www.vip.com

Redis在大数据业务中的应用

唯品会
vip.com
一家专门做特卖的网站

陈群@唯品会/数据基础架构/DBA

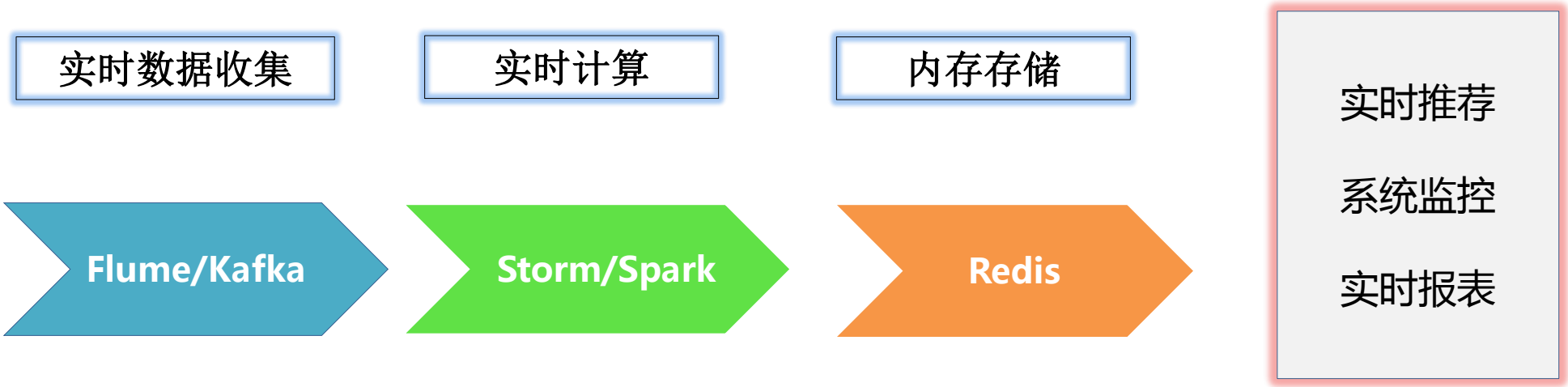
Agenda

1. Redis In Bigdata Ecosystem
2. Storage Architecture Evolution
3. Redis Cluster In Practice

计算和存储


计算
Map-Reduce Job
Storm
Spark
Java/C++/Python
存储
HDFS
HBase
Hive
Kafka
MySQL
Redis/Twemproxy/Cluster
MongoDB

