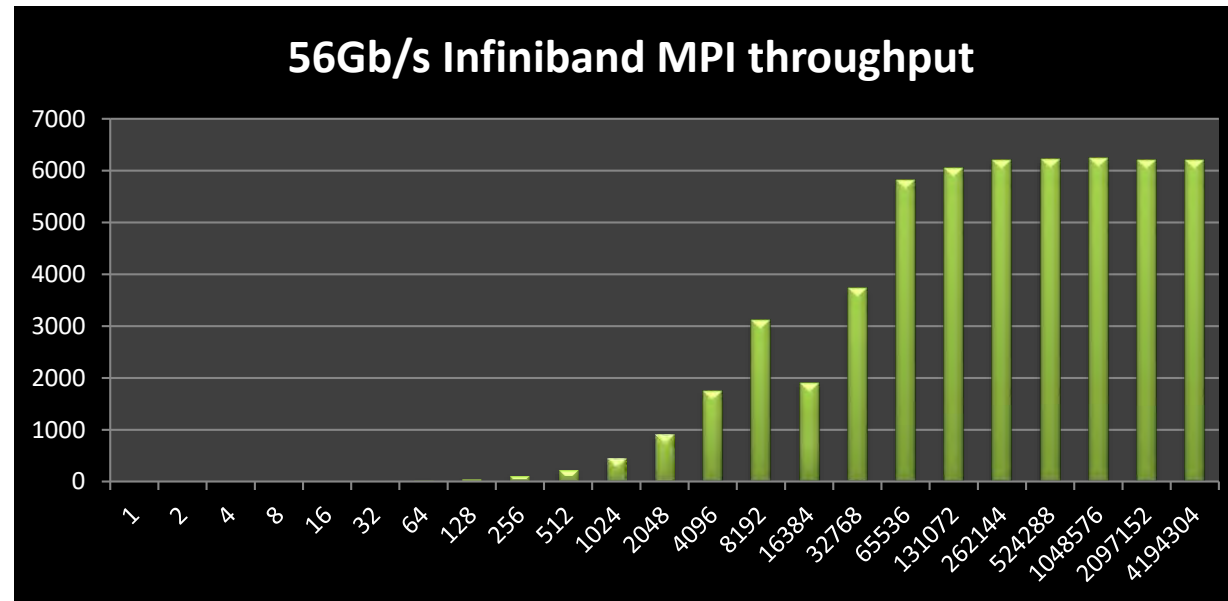


High-speed interconnects

□ Based on infiniband

- A protocol designed to meet high communication demand
- Low latency, high throughput (56 – 200 Gb/s)

