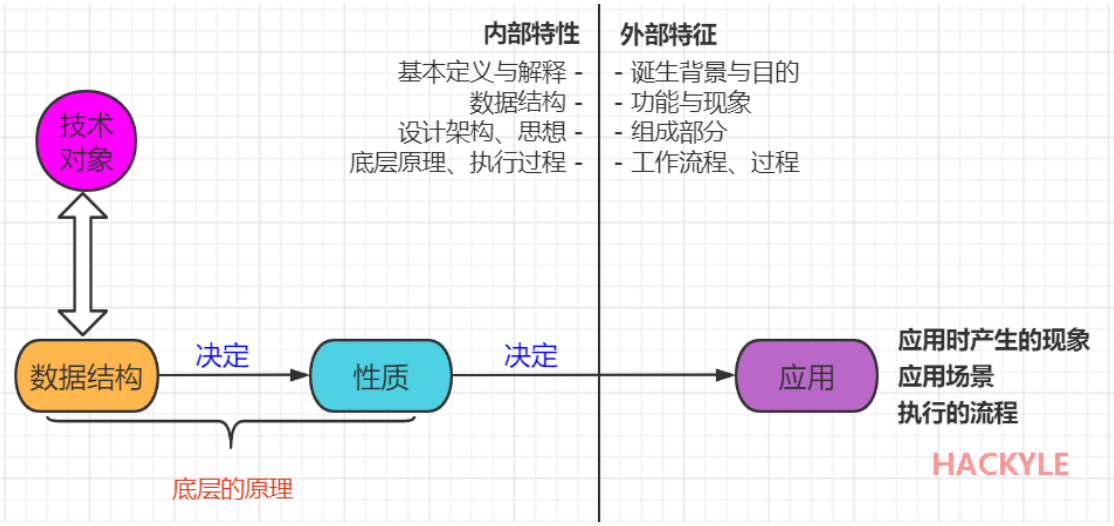
受化学的基本定义的启发，得出了技术的一般定义。

化学：是研究物质的组成、结构、性质及其变化规律的自然科学。



化学告诉我们：物质的结构决定物质的特性，物质的特性决定其用途

技术：研究技术的组成部分，数据结构、性质、应用场景，以及更好地解决问题。

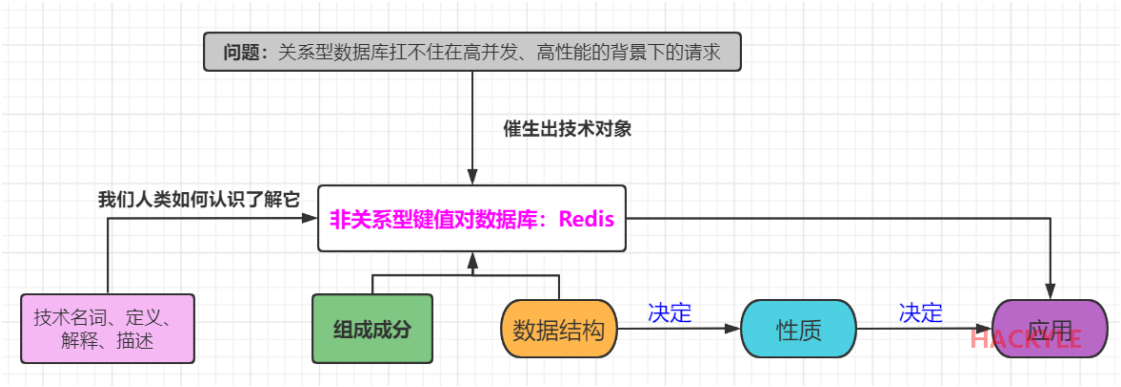


一项技术：数据结构决定了它有何种性质，性质决定了它有何种应用

例子：数组与链表的应用

- 数组的数据结构是线性顺序表，决定了它在随机查询时的高效特性，决定了它在随机查找的这种应用中有更高的效率

决随机查询少、增删操作多一类的问题（或者应用场景）时，最好是使用链表这种技术。



技术：Redis实例

为什么技术最终要更好地解决问题?

- 如上图，在高并发、高性能的要求下，当传统关系型数据库（例如MySQL）无法处理巨量的请求时，我们需要寻找更好的解决方案
- 我们进一步分析这个问题，MySQL它抗不住的原因是短时间内有大量的查询请求，并且许多查询操作都是一样的
- 可能的方案有：引入关系型数据库集群、对关系型数据库进行分布式处理（分库分表）.....
- 但考虑到成本与系统复杂度，我们尽可能选择成本与代价最小的解决方案：引入非关系型-内存型键值对数据库Redis，解决高并发的热数据查询请求。把查询请求分摊了
- 于是，在明白了Redis到底做了什么，解决了什么问题时，我们可以尽情投入到Redis相关细节知识的学习中

写法约束

本个标题下，我将阐述一些在写法上的强制要求。

客观性

- 一切的表达都是基于事实
- 所有描述和表达都有理可依，有据可判
- 不要写出以下内容：自己的主观判断、猜测、没有证明的结论等

代词的使用

- 代词指它前面出现过的名词、短语甚至整个句子，一定是前面出现过的
- 代词的位置和它要指向的目标不会隔得太远，1~3句话之内
- 代词的作用是减少小范围内某些词汇或句子重复出现的频率
- 代词前面出现的混淆目标如果太多，一定会重新调整句子，确保代词指向无歧义

