

刘清华

博客园

首页

新随笔

联系

订阅

管理

公告

昵称： Liutaohua  
园龄： 6个月  
粉丝： 1  
关注： 0  
+加关注

<	2020年6月						>
日	一	二	三	四	五	六	
31	1	2	3	4	5	6	
7	8	9	10	11	12	13	
14	15	16	17	18	19	20	
21	22	23	24	25	26	27	
28	29	30	1	2	3	4	
5	6	7	8	9	10	11	

搜索

找找看

谷歌搜索

常用链接

我的随笔  
我的评论  
我的参与  
最新评论  
我的标签

我的标签

IoTDB(5)  
数据库(5)  
时序数据(4)  
物联网(4)  
行式数据库(3)  
TsFile(3)  
车联网(3)  
列式数据库(3)  
InfluxDB(2)  
数据库性能(2)

随笔档案

2020年2月(5)

最新评论

1. Re:时序数据库 Apache-IoTDB 源码解析之文件数据块（四）  
楼主您好，我刚接触这个数据库一天 安装的时候有个问题卡住了 能加个微信吗。指点一下  
--Smile\_灰太狼

2. Re:时序数据库 Apache-IoTDB 源码解析之文件数据块（四）  
@daconglee 目前没有...  
--Liutaohua

3. Re:时序数据库 Apache-IoTDB 源码解析之文件数据块（四）  
有C#的读写代码吗？  
--daconglee

随笔 - 5 文章 - 0 评论 - 4

时序数据库 Apache-IoTDB 源码解析之前言（一）

IoTDB 是一款时序数据库，相关竞品有 Kairosdb, InfluxDB, TimescaleDB等，主要使用场景是在物联网相关行业，如：车联网、风力发电、地铁、飞机监控等等，具体应用案例及公司详情可以查看：[IoTDB在实际公司中的使用信息收集](#)

IoTDB 模块主要分为Client, JDBC, Server, TsFile, Grafana, Distribution 以及各种生态的连接器。整个系列的文章会从行业背景开始讲起，了解一个行业具体的使用场景，然后介绍 TsFile 是以什么样的格式来保存数据的，再介绍 Server 里怎样完成一次查询，最后在介绍一条完整的 SQL是怎样从 Client 使用 JDBC 到 Server 直至返回具体结果。如果有能力的话再介绍一下集群的一些内容和工作方式。

打一波广告本人专注车联网领域多年，现任四维智联架构师。目前正在参与 IoTDB 社区，志同道合的同伴欢迎加微信：liutaohua001

欢迎大家访问 [IoTDB 仓库](#)，求一波 Star 。

这一章主要想聊一聊：

- 1. 为什么重复造轮子，从物联网行业的数据特点到 IoTDB 的发展过程
- 2. 这个轮子造的怎么样，IoTDB 和竞品测试对比

时序数据

我个人理解时序数据是基于时间维度的同一个物体或概念的值构成的一个序列数据。在传统关系型数据库中，例如 MySQL，我们通常会放置一个自增的 Id 列作为主键标识，如下：

Id	人名	体温	测量时间
1	张三	36.5	2020-02-06 9:00:00
2	李四	36.9	2020-02-06 9:00:00
3	王五	36.7	2020-02-06 9:00:00
4	张三	36.3	2020-02-06 9:30:00
5	张三	36.9	2020-02-06 11:00:00

上面的表结构就是一个时序数据，将表结构做个变形更容易理解：

时间戳	人名	体温
1580950800	张三	36.5
1580950800	李四	36.9
1580950800	王五	36.7
1580952600	张三	36.3
1580958000	张三	36.9

如果把时间作为一个唯一键对齐展示，能够更像时序数据一些，这也是 IoTDB 中查询结果的展示方式：

时间戳	张三	李四	王五
1580950800	36.5	36.9	36.7
1580952600	36.3	NULL	NULL
1580958000	36.9	NULL	NULL

4. Re:时序数据库 Apache-IoTDB 源码解析之前言（一）  
有C#的读写代码吗？

--daconglee

阅读排行榜

- 1. 时序数据库 Apache-IoTDB 源码解析之文件索引块（五）(312)
- 2. 时序数据库 Apache-IoTDB 源码解析之系统架构（二）(227)
- 3. 时序数据库 Apache-IoTDB 源码解析之前言（一）(200)
- 4. 时序数据库 Apache-IoTDB 源码解析之文件数据块（四）(152)
- 5. 时序数据库 Apache-IoTDB 源码解析之文件格式简介（三）(125)