□ Scale expansion by stacking switches



Versatile Storage Support

Different strategies for different data

Images



- Large number of small files
- require random access

Text



- Aggregate small text documents into large files

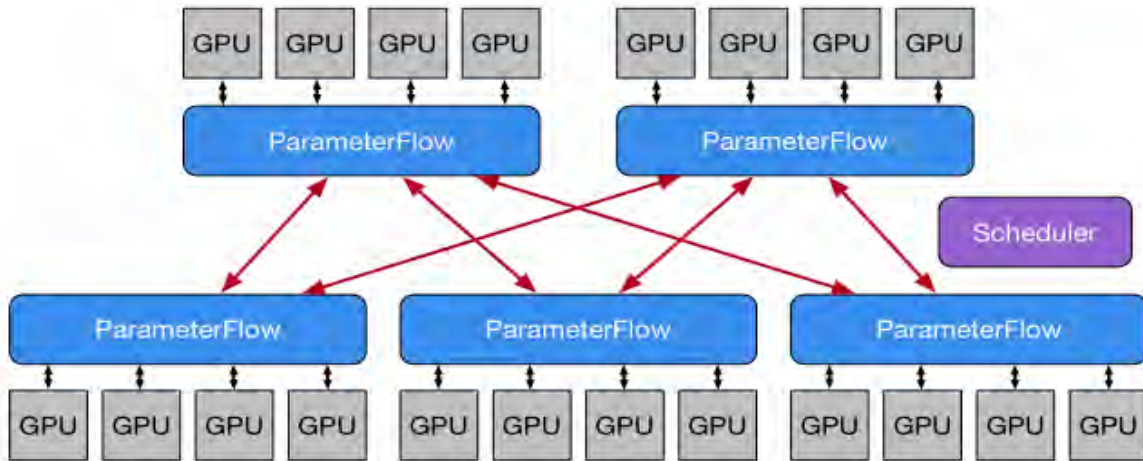
Unstructured



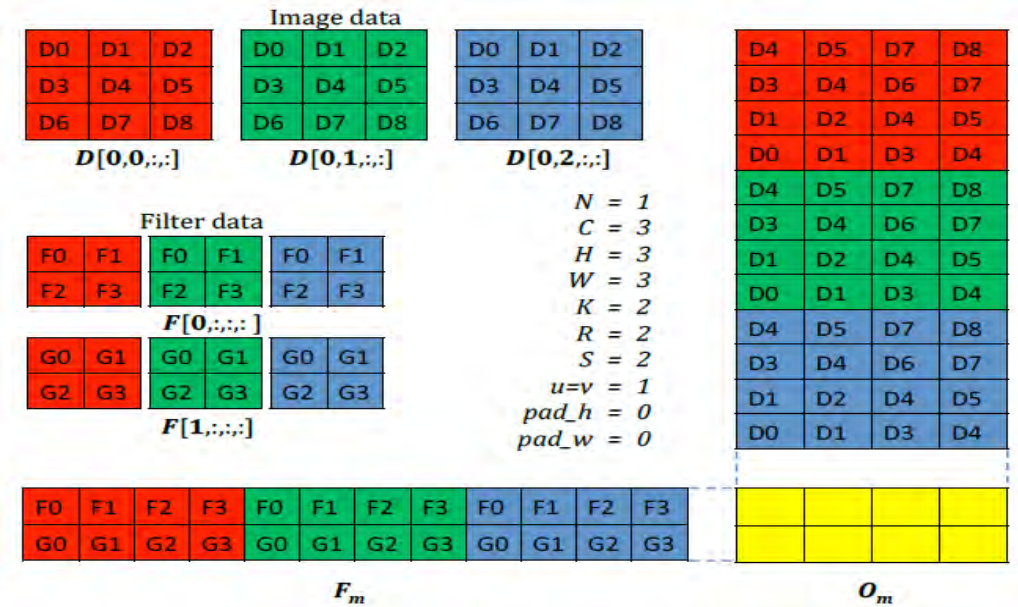
- Use NoSQL database, e.g. BigTable and HBase

Customized Middlewares

Parameter Flow



High-performance Convolution



Excellent Stability

□ Continually working for months

- Even with adverse conditions, e.g. GPU burned.

□ Quick restoration after catastrophe

- We recovered all training sessions after a power outage due to typhoon.



Summary

- **Technical infrastructure is crucial to the success of deep learning applications**
- **We have solutions across the entire stack**
 - **PPL:** computation modules (supporting CPU, GPU, and embedded devices)
 - **Parrots:** deep learning framework (efficient, scalable, and flexible)
 - **DeepLink:** sophisticated and versatile cluster training platform
- **Top-notch performance and reliability**

