阿里云
日志服务

产品简介

文档版本：20181213

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的”现状“、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含”阿里云”、Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **禁止：** 重置操作将丢失用户配置数据。
	该类警示信息可能导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **警告：** 重启操作将导致业务中断，恢复业务所需时间约10分钟。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 **说明：** 您也可以通过按**Ctrl** + **A**选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
粗体	表示按键、菜单、页面名称等UI元素。	单击 **确定**。
`courier` 字体	命令。	执行 `cd /d C:/windows` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *`Instance_ID`*
[]或者[a|b]	表示可选项，至多选择一个。	`ipconfig` *`[-all|-t]`*
{}或者{a|b}	表示必选项，至多选择一个。	`swich` *`{stand | slave}`*

目录

1 什么是日志服务

日志服务（Log Service，简称 LOG）是针对日志类数据的一站式服务，在阿里巴巴集团经历大量大数据场景锤炼而成。您无需开发就能快捷完成日志数据采集、消费、投递以及查询分析等功能，提升运维、运营效率，建立 DT 时代海量日志处理能力。

日志服务 学习路径

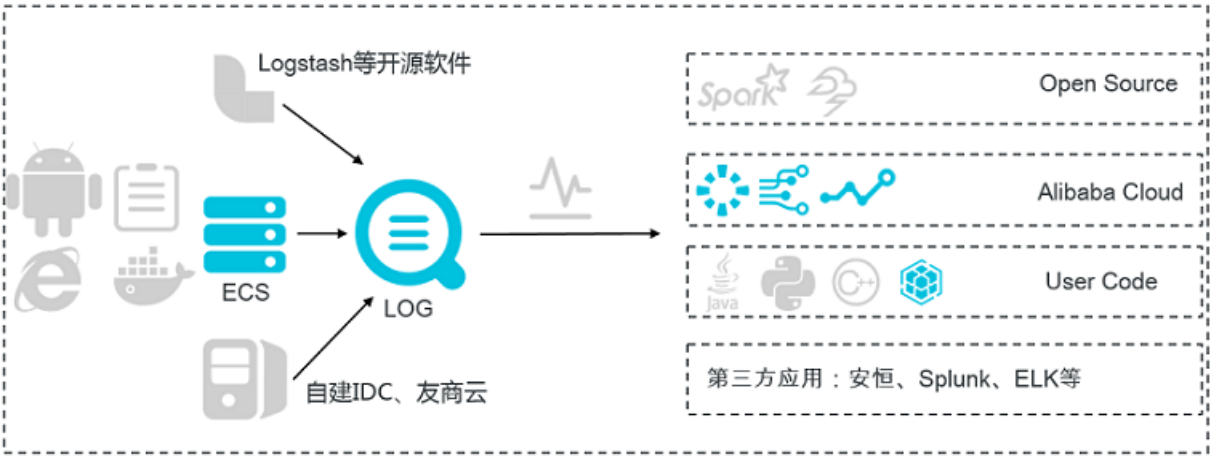
日志服务学习路径图为您推荐热门功能的操作指引文档，帮助您快速了解日志服务产品。视频与文档结合，全方位提升您的产品使用及文档阅读体验。

实时采集与消费（LogHub）

功能：

- 通过ECS、容器、移动端，开源软件，JS等接入实时日志数据（例如Metric、Event、BinLog、TextLog、Click等）
- 提供实时消费接口，与实时计算及服务对接

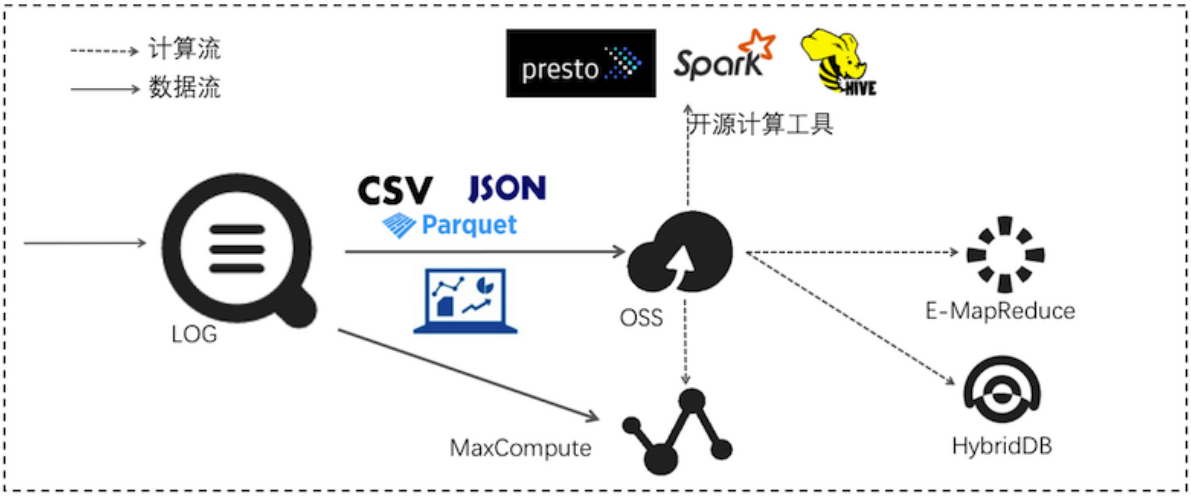
用途：数据清洗（ETL），流计算（Stream Compute），监控与报警， 机器学习与迭代计算。



投递数仓（LogShipper）

稳定可靠的日志投递。将日志中枢数据投递至存储类服务进行存储。支持压缩、自定义Partition、以及行列等各种存储方式。

用途：数据仓库 + 数据分析、审计、推荐系统与用户画像。

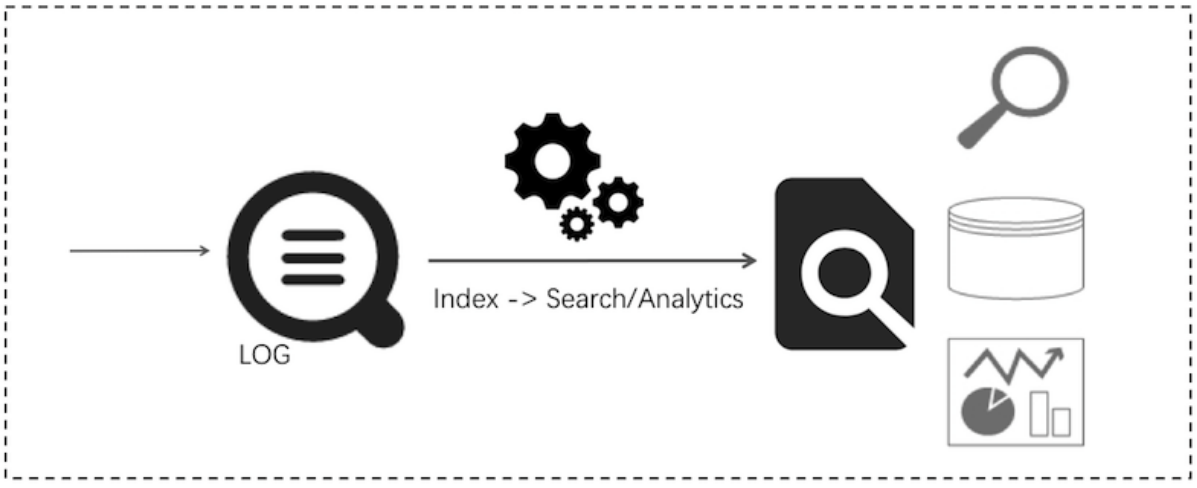


查询与实时分析（Search/Analytics）

实时索引、查询分析数据数据。

- 查询：关键词、模糊、上下文、范围
- 统计：SQL聚合等丰富查询手段
- 可视化：Dashboard + 报表功能
- 对接：Grafana，JDBC/SQL92