实时数据


实时数据收集
实时计算
内存存储
Flume/Kafka
Storm/Spark
Redis
实时推荐
系统监控
实时报表

离线数据

1) Hive --> Redis，定期同步全量/增量数据

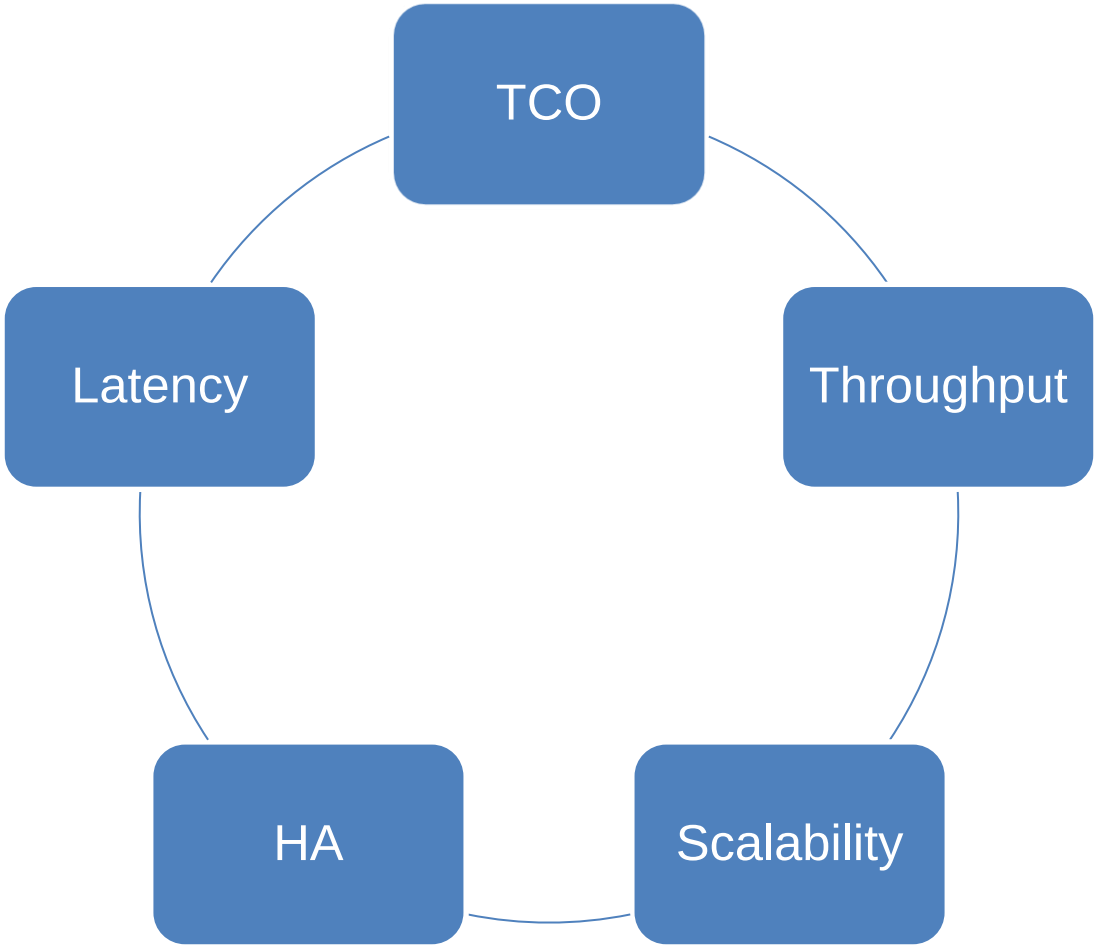
- 开发成通过用的hive2redis组件，支持twemproxy/redis-cluster。
- 提交配置即可完成数据推送，下面是一些常用参数。

> *arg1: data_type*
>
> *arg2: sql_query (hive进行数据提取的查询语句)*
>
> *arg3: redis_key_expire_seconds (redis的key的过期时间)*
>
> *arg4: redis_server_type (指定redis集群的架构, cluster or twemproxy)*

2) Java/C++/Python应用程序

- 数据源主要是MySQL、文件、Kafka等。

大数据对redis的挑战



Redis Features

- 全内存KV存储（NoSQL）
- 支持多种数据结构
- 单线程工作模型，Networking和IO异步方式
- 数据持久化
- 支持主从复制

Redis优点

1、Redis快速的内存读写，能够满足Storm/Spark的高速计算处理

2、丰富的数据结构和灵活的操作，简化开发复杂度

- *计数器，PV/UV等统计*
- *Hash，字典型数据。比如，user/goods/brand profile/tag；userid-mid设备映射。*
- *Set，数据去重。*
- *Zset，在推荐计算中方便Scoring/Ranking.*
- *HyperLogLog，统计UV/PV，有误差。*

3、TTL数据生命周期管理，数据自动完成删除

4、Schema-less，适合频繁的业务迭代

Storm/Spark and Redis

Storm Redis Integration

Storm/Trident integration for Redis

Storm-redis uses Jedis for Redis client.

Usage

How do I use it?

use it as a maven dependency:

```
<dependency>
    <groupId>org.apache.storm</groupId>
    <artifactId>storm-redis</artifactId>
    <version>${storm.version}</version>
    <type>jar</type>
</dependency>
```

- RedisLookupBolt : retrieves value from Redis using key
- RedisStoreBolt : stores key / value to Redis.
- One tuple will be matched to one key / value pair, and you can define match pattern to TupleMapper.

【相关资料】：*http://storm.apache.org/releases/1.0.1/storm-redis.html*

Storm/Spark and Redis

Spark and Redis

Spark, a general engine for large-scale data processing delivers significant advantages over using Hadoop MapReduce because of its cyclic data flow and use of in-memory computing. Redis with its blazing fast performance and optimized in-memory data structures reduces Spark processing time by up to 98%.