Q & A

□ **Thank you**

2017 CPP-Summit

predixy : 一站式redis解决方案

陈福真

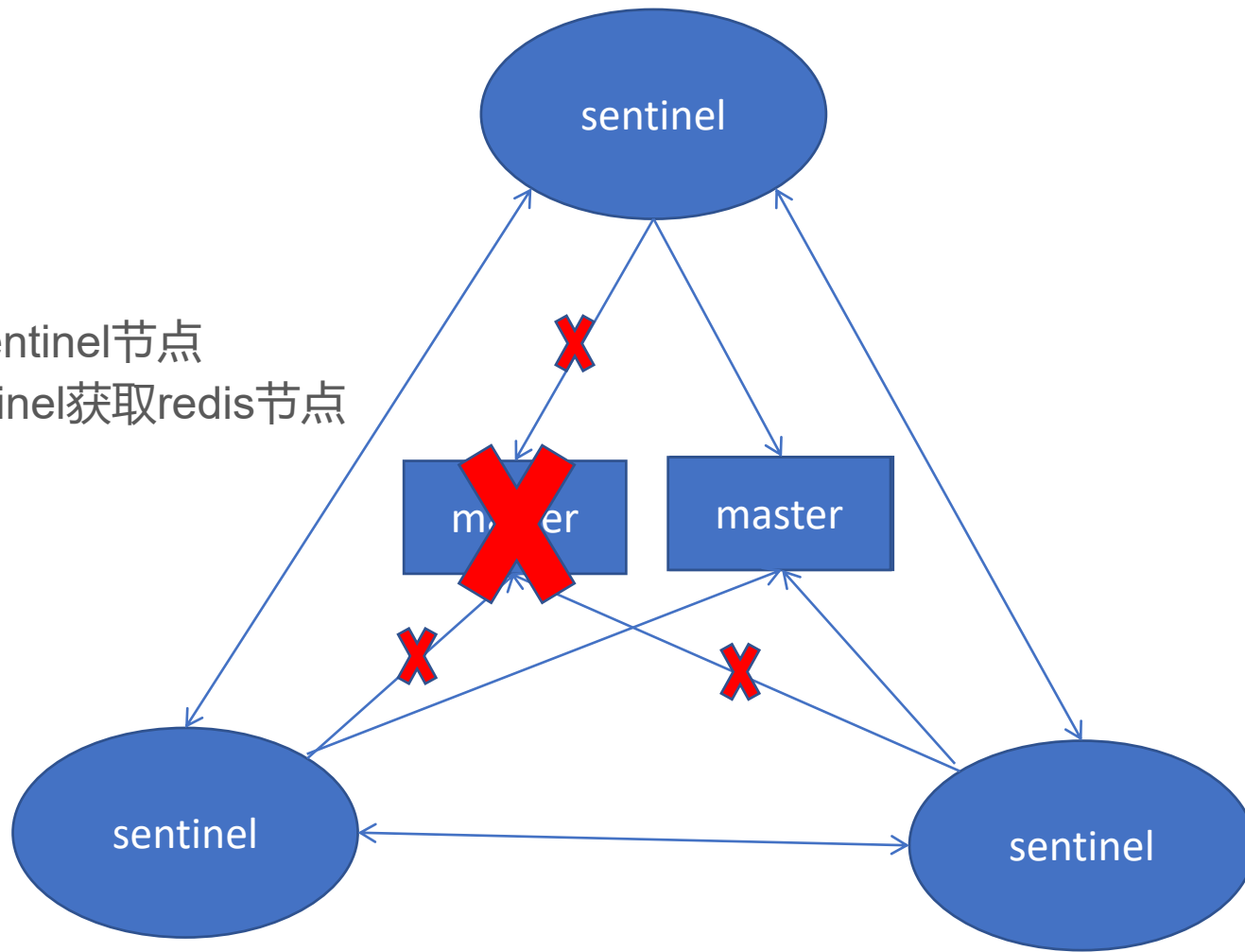
北京简易得科技有限公司创始人

- redis
 - 简介
 - 高可用
 - 可扩展
 - 使用方案
- predixy
 - 需求
 - 整体架构
 - 特性
 - 评测
- Q&A

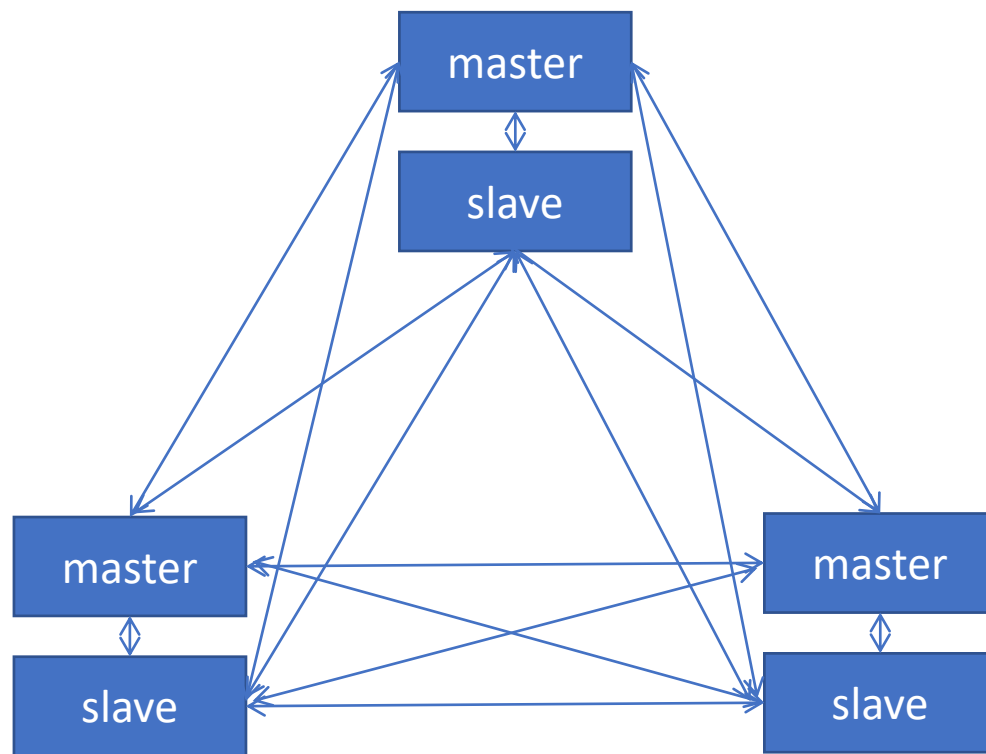


- REmote DIctionary Server
- 特性
 - 开源的、ANSI C编写的、高性能key-value内存存储系统
 - 丰富的存储数据结构：string/list/set/sorted set/hash
 - 发布订阅系统
 - 可持久化数据到硬盘
 - 可主从数据同步
 - redis sentinel高可用
 - redis cluster集群
- 使用场景
 - 缓存使用，类似memcached
 - key-value存储
 - 消息队列
 - 排行榜
 - 计数器
 - 聊天、推送

- lvs/haproxy
- 一致性哈希
 - twemproxy
- redis sentinel
 - 至少部署三个sentinel节点
 - 客户端通过sentinel获取redis节点信息



- 一致性哈希
- 代理或客户端自己分片
- redis cluster
 - 实用最小规模为主三三从
 - 最大推荐不超过1000个节点
 - CRC16求key哈希分布到16384个slot上
 - 要求客户端额外支持MOVE和ASK响应
 - 不支持跨slot的多key操作



- 常规redis
 - 功能完备
 - 内存容量有限，实例吞吐也有限
 - 通过redis sentinel来实现高可用需要客户端额外支持
- redis cluster
 - 功能受限，跨slot的多key操作不支持
 - 易于扩展，内存和吞吐都不会受到单机限制
 - 需要客户端额外支持