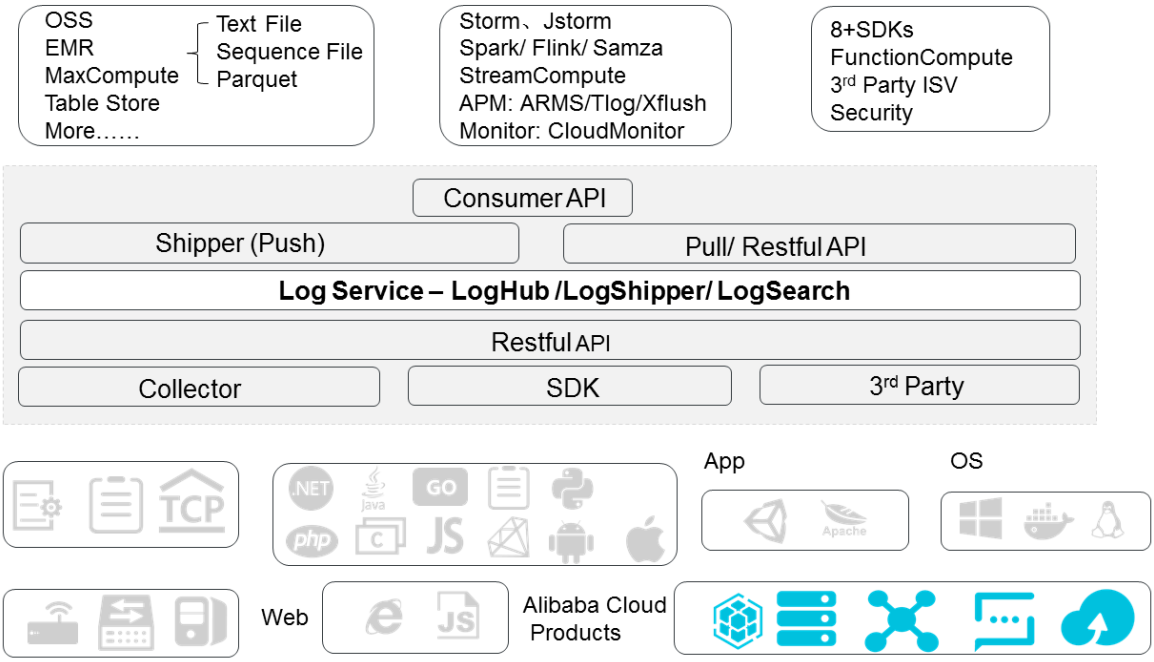
用途：DevOps/线上运维，日志实时数据分析，安全诊断与分析，运营与客服系统。



2 产品架构

日志服务的架构如下图所示。

图 2-1: 产品架构



Logtail

帮助您快速收集日志的Agent。其特点如下所示：

- 基于日志文件、无侵入式的收集日志
  - 只读取文件。
  - 日志文件无侵入。
- 安全、可靠
  - 支持文件轮转不丢失数据。
  - 支持本地缓存。
  - 网络异常重试。
- 方便管理
  - Web端操作。
  - 可视化配置。
- 完善的自我保护

- 实时监控进程CPU、内存消耗。
- 限制使用上限。

前端服务器

采用LVS + Nginx构建的前端机器。其特点如下所示：

- HTTP、REST协议
- 水平扩展
  - 流量上涨时可快速提高处理能力。
  - 支持增加前端机。
- 高吞吐、低延时
  - 纯异步处理，单个请求异常不会影响其他请求。
  - 内部采用专门针对日志的Lz4压缩，提高单机处理能力，降低网络带宽。

后端服务器

后端是分布式的进程，部署在多个机器上，完成实时对Logstore数据的持久化、索引、查询以及投递至MaxCompute。整体后端服务的特点如下所示：

- 数据高安全性：
  - 您写入的每条日志，都会被保存3份。
  - 任意磁盘损坏、机器宕机情况下，数据自动复制修复。
- 稳定服务：
  - 进程崩溃和机器宕机时，Logstore会自动迁移。
  - 自动负载均衡，确保无单机热点。
  - 严格的Quota限制，防止单个用户行为异常对其他用户产生影响。
- 水平扩展：
  - 以分区（Shard）为单位进行水平扩展。
  - 用户可以按需动态增加分区来增加吞吐量。

3 产品优势

3.1 功能优势

全托管服务

- 应用性强，5分钟即可接入服务进行使用，Agent支持任意网络下数据采集。
- LogHub覆盖Kafka 100%功能，提供完整监控、报警等功能数据，并支持弹性伸缩（可支持PB/Day规模），使用成本为自建50%以下。
- LogSearch/Analytics 提供快速查询、仪表盘和报警功能，使用成本为自建 20%以下。
- 0+接入方式，与云产品（OSS/E-MapReduce/MaxCompute/Table Store/MNS/CDN/ARMS等）、开源软件（Storm、Spark）无缝对接。

生态丰富

- LogHub 支持30+采集端，包括Logstash、Fluent等，无论是嵌入式设备，网页，服务器，程序等都能轻松接入。在消费端，支持与Storm、Spark Streaming、云监控等对接。
- LogShipper 支持丰富数据格式（TextFile、SequenceFile、Parquet等），支持自定义Partition，数据可以直接被Presto、Hive、Spark、Hadoop、E-MapReduce、MaxCompute、HybridDB等处理。
- LogSearch/Analytics 查询分析语法完整、兼容SQL92、支持JDBC协议与Grafana对接。

实时性强

- LogHub：写入即可消费；Logtail（采集Agent）实时采集传输，1秒内到服务端（99.9%情况）。
- LogSearch/Analytics：写入即可查询分析，在多个查询条件下1秒可查询10亿级数据，多个聚合条件下1秒可分析1亿级数据。

完整API/SDK

- 轻松支持自定义管理及二次开发。
- 所有功能均可通过API/SDK实现，提供多种语言SDK，可轻松管理服务和百万级设备。
- 查询分析语法简单便捷，兼容SQL92；接口友好、适合与生态软件对接，提供Grafana对接方案。