Advantages of Using Redis with Spark

- Redis can accelerate Spark performance by upto 50 times, in several use cases such as spark-timeseries
- Redis provides the shared distributed memory infrastructure for Spark
- Redis data structures allow individual elements of data to be accessed, minimizing serialization/deserialization overhead and avoiding having to transfer large chunks of data.
- Redis can run on flash memory used as a RAM extender, enabling very high performance, at a fraction of the cost

单线程 VS 性能

1、请求越简单，响应越快。

Get/Set（kv长度20字节）可以达到8w+ QPS。

2、Slow query影响服务的稳定性。

- 请求阻塞很长时间，大量请求超时
- 比如常见的O(N)操作，hmget/lrang/keys等
- 管理操作命令，bgrewriteaof/bgsave时fork子线程时造成短暂的写入阻塞
- 隐藏的操作，big-key expired

3）吞吐量受限

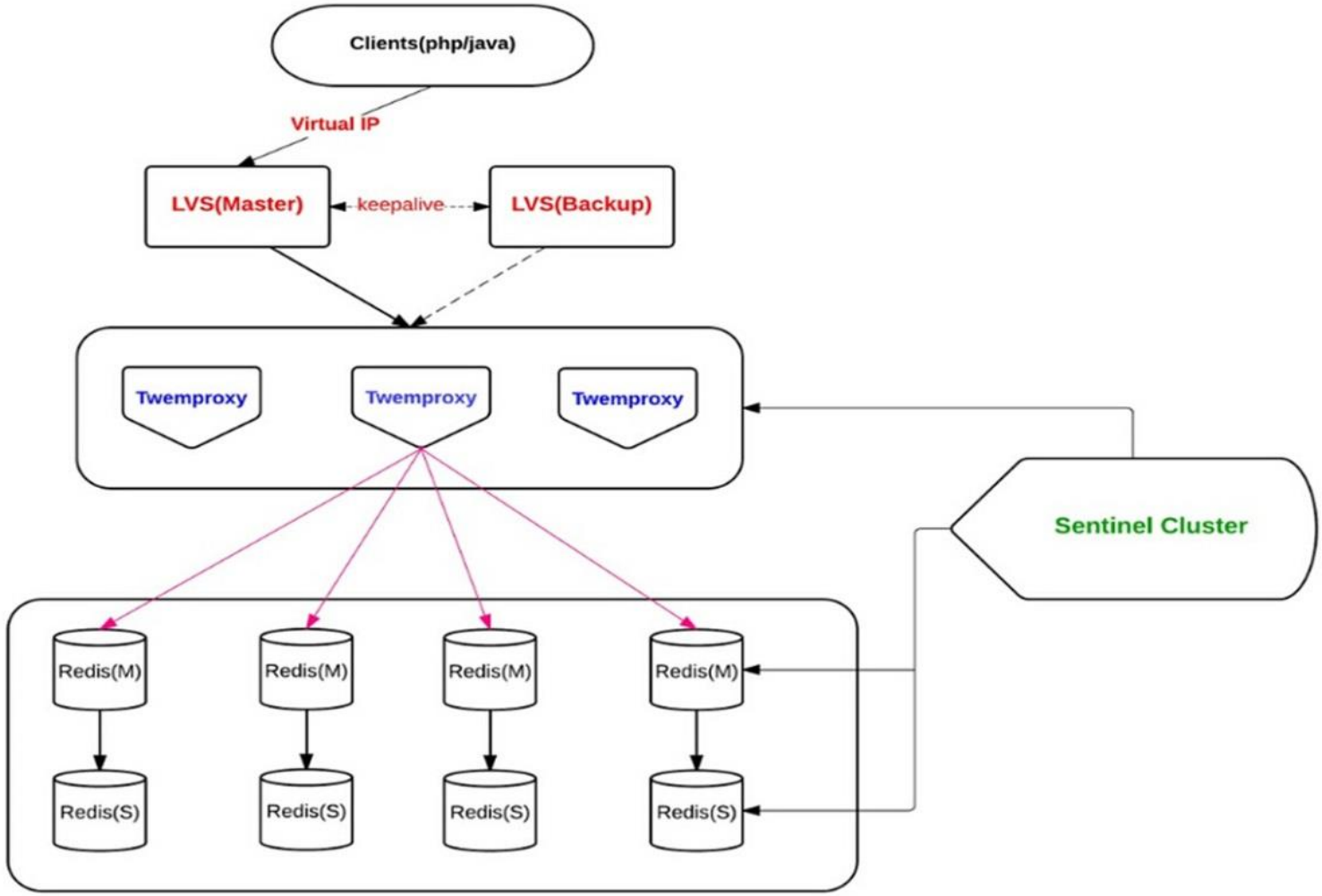
- 太高的并发请求反而导致qps下降
- 不适合存储长字符串，比如json/protocol buffer数据