评论排行榜

- 1. 时序数据库 Apache-IoTDB 源码解析之文件数据块（四）(3)
- 2. 时序数据库 Apache-IoTDB 源码解析之前言（一）(1)

这里可能会存在疑问就是假如人数是逐渐增加的，那么是动态创建列呢？还是提前创建足够多的列？这个问题等后面文章有机会继续介绍

物联网

物联网的特点是都会存在一个或多个设备，他们以各种各样的形式组织到一起，用来观测或记录同一时间里相同环境所产生的数据。下面的介绍中，使用由简单到复杂的数据逐步介绍在物联网行业中，通用的一些问题和方向。

1.基本存储

假如我是一个公司，对外播报北京、天津、上海三地的温度数据，从而实现盈利。

时间戳	北京	天津	上海
1580950800	20.5	22.9	21.7
1580952600	20.5	NULL	22.9
1580958000	20.5	21.7	22.9

2.保证数据质量

数据保证的质量是多方面的，一步一步介绍。

2.1 更多设备

首先可以看到上面数据是存在 NULL 值的，这个 NULL 值有可能是因为当时设备所在的区域停电了，所以并没有办法上报当时的情况，这样客户如果想获取1580952600 这个时间戳对应的天津的数据的时候，肯定是拿不到了，所以传统思维上，我们应该增加一个容灾设备，保证一个设备在坏掉、停电、人为损坏等等的情况的时候，依然能够有数据上报回来。

基于这样的思想，以上的表结构就会变成：

时间戳	北京 1	北京 2	天津1	天津2	上海1	上海2
1580950800	20.5	20.9	21.7	20.9	20.7	21.7
1580952600	21.5	21.0	NULL	21.7	21.7	21.7
1580958000	22.5	22.7	22.9	22.7	NULL	NULL

2.2 更高采样频率

这时候依然存在问题， 1580958000 这一刻两个设备都没有数据，有可能是放置设备的区域同时出现了断网或者断电，这种情况下，我们可以采用提高采集数据的频率或者补传数据来解决(补传暂不讨论)。

我们将每天数据分为3组，每组采样3次，间隔为1个小时，假如时间分布为：上午（7、8、9）、中午（12、13、14）、下午（18、19、20）。当增加了采样频率之后，即便某一刻出现了 NULL 数据，我们也可以采用临近时间做为补充。为了方便对应，下表数据中增加时间点列辅助查看。

时间点	时间戳	北京 1	北京 2	天津1	天津2	上海1	上海2
7点	1580943600	20.5	20.9	21.7	20.9	20.7	21.7
8点	1580947200	21.5	21.0	NULL	21.7	21.7	21.7
9点	1580950800	22.5	22.7	22.9	22.7	NULL	NULL
12点	1580961600	20.5	20.9	21.7	20.9	20.7	21.7
13点	1580965200	21.5	21.0	NULL	21.7	21.7	21.7
14点	1580968800	22.5	22.7	22.9	22.7	NULL	NULL
18点	1580983200	20.5	20.9	21.7	20.9	20.7	21.7
19点	1580986800	21.5	21.0	NULL	21.7	21.7	21.7

20点	1580990400	22.5	22.7	22.9	22.7	NULL	NULL
-----	------------	------	------	------	------	------	------

可以看到经过各种各样的需求之后，上传的数据是成倍增长的，不难想象如果这个温度数据希望精准的获取到每个县城的温度，那么中国有 2854 个县城 \* 2 个温度设备 \* 9 条数据 = 1 天产生的数据总量 = 51372 条，那么一个月就是 1541160 条。