3.2 成本优势

成本优势

日志服务产品在日志处理的三种场景下具有以下成本优势：

- LogHub：
  - 以购买云主机 + 云磁盘搭建 Kafka 相比，对于 98% 场景下用户价格有优势。对小型网站而言，成本为 kafka 的30% 以下。
  - 提供 RESTful API，可以直接针对移动设备提供数据收集功能，节省了日志收集网关服务器的费用。
  - 免运维，随时随地弹性扩容使用。
- LogShipper：
  - 无需任何代码/机器资源，灵活配置与丰富监控数据
  - 规模线性扩展（PB级/Day），功能当前免费
- LogSearch/Analytics：
  - 以购买云主机 + 自建 ELK 相比，成本为自建的 15% 以下，并且查询能力与数据规模有极大提升。日志管理软件相比，能无缝各种流行支持流计算 + 离线计算框架，日志流动畅通无阻

成本对比

以下是在计费模型下，日志服务功能与自建方案对比，仅供参考。

日志中枢（LogHub vs Kafka）

-	关注点	LogHub	自建中间件（如Kafka）
使用	新增	无感知	需要运维动作
	扩容	无感知	需要运维动作
	增加备份	无感知	需要运维动作
	多租户使用	隔离	可能会相互影响
费用	公网采集（10GB/天）	2 元/天	16.1 元/天
	公网采集（1TB/天）	162 元/天	800 元/天
	内网采集（数据量小）	-	-
	内网采集（数据量中）	-	-

-	关注点	LogHub	自建中间件（如Kafka）
	内网采集（数据量大）	-	-

日志存储与查询引擎

关注点-		LogSearch	ES（Lucene Based)	NoSQL	Hive
规模	规模	PB	TB	PB	PB
成本	存储（元/GB *天）	0.0115	3.6	0.02	0.035
	写入（元/GB）	0.35	5	0.4	0
	查询（元/GB）	0	0	0.2	0.3
	速度-查询	毫秒级-秒级	毫秒级-秒级	毫秒级	分钟级
	速度-统计	弱+	较强	弱	强
延时	写入->可查询	实时	分钟级	实时	十分钟级

说明：

此处价格对比主要基于ECS上部署软件，并且设置为3份副本后的计算结果。

更多内容请参考：*#unique_8*

3.3 查询分析全方位对比（ELK）

提到日志实时分析，很多人都会想到很火的ELK Stack（Elastic/Logstash/Kibana）来搭建。ELK方案开源，在社区中有大量的内容和使用案例。

阿里云日志服务产品在新版中增强查询分析功能（LogSearch/Analytics），支持对日志数据实时索引与查询分析，并且对查询性能和计算数据量做了大量优化。在这里我们做一个全方位的比较，对于用户关心的点，我们依次展开分析：

- 易用：上手及使用过程中的代价

- 功能（重点）：主要针对查询与分析两块
- 性能（重点）：对于单位大小数据量查询与分析需求，延时如何
- 规模（重点）：能够承担的数据量，扩展性等
- 成本（重点）：同样功能和性能，使用分别花多少钱

易用

对日志分析系统而言，有如下使用过程：

1. 采集：将数据稳定写入
2. 配置：如何配置数据源
3. 扩容：接入更多数据源，更多机器，对存储空间，机器进行扩容
4. 使用：这部分在功能这一节介绍
5. 导出：数据能否方便导出到其他系统，例如做流计算、放到对象存储中进行备份
6. 多租户：如何将数据分享给他人使用，使用是否安全等

以下是比较结果：

项目	小项	自建**ELK**	**Log Analytics**
采集	协议	Restful API	Restful API
	Agent	Logstash/Beats/ FluentD，生态十分丰富	Logtail（为主）+其他（例如Logstash）
配置	单元	提供Index概念用以区分不同日志	Project + Logstore。提供两层概念，Project相当于命名空间，可以在Project下建立多个Logstore
	属性	API + Kibana	API + SDK + 控制台
扩容	存储	增加机器，购买云盘	无需操作
	机器	新增机器	无需操作
	配置	配置Logstash并通过配管系统应用机器	控制台/API 操作，无需配管系统
	采集点	通过配管系统控制，将配置和Logstash安装到机器组	控制台/API 操作，无需配管系统

项目	小项	自建ELK	Log Analytics
导出	方式	API/SDK	API/SDK + 各流计算引擎（Spark，Storm，Flink，StreamCompute，CloudMonitor，ARMS） + 存储（OSS + MaxCompute）
多租户	安全	无	https + 传输签名 + 多租户隔离 + 访问控制
	使用	同一账号	子账号，角色，产品，临时授权等

总体而言：

ELK 有非常多的生态和写入工具，使用中的安装、配置等都有较多工具可以参考。LogSearch/Analytics是一个托管服务，从接入、配置、使用上集成度非常高，普通用户5分钟就可以接入。并且过程中不需要担心容量、并发等问题，按量计费，弹性伸缩。

功能（查询+分析）

查询主要将符合条件的日志快速命中，分析功能是对数据进行统计与计算。

例如我们有如下需求：所有状态码大于200读请求，根据Ip统计次数和流量，这样的分析请求就可以转化为两个操作：查询到指定结果，对结果进行统计分析。在一些情况下我们也可以不进行查询，直接对所有日志进行分析。

```
1. Status in (200,500] and Method:Get*
 2. select count(1) as c, sum(inflow) as sum_inflow, ip group by Ip
```

查询能力对比

类型	小项	自建ELK	Log-Search/Analytics
文本	索引查询	支持	支持
	分词	支持	支持
	中文分词	支持	支持
	前缀	支持	支持
	后缀	支持	

类型	小项	自建ELK	Log-Search/ Analytics
	模糊	支持	支持
	Wildcast	支持	（计划支持）
数值	long	支持	支持
	double	支持	支持
Nested	Json查询	支持	
Geo	Geo查询	支持	不直接支持，但可以通过区间查询代替
Ip	Ip查询	支持	不直接支持，通过字符串支持
上下文	上下文查询		支持
	上下文过滤		支持

ElasticSearch 支持的数据类型，以及查询方式上要更强。Log-Search/Analytics 支持大部分常用查询，也有一些特色（例如程序日志上下文查询与展开）。

分析能力对比

- *ES 5.5 Aggregation*
- *#unique_10*

类型	小项	自建ELK	Log-Search/ Analytics
接口	方式	API/SDK	API/SDK + SQL92
	其他协议		JDBC
Agg	Bucketing	支持	支持
	Metric	支持	支持
	Matrix	支持	支持
	Pipeline	有限支持	完整支持
运算	数值		支持
	字符串		支持
	估算		支持

类型	小项	自建ELK	Log-Search/Analytics
	数理统计		支持
	日期转换		支持
GroupBy	Agg	支持	支持
	Having条件		支持
排序	排序		支持
Join	多表Join		支持

从结果看Log-Search/Analytics 提供的功能是ES 超集，完整支持SQL 92 语义，只要会写SQL 都能直接使用。

性能

接下来我们测试针对相同数据集，分别对比写入数据及查询，和聚合计算能力。

实验环境

1. 测试配置

类别	自建ELK	LogSearch/LogAnalytics
环境	ECS 4核16GB * 4台 + 高效云盘 SSD	-
Shard	10	10
拷贝数	2	3（默认配置，对用户不可见）

