

MySQL、Elasticsearch、Redis和MongoDB区别总结

关于MySQL、Elasticsearch、Redis和MongoDB在不同特性方面的区别:

![cgi-bin_mmwebwx-bin_webwxgetmsgimg_&MsgID=5030483865624037308&skey=@crypt_33141adc_94de3b9285f5a40b19f5a1230673ac24&mmweb_appid=wx_webfilehelper.jpg](https://p9-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/b11975a11782423b96b13a1634445957~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1024&h=1024&s=656037&e=jpg&b=053771)

特性	MySQL	Elasticsearch	Redis	MongoDB
数据模型	关系型数据库,使用表和行存储数据	分布式搜索引擎,使用倒排索引存储数据	键值数据库,支持多种数据结构	面向文档的NoSQL数据库
查询语言	SQL	DSL和查询DSL	基于数据结构的操作命令	类似于JavaScript的查询语言
事务支持	支持ACID事务	不直接支持事务	部分支持事务(仅针对单个操作)	支持事务(从4.0版本开始)
数据一致性	强一致性	最终一致性	强一致性	默认情况下强一致性,也支持最终一致性
扩展性	垂直扩展,可通过分库分表来扩展	水平扩展,易于扩展	支持主从复制和集群模式,可扩展读写能力	水平扩展,支持分片和复制集
复制策略	主从复制	多节点分片复制	主从复制和集群模式	复制集
使用场景	在线事务处理(OLTP)系统	全文检索、数据分析	缓存、消息队列、计数器等高速数据访问场景	大数据存储、内容管理等
查询性能	使用索引可提高查询性能	基于倒排索引,查询性能优秀	基于内存操作,查询性能极高	使用索引可提高查询性能
持久化	数据持久化存储在磁盘	数据持久化存储在磁盘	可配置持久化,默认存储在内存	数据持久化存储在磁盘
主键约束	支持主键约束	无主键约束概念	支持键值唯一性约束	支持主键约束
内存使用	可控,取决于数据量和配置	较高,取决于数据量和配置	可配置内存使用上限	可控,取决于数据量和配置

图表总结:

- * MySQL是传统关系型数据库,擅长OLTP场景,支持事务和SQL。
- * Elasticsearch是分布式搜索引擎,擅长全文检索和数据分析。
- * Redis是高性能键值数据库,适合缓存和高速数据访问场景。
- * MongoDB是面向文档的NoSQL数据库,适合大数据存储和内容管理场景。

特性列解释:

1. **数据模型**:

- * MySQL是一种传统的关系型数据库,采用表格和行来存储数据,数据之间通过关系建立联系。
- * Elasticsearch使用倒排索引来存储数据,适合于全文检索和分析。
- * Redis是一种键值数据库,支持多种数据结构(如字符串、列表、哈希、集合、有序集合等)的存储。
- * MongoDB是一种面向文档的NoSQL数据库,将数据存储为类似于JSON格式的文档。

2. **查询语言**:

- * MySQL使用SQL (Structured Query Language) 来进行数据的查询、更新、删除和插入等操作。
- * Elasticsearch使用DSL (Domain Specific Language) 和查询DSL来构建查询语句,支持丰富的搜索和分析功能。
- * Redis和MongoDB都没有专门的查询语言,但MongoDB提供了类似于JavaScript的查询语言。

3. **事务支持**:

- * MySQL支持事务,保证事务的ACID特性,可以进行回滚和提交等操作。
- * Elasticsearch在分布式场景下不直接支持事务,但可以通过乐观并发控制和部分事务支持来实现一定程度的数据一致性。
- * Redis仅支持单个操作的事务,复杂事务需要通过应用程序来保证数据的一致性。
- * MongoDB从版本4.0开始支持事务。

4. **数据一致性**:

- * MySQL和MongoDB都支持强一致性,保证在数据操作完成后数据的一致性和完整性。
- * Elasticsearch和Redis采用最终一致性,即数据操作的结果最终会达到一致状态,但在操作过程中可能存在一段时间的不一致。

5. **扩展性**:

- * MySQL和MongoDB支持垂直和水平扩展，可以通过分库分表或分片来扩展存储容量和吞吐量。
- * Elasticsearch和Redis都支持水平扩展，可以通过添加更多的节点来增加存储和处理能力。

6. **复制策略**:

- * MySQL和MongoDB都支持复制机制，可以通过主从复制和集群复制来实现数据的备份和故障恢复。
- * Elasticsearch和Redis都支持分片复制，可以将数据分布在多个节点上以提高数据的可用性和容错性。

7. **使用场景**:

- * MySQL适用于在线事务处理（OLTP）系统，如电子商务、金融系统等。
- * Elasticsearch适用于全文检索、日志分析和实时数据分析等场景。
- * Redis适用于缓存、消息队列、计数器等高速数据访问场景。
- * MongoDB适用于大数据存储、内容管理和实时分析等场景。

8. **查询性能**:

- * MySQL和MongoDB都支持索引来提高查询性能，可以根据需求对字段建立索引以加速查询。
- * Elasticsearch基于倒排索引，查询性能优秀，特别适用于全文搜索和复杂查询。
- * Redis基于内存操作，查询性能极高，适用于高速数据访问场景。

9. **持久化**:

- * MySQL、Elasticsearch和MongoDB都支持数据持久化存储在磁盘上，保证数据在重启后不丢失。
- * Redis可以通过配置来选择是否将数据持久化到磁盘，默认情况下数据存储在内存中。

10. **主键约束**:

- * MySQL和MongoDB支持主键约束，可以保证数据的唯一性和完整性。
- * Elasticsearch没有主键约束的概念，但可以通过数据模型设计来实现唯一性。
- * Redis并没有主键的概念，但可以通过键值唯一性约束来保证数据的唯一性。

11. **内存使用**:

- * MySQL、Elasticsearch和MongoDB的内存使用量受数据量和配置的影响，可以通过合理配置来控制内存的使用。
- * Redis的内存使用量较高，但可以通过配置来限制内存使用上限，避免出现内

存溢出的情况。

概念及优缺点

以下是对MySQL、Elasticsearch、Redis和MongoDB的概念、优缺点进行总结对比：

MySQL

![Designer.png](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/6b2fa163f5f945b2b1e62b59986c91c4~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1792&h=1024&s=2917636&e=png&b=43697c)

* 概念：关系型数据库，基于关系模型，使用表和行存储数据。

* 优点：

- + 支持ACID事务，数据具有很高的一致性和完整性。
- + 功能丰富，支持SQL标准查询。

+ 社区活跃，有大量成熟的工具和资源。

* 缺点：

- + 垂直扩展能力有限，需要分库分表等方式扩展。
- + 对于复杂的查询和大量的连接操作效率较低。
- + 固定的模式结构，不太适合存储非结构化数据。

Elasticsearch

![Designer (1).png](https://p9-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/f8371966fb4b4747be311773ceaf27a5~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1792&h=1024&s=2776666&e=png&b=bf7db)

* 概念：分布式搜索引擎和分析引擎，基于Lucene，使用倒排索引存储数据。

* 优点：

- + 全文检索能力极佳，支持各种查询和分析操作。
- + 易于水平扩展，支持分片和复制机制。
- + 提供RESTful API，客户端语言多样。

* 缺点：

- + 不支持事务，最终一致性可能会导致数据不一致。
- + 对于更新和删除操作相对比较慢。

- + 复杂的查询和聚合操作占用大量资源。

Redis

![Designer (2).png](https://p1-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/6ee3e6d59f9a466890a421776630a6be~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1792&h=1024&s=1636780&e=png&b=b3d2df)

- * 概念:开源的内存键值数据库,支持多种数据结构。

- * 优点:

- + 极高的性能,基于内存操作,适合高速缓存。
- + 支持数据持久化,可将内存中的数据保存到磁盘。
- + 丰富的数据结构,如字符串、列表、集合等。
- + 支持主从复制和集群,可扩展读写能力。

- * 缺点:

- + 内存占用较高,对内存资源要求较多。
- + 单线程反应bereiter会a大量并发连接,有性能瓶颈。
- + 不支持真正意义上的事务,只能保证单个操作的原子性。

MongoDB

![Designer (3).png](https://p1-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/86f9d464566947ceba0e8fb09d5f0ec8~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1792&h=1024&s=2141586&e=png&b=d2eef8)

- * 概念:文档型数据库,属于NoSQL数据库,文档类似于JSON格式。

- * 优点:

- + 灵活的数据模型,很适合存储层次结构复杂、无固定模式的数据。
- + 支持水平扩展,通过分片和复制集实现扩展。
- + 从4.0版本开始支持事务,提供更好的数据一致性。
- + 丰富的查询语言,类似于面向对象查询。

- * 缺点:

- + 对于大量的非结构化数据,查询性能较差。
- + 没有成熟的SQL支持和JOIN关联操作。
- + 每个文档由于需要拷贝所有键名,存在一定的内存开销。

总结:

MySQL更加适合在线事务处理、关系型数据存储场景;

Elasticsearch擅长全文检索和数据分析;

Redis适合作为缓存使用和高并发的数据访问场景;

MongoDB适合存储非结构化数据或者层次结构复杂的数据。

原文链接: <https://juejin.cn/post/7362084722821038120>