例子1

- 反例：C++语言发明于1980年代，它支持“指针”和“面向对象（Object-Oriented）”两个特性，其价值在计算机编程语言历史上数一数二。
- 分析：上面这个句子中出现了两个代词“它”和“其”，抛开句子内容本身对错不论，第二个代词指向的对象其实并不明确，“其”指的是“指针”、“面向对象”还是“C++语言”？或者是指“C++语言同时支持.....两个特性”这个陈述?
- 解决方案：像这种有歧义的场合，我们应该少用代词，尽量用具体的主语去代替
- 正例：C++语言发明于1980年代，它支持“指针”和“面向对象（Object-Oriented）”两个特性，C++的价值在计算机编程语言历史上数一数二。

例子2

- 反例：该模块主要负责对视频进行解码，输出单张YUV格式的图片，并对输出的图片进行压缩和裁剪，前者基于Resize方法来完成，后者基于Crop()方法完成。
- 分析：代词前面提到了很多东西，“前者”和“后者”指向不明确，到底是指“解码”、“输出单张图片”，还是后面的“压缩”和“裁剪”？
- 正例1：该模块主要负责对视频进行解码，输出单张YUV格式的图片，并对输出的图片进行压缩和裁剪，压缩基于Resize方法来完成，裁剪基于Crop()方法完成。
- 正例2：该模块主要负责对视频进行解码，输出单张YUV格式的图片。同时，它还对输出的图片进行压缩和裁剪，前者基于Resize()方法完成，后者基于Crop()方法完成。

被动	主动
角色权限是由管理员	控制管理员控制角色权限
API结果无法被系统正常解析	系统无法正常解析API结果
图像特征是通过CNN逐步降维的方式提取的特征	CNN通过逐步降维的方式提取图像
这种检测效果无法被客户接受	客户无法接受这种检测效果
经过研发排查发现，这个现象是正常的	经过研发排查发现，这个属于正常现象

少用定性词，多用定量词

- 定性：形容词和副词，例如“尽可能”、“非常”、“很高”、“很多”
- 定量：具体数值

定性	定量
经过优化，接口响应速度提升明显	经过优化，接口响应速度提升2倍
很多人反应现场误报很多	数据统计发现，现场误报率为11%
大部分客户投诉说系统很不好用	最近一个月有超过50个客户投诉说系统不好用
升级依赖库后，该函数运行很快	将依赖库升级到2.3.1版本后，该函数执行时间缩短到100ms以内
研发同事很辛苦，每天加班	研发同事很辛苦，每天23:00之后才下班

术语的使用

- 术语的本质上是一个名词
- 通用术语（例如SDK、面向对象、TCP/IP、微服务等这些名词）：不会随意再重新去命名、调整或者改变拼写。
- 业务术语（是通用术语的反面）：在当前业务场景下作者新定义的术语，需要联系文档上下文去理解。
- 有概念的解释说明
- 有例子说明
- 任何方式的术语解释只有一次（术语第一次出现时）
- 术语的全称和简称会保持使用一致，要么整篇文档都用全称、要么都用简称

段落

- 与面向对象编程中“类的单一职责原则”一样，文档中的句子（特指以句号结尾的一句话）、段落也应该遵循“单一职责原则”。
- 一个段落只负责讲一个内容，两个不同的主题拆分成两个段落。
- 段落开头的第一句话，是整个段落的主要内容，后续句子都是在解释说明，描述、举例子、论证、总结等。

呈现方式

呈现方式：列表、表格、图片

列表

- 用于枚举、过程描述、要点归纳
- 列表中各项内容结构保持一致，都是名词、短语或者句子
- 有序列表：有先后顺序的步骤、流程

- 囊括所有：图有多大，就说明图上的每个地方都有用。例如展示核心内容发生的背景、元素之间的关系
- 突出重点：会标记、框出最核心的内容点

最佳实践

自问自答

- 根据某一主题、话题，对自己提问，然后自己回答。反复进行“自问自答”，这也是写文章的本质，在反复进行“自问自答”的过程中，引导读者从问题到解答的过程。
- 文章的质量取决于自问自答的质量。如果你想写得犀利一些，就要对自己提出犀利的问题；如果你想写得具体一点，就要对自己提出具体的问题；如果你想写得深刻一点，就要对自己提出深刻的问题。

内容拓展

- 首先进行横向拓展的自问自答，然后再进行纵向挖掘的自问自答。横向拓展决定了你将要写那些内容；纵向挖掘决定了你写得怎么样（深度）
- 横向拓展

1. 类似于头脑风暴，尽可能搜集与话题相关的点，对该点可以进行纵向挖掘；
2. 近义词和反义词：与话题的近似面与反例面；与话题相关的例子与完全无关的例子；
3. 类别：该话题下可以还可以有其他那些小类别，即再分类；

- 纵向挖掘

1. 是什么？吃的太快不好。【主题、观点】
2. 为什么？为什么吃得太快不好？【原因+分析+论证+例子】
3. 怎么样解决？怎样才能改掉吃得太快的坏习惯呢？还有其他解决方案吗？【解决方案】

文章结尾

- 列出重要的**参考文献**
- 告诉**读者**应该**注意**什么
- **总结**全文，明确地**提出结论**。尽量避免让读者产生“结论到底是什么？”、“作者到底想说什么呢？”的疑惑

Clemence：

写的很具体，非常好，希望之后能看到更多作品 🌷

Sun Dec 04 14:41:47 CST 2022 回复