2. 测试数据

   - 5列double
   - 5列long
   - 5列text，字典大小分别是256，512，768，1024，1280

   以上字段完全随机，如下为一条测试日志样例：

```
timestamp:August 27th 2017, 21:50:19.000
 long_1:756,444 double_1:0 text_1:value_136
 long_2:-3,839,872,295 double_2:-11.13 text_2:value_475
 long_3:-73,775,372,011,896 double_3:-70,220.163 text_3:value_3
 long_4:173,468,492,344,196 double_4:35,123.978 text_4:value_124
```

```
long_5:389,467,512,234,496 double_5:-20,10.312 text_5:value_1125
```

3. 数据集大小

   - 原始数据大小：50 GB
   - 原始数据除去Key后内容大小：27 GB（LogSearch/Analytics 以该大小作为存储计费单元）
   - 日志行数：162,640,232 （约为1.6亿条）

写入测试结果

ES采用bulk api批量写入，LogSearch/Analytics 用PostLogstoreLogs API批量写入，结果如下：

类型	项目	自建ELK	LogSearch/Analytics
延时	平均写入延时	40 ms	14 ms
存储	单拷贝数据量	86G	58G
	膨胀率：数据量/原始数据大小	172%	121%

图 3-1: 平均写入延时对比

图 3-2: 数据膨胀率对比

说明：

LogSearch/Analytics 产生计费的存储量包括压缩的原始数据写入量（23G）+索引流量（27G），共50G存储费用。

从测试结果来看

- LogSearch/Analytics写入延时好于ES，14 ms vs 40ms
- 空间：原始数据50G，因测试数据比较随机所以存储空间会有膨胀（大部分真实场景下，存储压缩后会比原始数据小）。ES胀到86G，膨胀率为172%，在存储空间超出LogSearch/Analytics 58%

读取（查询+分析）测试

测试场景

我们在此选取两种比较常见的场景：日志查询和聚合计算。分别统计并发度为1，5，10时，两种case的平均延时。

1. 针对全量数据，对任意text列计算group by，计算5列数值的avg/min/max/sum/count，并按照count排序，取前1000个结果，例如：

```
select count(long_1) as pv,sum(long_2),min(long_3),max(long_4),sum(
long_5)
group by text_1 order by pv desc limit 1000
```

2. 针对全量数据，随机查询日志中的关键词，例如查询“value_126”，获取命中的日志数目与前100行，例如：

```
value_126
```

测试结果

类型	并发数	**ES**延时（单位**s**）	**LogSearch/Analytics** 延时（单位**s**）
case1：分析类	1	3.76	3.4
	5	3.9	4.7
	10	6.6	7.2
case2：查询类	1	0.097	0.086
	5	0.171	0.083
	10	0.2	0.082

结果分析

- 从结果看，对于1.5亿数据量这个规模，两者都达到了秒级查询与分析能力。
- 针对统计类场景（case 1），ES和日志服务延时处同一量级。ES采用SSD云盘，在读取大量数据时IO优势比较高。
- 针对查询类场景（case 2），LogAnalytics在延时明显优于ES。随着并发的增加，ELK延时对应增加，而LogAnalytics延时保持稳定甚至略有下降。

图 3-3: case1 分析类延时对比

图 3-4: case2 查询类延时对比

规模

1. LogSearch/Analytics 一天可以索引PB级数据，一次查询可以在秒级过几十TB规模数据，在数据规模上可以做到弹性伸缩与水平扩展
2. ES比较适合服务场景为：写入GB-TB/Day、存储在TB级。主要受限于2个原因：
   - 单集群规模：比较理想为20台左右，业界比较大的为100节点一个集群
   - 写入扩容：shard创建后便不可再修改，当吞吐率增加时，需要动态扩容节点，最多可使用的节点数便是shard的个数
   - 存储扩容：主shard达到磁盘的上线时，要么迁移到更大的一块磁盘上，要么只能分配更多的shard。一般做法是创建一个新的索引，指定更多shard，并且rebuild旧的数据

   LogSearch/Analytics 则没有扩展性方面的问题，每个shard都是分布式存储。并且当吞吐率增加时，可以动态分裂shard，达到处理能力水平扩展。

成本

以上述测试数据为例，一天写入50GB数据（其中27GB 为实际的内容），保存90天，平均一个月的耗费。

1. 日志服务（LogSearch/LogAnalytics）计费规则参考计费方式，包括读写流量、索引流量、存储空间等计费项，查询功能免费。

计费项目	值	单价	费用（元)
读写流量	23G * 30	0.2 元/GB	138
存储空间（保存90天）	50G * 90	0.3 元/GB*Month	1350
索引流量	27G * 30	0.35 元/GB	283
总计	-	-	1771

2. ES费用包括机器费用，及存储数据SSD云盘费用
   - 云盘一般可以提供高可靠性，因此我们这里不计费副本存储量。
   - 存储盘一般需要预留15%剩余空间，以防空间写满，因此乘以一个1.15系数。

计费项目	值	单价	费用（元)
服务器	4台4核16G（三个月）（ecs.mn4.xlarge）	包年包月费用：675 元/Month	2021
存储	86 * 1.15 * 90 （这里只计算一个副本）	SSD：1 元/GB*M	8901
	-	SATA：0.35 元/GB*M	3115
总计			12943 （SSD）
			5135 （SATA）

图 **3-5:** 成本对比

同样性能，使用LogSearch/Analytics与ELK（SSD）费用比为 13.6%。在测试过程中，我们也尝试把SSD换成SATA以节省费用（LogAnalytics与SATA版费用比为 34%），但测试发现延时会从40ms上升至150ms，在长时间读写下，查询和读写延时变得很高，无法正常工作了。

总结

LogSearch/Analytics在提供同样查询速度、更高吞吐量、更强分析能力背后，只需要开源方案13%成本，并且按量付费免运维，让您把精力放在业务分析上。

日志服务除了LogSearch/Analytics外，还提供LogHub、LogShipper功能，支持实时采集数据、与流计算系统（Spark、Storm、Flink、StreamCompute）、离线分析系统（MaxCompute、E-MapReduce、Presto、Hive等）打通，提供一站式实时数据解决方案。

3.4 查询方案（ELK/Hive）对比

DevOps 场景日志查询方案对比：ELK（搜索类）、Hadoop/Hive

在互联网大潮中，为应对不断加速的软件、服务交付需求，无论是在创业团队还是大型互联网公司，都已经转向或逐步转向 DevOps 模式，通过开发（Dev）和运维支持（Ops）之间有效协作，解决跨部门协作、快速响应客户需求、进行持续交付。

日志在 DevOps 下显得越发重要，无论是在问题调查、安全审计、运营支撑等各方面，日志都起到了重要的支撑作用。一个合适的日志解决方案，对于 DevOps 显得尤为重要。