Storage Architecture Evolution

分布式存储架构


Client Sharding
(2014.7之前)
Twemproxy
2015.12不再上线
Redis Cluster

Twemproxy


Clients(php/java)
Virtual IP
LVS(Master)
keepalive
LVS(Backup)
Twemproxy
Twemproxy
Twemproxy
Sentinel Cluster
Redis(M)
Redis(M)
Redis(M)
Redis(M)
Redis(S)
Redis(S)
Redis(S)
Redis(S)

Twemproxy

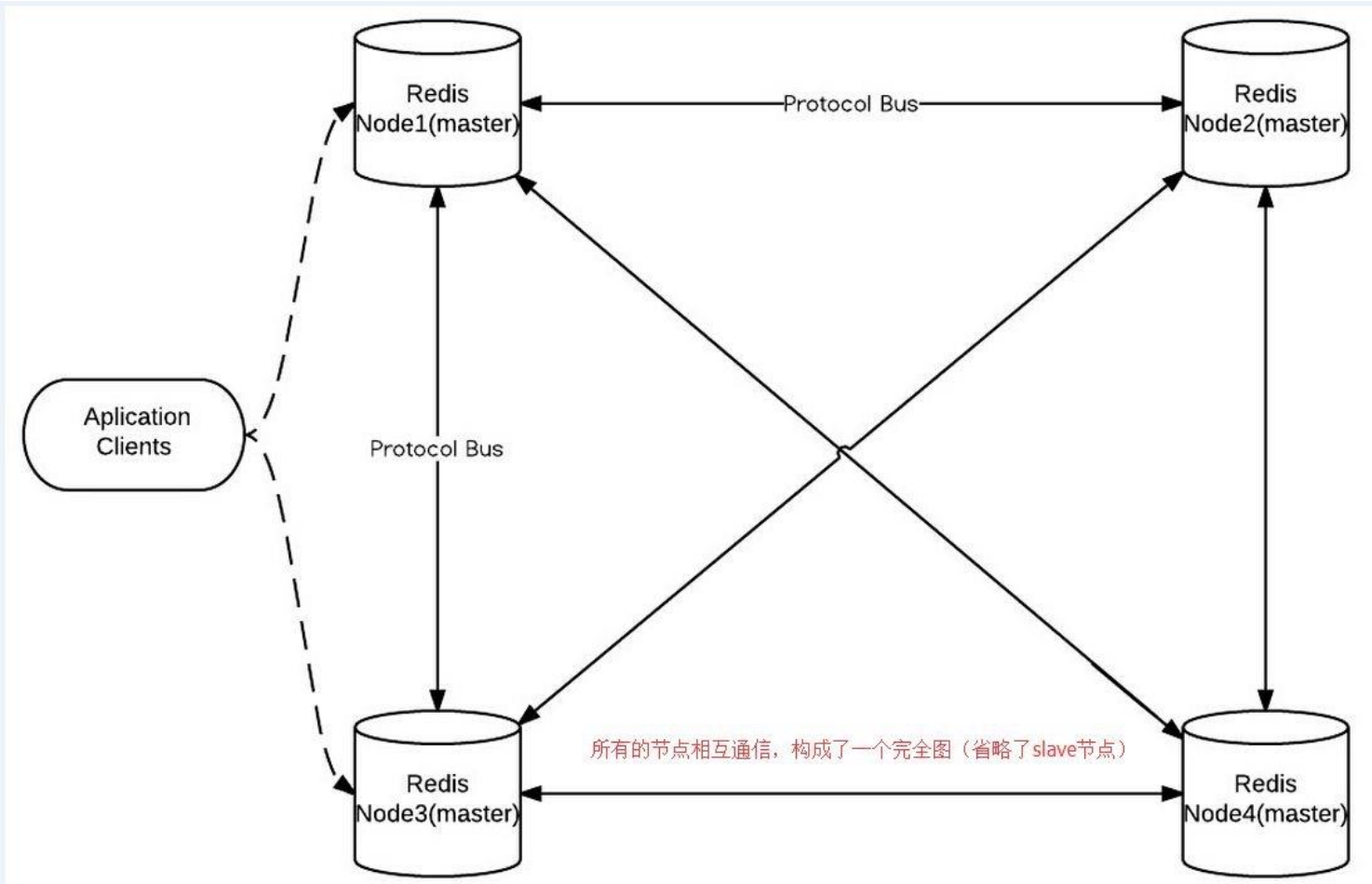
优点

- sharding逻辑对开发透明，读写方式和单个redis一致
- 可以作为cache和storage的proxy（by auto-eject）

缺点

- 架构复杂，层次多。包括lvs、twemproxy、redis、sentinel和其控制层程序
- 管理成本和硬件成本很高
- 2 * 1Gbps 网卡的lvs机器，最大能支撑140万pps
- 流量高的系统，proxy节点数和redis个数接近
- Redis层仍然扩容能力差，预分配足够的redis实例

Redis Cluster



Redis Cluster

优点

- 无中心架构
- 数据按照slot存储分布在多个redis实例上
- 增加slave做standby数据副本，用于failover，集群快速恢复
- 实现故障auto failover
- 亦可manual failover，为升级和迁移提供可操作方案
- 降低硬件成本和运维成本，提高系统的扩展性和可用性

缺点

- 重度依赖smart client
- 数据迁移比较慢，redis-3.0.6开始单次操作可以完成多个key的迁移

方案对比

	Client Sharding	Twemproxy	Redis Cluster
架构	• 后端仅多个独立的实例 • 主从复制 • 客户端实现分片	• Proxy完成分片 • 仅转发请求 • 部署在应用服务器 • proxy层加负载均衡，简化开发 • Lvs + twemproxy + redis三层架构	✓ 无中心架构 ✓ server端实现分片(crc32) • 主从复制 □ 强依赖smart client
兼容性	• 跨实例操作依赖客户端实现	• 兼容大部分redis命令 • Proxy层支持部分跨实例操作 • 支持mget/mset	• 跨实例操作依赖客户端实现