数据实时性及总量

假如上面的数据我们继续提高频率到每1分钟每个设备上报一次，那么数据量就会成为 2854 \* 2 \* 60 \* 24 = 246585600 条/天。

在这样的数据量下，实时插入实时做一些聚合计算，应该传统数据库就有些处理不过来了。

IoTDB 的前身

某公司在实际业务中，20 万设备保存了 3 年的数据，TB级别的数据使得 Oracle 被拖的根本吃不消。关键的问题点还不仅仅是存量数据大，新增数据依然以非常快的速度在增长。后来公司联系到了 IoTDB 的第一批开发者，但是当时的方案还是基于 Cassandra 来做设计，当时规划了 5 台机器的集群，性能刚满足，但随着时间的推移设备总量在增加，业务系统的查询请求量在增加。Cassandra 在经过大量的努力之后，最后发现如果再改可能需要大面积的重构 Cassandra 数据的代码了，最终决定重新设计一个存储方式，来解决物联网场景下的时序数据高效写入、低延迟读取、高压缩比持久化。

PS：以上都是黄向东 (IoTDB PPMC)，在 meetup 中讲到的，我只是在脑中存留了一部分，具体的细节大家可以到 IoTDB 社区交流。

性能对比

测试工具使用的是由清华大学大数据实验室开发的*iotdb-benchmark*

1.写入性能对比

数据集 2	客户端	存储组	设备	变量	batch size	LOOP	数据量	写入速度 (point /s)	硬盘写入速度 (MB/s)
IoTDB	10	10	10	10	1000	1000000	1E+11	24750321.93	3830092
InfluxDB	10	10	10	10	1000	1000000	1E+11		30462932
TimescaleDB	10	10	10	10	1000	1000000	1E+11	737689.22	16101906

数据集 1	客户端	存储组	设备	变量	batch size	LOOP	数据量	写入速度 (point /s)	硬盘写入速度 (MB/s)
IoTDB	10	10	10	10	1000	100000	1000000000	20706345.15	359932
InfluxDB	10	10	10	10	1000	100000	1000000000	1729907.81	3054560
TimescaleDB	10	10	10	10	1000	100000	1000000000	715857	16106468
Kairos							10000	24924.	7626

DB	10	10	10	10	10000	10000	000000	97	380
----	----	----	----	----	-------	-------	--------	----	-----

上面一组数据可以看出写入性能高于同款数据库10倍有余，单机写入速度高达每秒2千万。且硬盘占用是最小的，这在数据比较大的线上业务中，可能每个月会差出来 1 到 2 块硬盘。

## 2. 查询性能对比

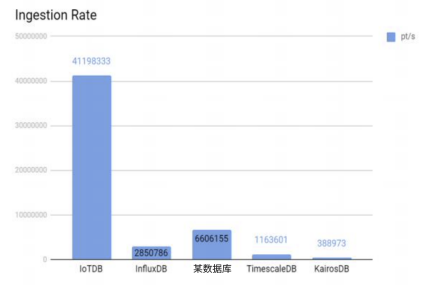
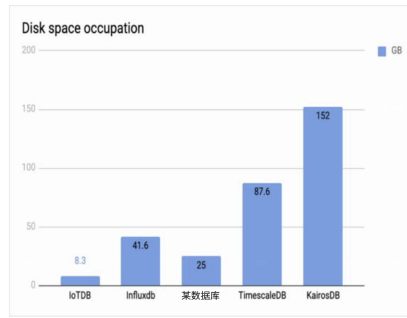
原始数据查询

	客户端	存储组	设备	序列-数据量	变量	查询点数	LOOP	速度 (point/s)	AVG
IoTDB	10	10	10	1.00E+09	1	1000000	100	12942984.85	740.1
InfluxDB	10	10	10	1.00E+09	1	1000000	100	1779606.4	5591
TimescaleDB	10	10	10	1.00E+09	1	1000000	100	3781467.52	23459