我们从如下方面考察 LogSearch 与 ELK、Hadoop/Hive 类方案的对比：

- 延时：日志产生后，多久可查询
- 查询能力：单位时间扫描数据量
- 查询功能：关键词查询、条件组合查询、模糊查询、数值比较、上下文查询
- 弹性：快速应对百倍流量上涨
- 成本：每 GB 费用
- 可靠性：日志数据安全不丢失

常用方案以及对比

- 自建 ELK：通过 Elastic、Logstash、Kibana 进行对比
- 离线 Hadoop + Hive：将数据存储在 Hadoop，利用 Hive 或 Presto 进行查询（非分析）
- 使用日志服务（LogSearch）

以应用程序日志和 Nginx 访问日志为例（每天 10GB），对比几种方案。

功能项	ELK 类系统	Hadoop + Hive	日志服务
可查延时	1~60 秒(由 refresh_interval 控制)	几分钟~数小时	实时
查询延时	小于 1 秒	分钟级	小于 1 秒
超大查询	几十秒~数分钟	分钟级	秒级（查询 10 亿日志）
关键词查询	支持	支持	支持
模糊查询	支持	支持	支持
#unique_13	不支持	不支持	支持
数值比较	支持	支持	支持
连续字符串查询	支持	支持	不支持
弹性	提前预备机器	提前预备机器	秒级 10 倍扩容
写入成本	写入5 元/GB，查询免费	写入免费，查询一次 0.3 元/GB	写入 0.5 元/GB，查询免费
存储成本	≤ 3.36 元/GB*天	≤ 0.035 元/GB*天	≤ 0.016 元/GB*天
可靠性	设置拷贝数	设置拷贝数	SLA > 99.9%，数据 > 99.99999999%