成本	✓ 低	□ 层次多 □ 机器数量多 □ 管理和维护成本高	✓ 低
扩展性	□ 扩展性差 □ 修改配置和代码，发布变更 • 简单扩容方式：倍增节点	□ Redis无法扩展 □ Lvs层达到瓶颈后扩容难 ✓ Proxy层无状态，扩容方便 ✓ Redis到Twemproxy迁移快速	✓ 扩展能力强 ✓ 可以在线增加/缩容节点
HA	• Sentinel + Zookeeper • Sentinel + D N S	• 自己开发Redis层HA模块	• slave 热备 ✓ auto-failover

为什么选择Redis Cluster

1、架构简单

- 无中心架构，各个节点对等
- slave节点提供数据冗余，master节点异常时提升为master
- 取代twemproxy三层架构，系统复杂性降低
- 可以节约大量的硬件资源
- 少了lvs和twemproxy层，读写性能提升明显。响应时间从100-200us减少到50-100us
- 相比twemproxy集群，系统瓶颈更少。

为什么选择Redis Cluster

2、支持在线扩展

- 在线水平扩展能力，能够解决我们大量的扩容需求。
- Failover能力和高可用性。
- redis不保证主从数据强一致性，但后端业务能够容忍少量数据丢失(异步复制)。

Redis Cluster In Practice

最佳实践

1、保持稳定快速的响应请求速度

- 保持请求的简单性，避免使用O(N)操作
- 不做复杂计算操作，尽量把计算迁移到代码层。eg：set求并集/交集操作。
- 拒绝扫描大数据集，eg：lrange、hkeys操作。

2、提高请求效率

- 单实例/twemproxy集群使用pipeline
- 使用连接池/长连接，避免server反复创建连接对象。
- 异步请求（需要client lib支持）
- 本地sink请求，把多次请求减少为一次，比如incr/incrby操作。
- 去掉select db操作，key使用前缀，空间换时间。

最佳实践

3、应用做好容错机制

- 连接或者请求异常，进行连接retry和reconnect
- 重试时间应该大于cluster-node-time时间

4、优化连接池使用

- 避免server端维持大量的连接
- 合理的连接池大小
- 合理的心跳检测时间
- 快速释放使用完的连接
- 对于storm job，建议一个线程一个连接。

Q&A


vip-陈群
Pudong New District,Shanghai
Scan the QR code to add me on WeChat

唯品会
vip.com
一家专门做特卖的网站

Thanks !

www.vip.com