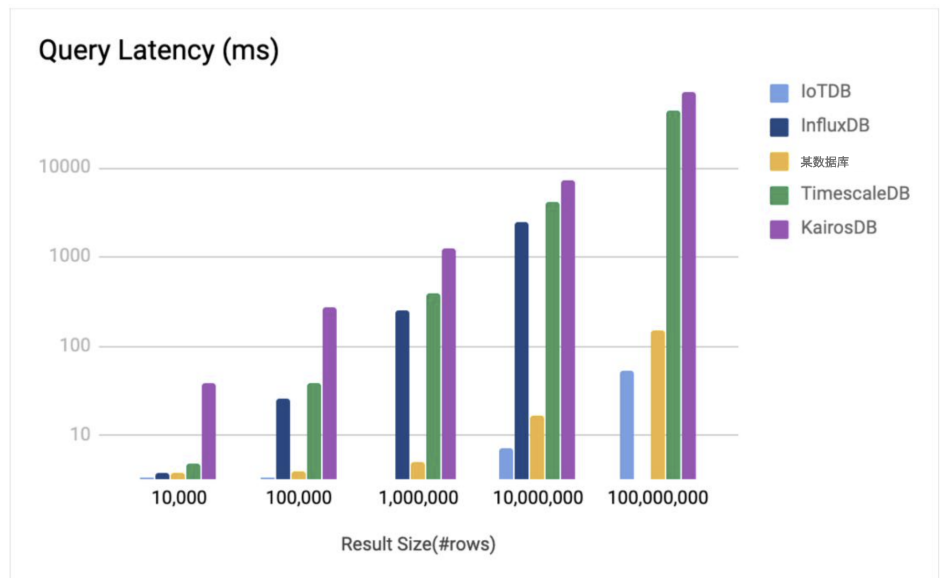
聚合数据查询

	客户端	存储组	设备	序列-数据量	变量	LOOP	范围	速度 (point/s)	AVG
IoTDB-1	10	10	10	1.00E+09	1	100	0.0001	49.75	27.8
IoTDB-2	10	10	10	1.00E+09	1	100	0.001	49.75	49.1
IoTDB-3	10	10	10	1.00E+09	1	100	0.01	49.76	48.6
IoTDB-4	10	10	10	1.00E+09	1	100	0.1	48.68	99.1
IoTDB-5	10	10	10	1.00E+09	1	100	1	14	595.1
InfluxDB-1	10	10	10	1.00E+09	1	100	0.0001	234.32	40.2
InfluxDB-2	10	10	10	1.00E+09	1	100	0.001	28.88	341.1
InfluxDB-3	10	10	10	1.00E+09	1	100	0.01	3.07	32267
TimescaleDB-1	10	10	10	1.00E+09	1	100	0.0001	42.39	220.1
TimescaleDB-2	10	10	10	1.00E+09	1	100	0.001	5.8	1502
TimescaleDB-3	10	10	10	1.00E+09	1	100	0.01	1.02	97115

### 3. 对比图



Raw data:  
 - 12 Bytes per point  
 - 112 GB totally



整体来看 IoTDB 无论在写入、原始数据查询还是聚合查询，都几乎是10倍的性能于竞品数据库，而且硬盘占用又小于同款数据库10倍，那么 IoTDB 是怎样完成如此高的压缩比、如此恐怖的写入速度、如此高效的查询呢？欢迎继续关注。。。

标签: IoTDB, 数据库, 物联网, 时序数据, InfluxDB, 数据库性能

好文要顶

关注我

收藏该文

Liutaohua  
 关注 - 0  
 粉丝 - 1  
[+加关注](#)

0 0

» 下一篇: 时序数据库 Apache-IoTDB 源码解析之系统架构（二）

posted @ 2020-02-08 12:37 Liutaohua 阅读(200) 评论(1) 编辑 收藏

评论列表

#1楼 2020-03-08 10:34 daconglee

有C#的读写代码吗？

支持(0) 反对(0)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问](#) 网站首页。

【推荐】超50万行VC++源码：大型组态工控、电力仿真CAD与GIS源码库

【推荐】SaaS模式云数据仓库MaxCompute百问合集

#### 相关博文：

- 时序数据库Apache-IoTDB源码解析之文件索引块（五）
- 时序数据库Apache-IoTDB源码解析之文件索引块（五）
- 时序数据库Apache-IoTDB源码解析之文件数据块（四）
- 时序数据库Apache-IoTDB源码解析之文件格式简介（三）
- 时序数据管理引擎ApacheIoTDB

» 更多推荐...

#### 最新 IT 新闻：

- GitHub 开源 Super Linter，用自动化解决开发者的需求
- 量子计算机领域内第一种高级编程语言 Silq 诞生
- 微软正式推出 gRPC-Web for .NET
- Visual Studio Code 6 月 Python 扩展更新
- 再见 Python，你好 Julia!

» 更多新闻...