4 应用场景

日志服务的典型应用场景包括：数据采集、实时计算、数仓与离线分析、产品运营与分析、运维与管理等场合。典型应用场景如下。

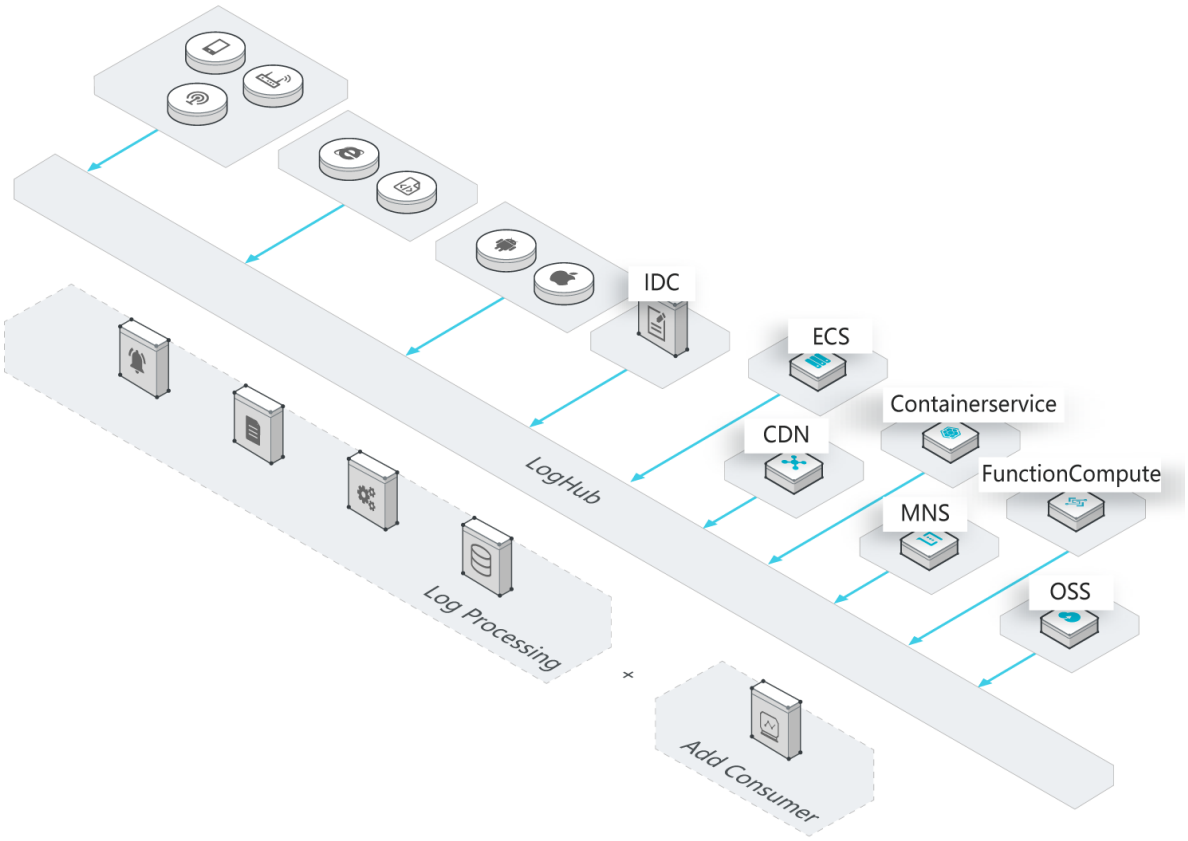
数据采集与消费

通过日志服务LogHub功能，可以大规模低成本接入各种实时日志数据（包括Metric、Event、BinLog、TextLog、Click等）。

方案优势：

- 使用便捷：提供30+实时数据采集方式，让您快速搭建平台；强大配置管理能力，减轻运维负担。
- 弹性伸缩：无论是流量高峰还是业务增长都能轻松应对。

图 4-1: 数据采集与消费

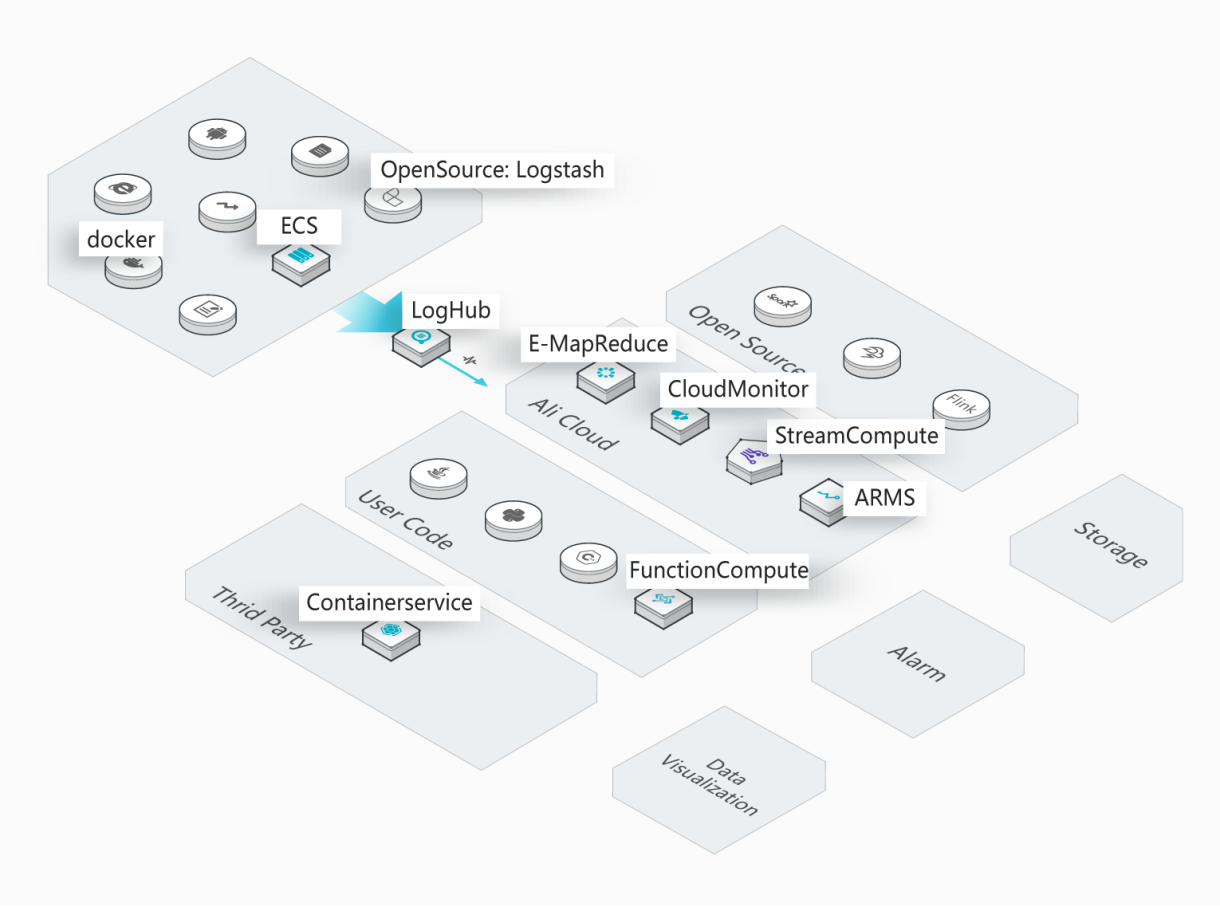


数据清洗与流计算 (ETL/Stream Processing)

日志中枢（LogHub）支持与各种实时计算及服务对接，并提供完整的进度监控，报警等功能，并可以根据SDK/API实现自定义消费。

- 操作便捷：提供丰富SDK以及编程框架，与各流计算引擎无缝对接。
- 功能完善：提供丰富监控数据，以及延迟报警机制。
- 弹性伸缩：PB级弹性能力，0延迟。

图 4-2: 数据清洗与流计算

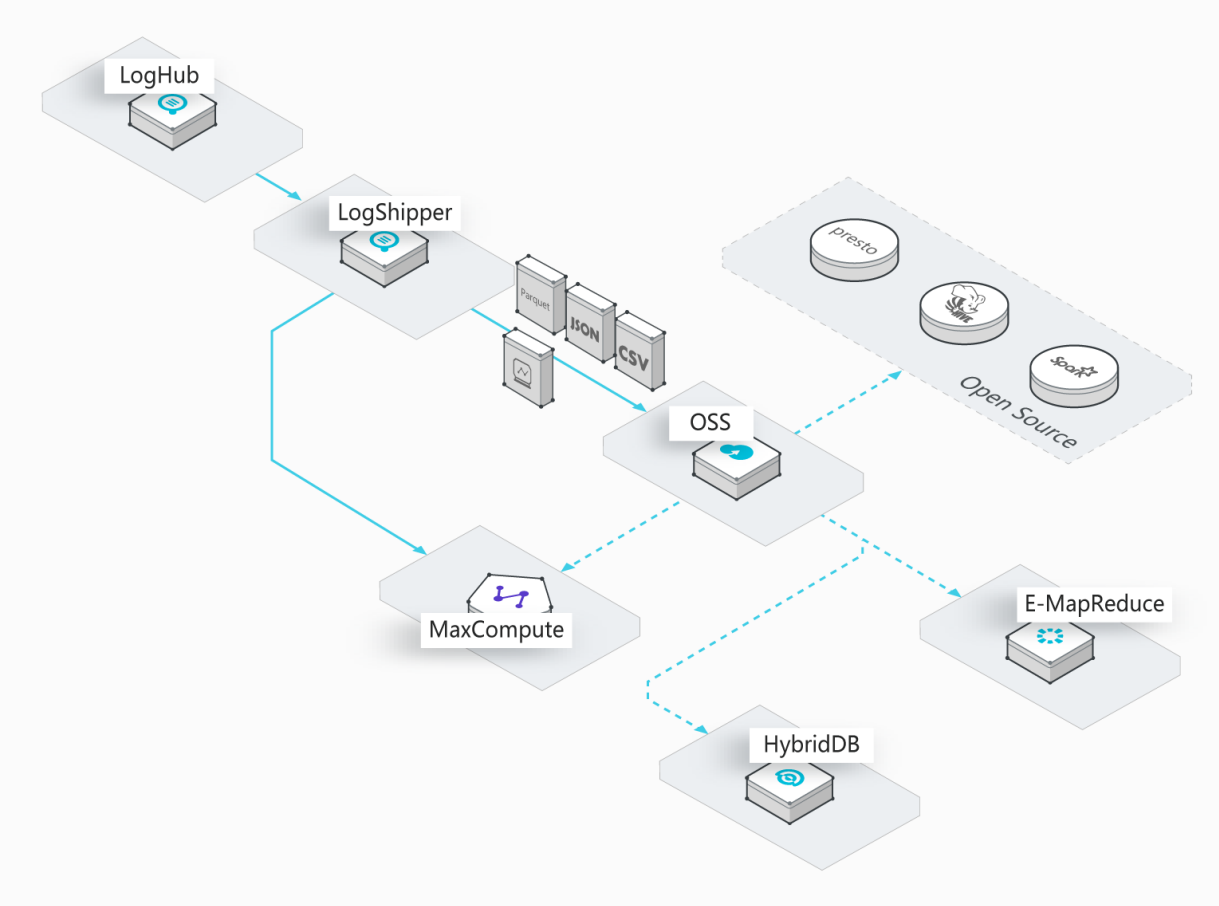


数据仓库对接(Data Warehouse)

日志投递（LogShipper）功能可以将日志中枢（LogHub）中数据投递至存储类服务，过程支持压缩、自定义Partition、以及行列等各种存储格式。

- 海量数据：对数据量不设上限。
- 种类丰富：支持行、列、TextFile等各种存储格式。
- 配置灵活：支持用户自定义Partition等配置。

图 **4-3:** 数据仓库对接

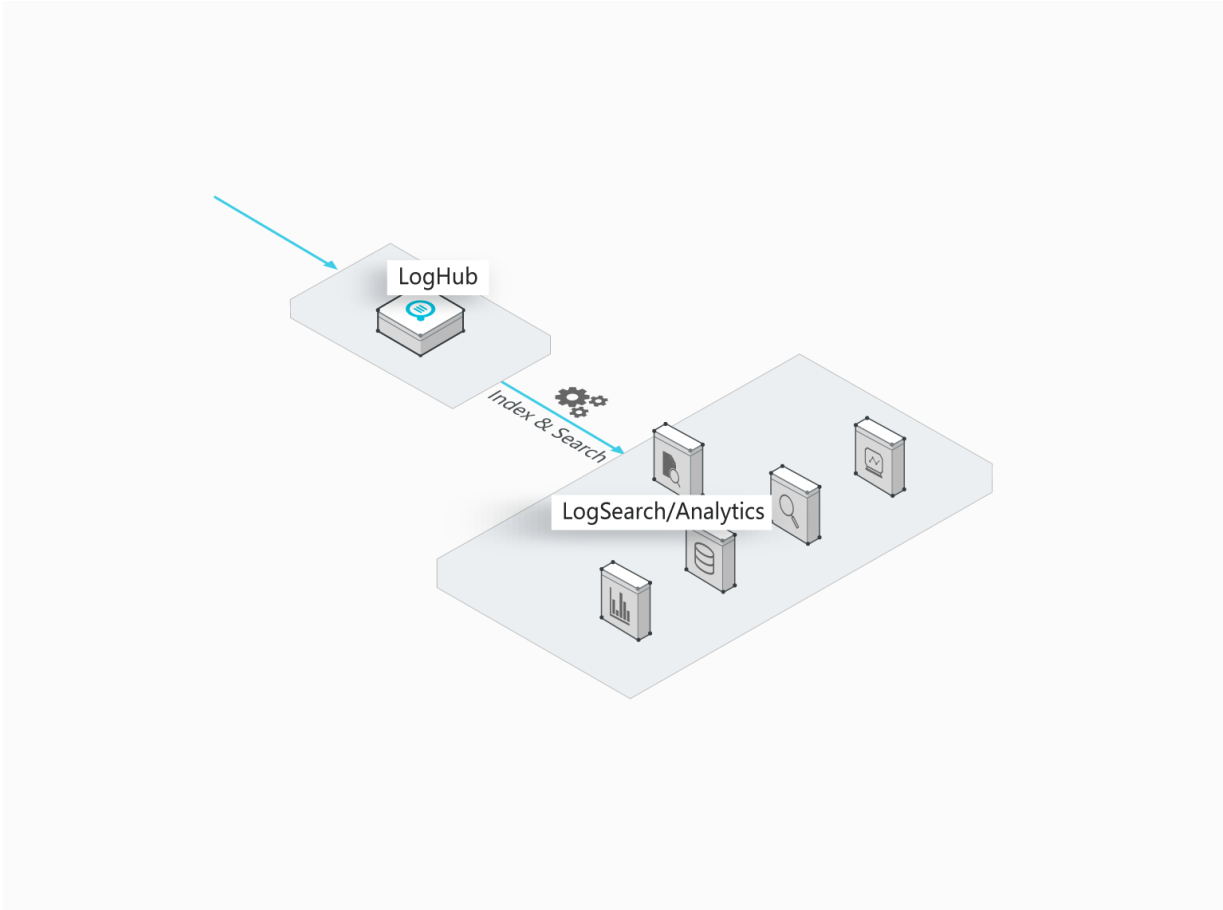


日志实时查询与分析

实时查询分析（LogAnalytics）可以实时索引LogHub中数据，提供关键词、模糊、上下文、范围、SQL聚合等丰富查询手段。

- 实时性强：写入后即可查询。
- 海量低成本：支持PB/Day索引能力，成本为自建方案15%。
- 分析能力强：支持多种查询手段，及SQL进行聚合分析，并提供可视化及报警功能。

图 4-4: 日志实时查询与分析



5 限制说明

5.1 基础资源

分类	限制说明	备注
Project	每个账号下最多可创建50个Project。	如您有更大的使用需求，请提工单申请。
Logstore	一个Project中最多可创建200个Logstore。	如您有更大的使用需求，请提工单申请。
Shard	• 一个Project中最多可创建200个Shard。 • 控制台创建Logstore时，一个Logstore最多可创建10个Shard；API创建Logstore时，最多可创建100个。但可以通过分裂操作来增加Shard。	如您有更大的使用需求，请提工单申请。
Logtail配置（LogtailConfig）	每个Project最多可创建100个Logtail配置。	如您有更大的使用需求，请提工单申请。
日志保存时间	支持永久保存。 您也可以自定义日志保存时间，取值范围为1~3000。	-
机器组（MachineGroup）	每个Project最多可创建100个机器组。	如您有更大的使用需求，请提工单申请。
协同消费组（ConsumerGroup）	每个Logstore最多可创建10个协同消费组。	可以删除不使用消费组。
快速查询（SavedSearch）	每个Project最多可创建100个快速查询。	-
仪表盘（Dashboard）	• 每个Project最多可创建50个仪表盘。 • 每个仪表盘最多可包含50张分析图表。	-

分类	限制说明	备注
LogItem	长度最大为1 MB。	以上为API限制参数，如通过Logtail采集日志，单个LogItem最大为 512KB。
LogItem（Key）	长度最大为128 Bytes。	-
LogItem（Value）	长度最大为1 MB。	-
日志组（LogGroup）	每个日志组中最多包含4096条日志，且最大长度为5 MB。	-

说明：

通过日志服务命令行工具CLI可以查看当前的Logstore、Shard、仪表盘等基础资源使用情况，详细说明请查看使用*CLI*查看基础资源使用状况。

5.2 数据读写

分类	限制项	限制说明	备注
Project	写入流量保护	写入流量最大为30 GB/min。	如超过限制，状态码会返回403，提示Inflow Quota Exceed。如您有更大的使用需求，请提工单申请。
	写入次数保护	写入次数最大为600000 次/min。	如超过限制，状态码会返回403，提示Write QPS Exceed。如您有更大的使用需求，请提工单申请。
	读取次数保护	读取次数最大为600000 次/min。	如超过限制，状态码会返回403，提示Read QPS Exceed。如您有更大的使用需求，请提工单申请。
Shard	写入流量	写入流量最大为5 MB/s。	非硬性限制，超过时系统会尽可能服务，但不保证服务质量。

分类	限制项	限制说明	备注
	写入次数	写入次数最大为500次/s。	非硬性限制，超过时系统会尽可能服务，但不保证服务质量。
	读取流量	读取流量最大为10 MB/s。	非硬性限制，超过时系统会尽可能服务，但不保证服务质量。
	读取次数	读取次数最大为100次/s。	非硬性限制，超过时系统会尽可能服务，但不保证服务质量。

5.3 查询分析与可视化

分类	分类	限制说明	备注
查询（Search）	关键词个数	关键词，即单词查询时布尔逻辑符外的条件个数。每次查询最多30个。	例如"a and b or c and d..."。
	单个字段长度	单个字段（Value）长度最大为10 KB，超出部分不参与查询。	如果单个字段长度大于10 KB，有一定几率无法通过关键词查询到日志，但数据仍然是完整的。
	单个Project并发	单个Project并发最大为100个。	-
	返回的查询结果条数	每次查询时，默认最多返回100条查询结果。	可以通过翻页读取完整的查询结果。
	单条日志内容显示	由于网页浏览器性能原因，对于超过1w个字符的日志，日志服务只会对前10,000个字符进行DOM切词处理。	如果超出10,000个字符，控制台会提示“该日志存在超过10000个字符的日志数据，部分显示上会有降级处理”。
SQL分析（Analytics）	单个字段（Value）最大长度	单个字段（Value）最大长度为2 KB，超出部分不参与查询。	超出限制时查询结果可能不精确，但数据仍然是完整的。
	单个Project并发	单个Project并发不超过30个。	-

分类	分类	限制说明	备注
	每次分析的返回结果	• 每次分析默认返回100条日志数据。 • 每次分析的返回结果最大100 MB或100,000条。	• 可以通过limit指定返回日志的条数。 • 可以通过翻页方式获取分批读取结果。
	字段长度	使用分析语句时，字段长度最大支持2 KB。	分析日志时，超出2 KB的字段会被截断。
	double精度	Double类型最多用52位表示，如果浮点数编码位数超过52位，会有精度损失。	例如，0.1+0.2会表示为0.30000000000000004。建议您使用round(0.1+0.2,2)来取两位有效数值。

5.4 保留字段

日志服务中有部分字段为保留字段，使用API写入数据，或添加Logtail采集配置时，请不要将字段名称设置为日志服务的保留字段。

在采集日志或投递数据到其他云产品时，日志服务可以将日志来源、时间戳等信息以Key-Value对的形式添加到日志中，其中字段名称为`__source__`等固定名称，这些字段是日志服务的保留字段。使用API写入日志数据或添加Logtail配置时，请不要将Key即字段名称设置为这些保留字段，否则可能会造成字段名称重复、查询不精确等问题。

目前，日志服务的保留字段包括：

表 5-1: 日志服务保留字段

保留字段名称	数据格式	索引与统计设置	说明
`__time__`	整型，Unix标准时间格式。	• 索引设置：`__time__`通过API中的参数from和to选择，无需添加该字段的索引。 • 统计设置：当用户为任何一列开启统计后，默认为`__time__`开启统计。	使用API/SDK写入日志数据时指定的日志时间，该字段可用于日志投递、查询、分析。
`__source__`	字符串格式。	• 索引设置：开启索引后，日志服务默认为`__source__`创建索引，索引数据类	日志来源设备。该字段可用于日志投递、查询、分析、自定义消费。

保留字段名称	数据格式	索引与统计设置	说明
		型为text类型，分词字符为空。查询时输入 `source:127.0.0.1` 或者 `__source__:127.0.0.1` 。 • 统计设置：当用户为任何一列开启统计后，默认为 `__source__` 开启统计。	
`__topic__`	字符串格式。	• 索引设置：开启索引后，日志服务默认为 `__topic__` 创建索引，索引数据类型为text类型，分词字符为空。查询时输入 `__topic__:XXX` 。 • 统计设置：当用户为任何一列开启统计后，默认为 `__topic__` 开启统计。	日志主题（Topic）。如果您设置了日志主题，日志服务会自动为您的日志添加日志主题字段，Key为 `__topic__` ，Value为您的主题内容。该字段可用于日志投递、查询、分析、自定义消费。
`__partition_time__`	字符串格式。	日志内容中不存在该字段，无需设置索引。	投递MaxCompute的日志分区时间列。由 `__time__` 计算得到，用于日志投递MaxCompute时设置日期格式分区列，详细说明请参考通过日志服务投递日志到*MaxCompute*。
`__extract_others__`	字符串，可反序列化成JSON Map。	日志内容中不存在该字段，无需设置索引。	日志中投递MaxCompute的未配置字段组装为一个JSON Map。用于日志投递MaxCompute时打包其它未单独配置的字段，详细说明请参考通过日志服务投递日志到*MaxCompute*。
`_extract_others_`	字符串，可反序列化成JSON Map。	日志内容中不存在该字段，无需设置索引。	与 `__extract_others__` 相同，建议使用 `__extract_others__` 。

保留字段名称	数据格式	索引与统计设置	说明
__tag__:__client_ip__	字符串格式。	• 索引设置：开启索引后，日志服务默认为所有标签#Tag#字段创建索引，索引数据类型为text类型，分词字符为空，在查询时要完全命中，或采用模糊查询。 • 统计设置：默认没有为该列开启统计。如需开启统计，请手动添加__tag__:__client_ip__的索引，并开启统计功能。	日志来源设备的公网IP，该字段为系统标签（Tag）。开启记录外网IP功能后，服务端接收日志时为原始日志追加该字段。可用于日志投递、查询、分析、自定义消费。
__tag__:__receive_time__	字符串，可转换为整型的Unix标准时间格式。	• 索引设置：开启索引后，日志服务默认为所有标签#Tag#创建索引，索引数据类型为text类型，分词字符为空，在查询时要完全命中，或采用模糊查询。 • 统计设置：默认没有为该列开启统计。如需开启统计，请手动添加__tag__:__receive_time__的索引，并开启统计功能。	日志到达服务端的时间，该字段为系统标签#Tag#。开启记录外网IP功能后，服务端接收日志时为原始日志追加该字段。可用于日志投递、查询、分析、自定义消费。
__tag__:__path__	字符串格式。	• 索引设置：开启索引后，日志服务默认为__tag__:__path__创建索引，索引数据类型为text类型，分词字符为空。查询时输入__tag__:__path__:XXX。 • 统计设置：当用户为任何一列开启统计后，默认为__tag__:__path__开启统计。	Logtail采集的日志文件路径，Logtail为日志自动填加该字段。
__tag__:__hostname__	字符串格式。	• 索引设置：开启索引后，日志服务默认为__tag__:__hostname__创建索	logtail采集数据的来源机器主机名，Logtail为日志自动填加该字段。

保留字段名称	数据格式	索引与统计设置	说明
		引，索引数据类型为text类型，分词字符为空。 查询时输入`__tag__:__hostname__:XXX`。 • 统计设置：当用户为任何一列开启统计后，默认为`__tag__:__hostname__`开启统计。	
`__raw_log__`	字符串格式。	请手动添加并设置该字段的索引，索引数据类型为text，并根据需求选择是否开启统计。	解析失败的原始日志。关闭丢弃解析失败日志功能后，Logtail在解析日志失败时上传原始日志。其中Key为`__raw_log__`、Value为日志内容。
`__raw__`	字符串格式。	请手动添加并设置该字段的索引，索引数据类型为text，并根据需求选择是否开启统计。	解析成功的原始日志。开启上传原始日志功能后，Logtail会将原始日志作为`__raw__`字段，和解析后的日志一并上传。一般用于审计、合规审查等场景。