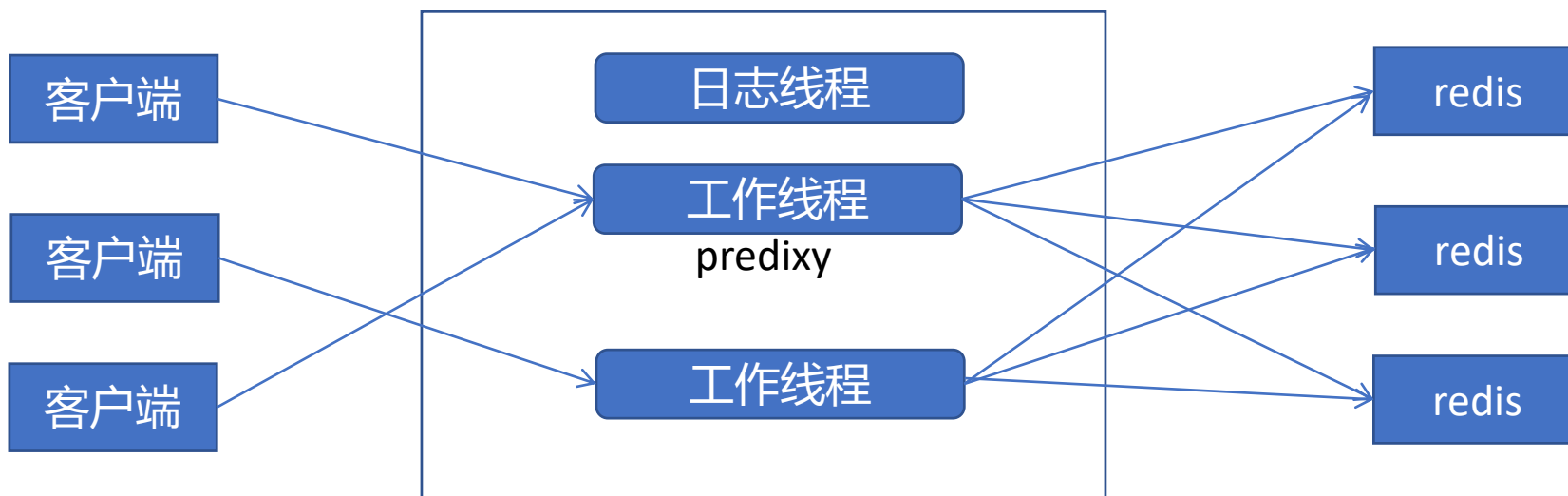


- 高性能
- 支持redis sentinel监控的常规redis
- 支持redis cluster
- 完整支持客户端命令
- 易运维

现有的一些代理

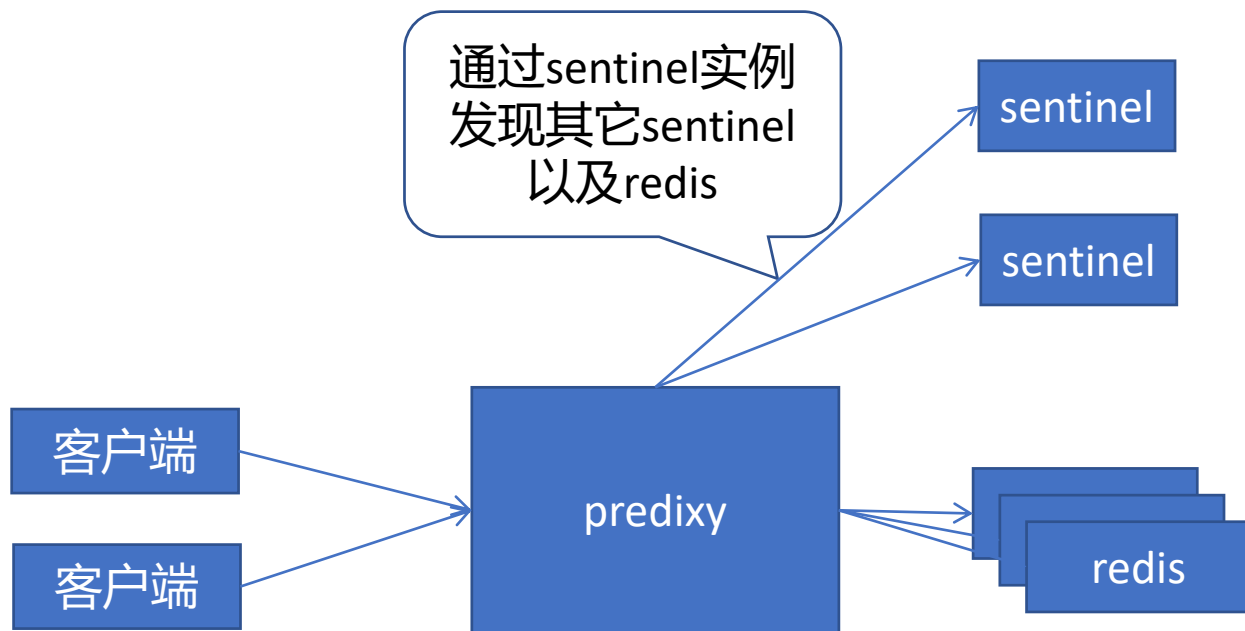
代理	语言	简介
twemproxy	C	支持命令有限，不支持redis sentinel和redis cluster
codis	GO	利用redis-sentinel来做高可用，自己实现的类似redis cluster的集群方案
cerberus	C++	只支持redis cluster
corvus	C	只支持redis cluster

- 语言选择
 - C
 - C++ ✓
 - GO
- 进程/线程模型
 - 单线程
 - 多线程 ✓
 - 多进程

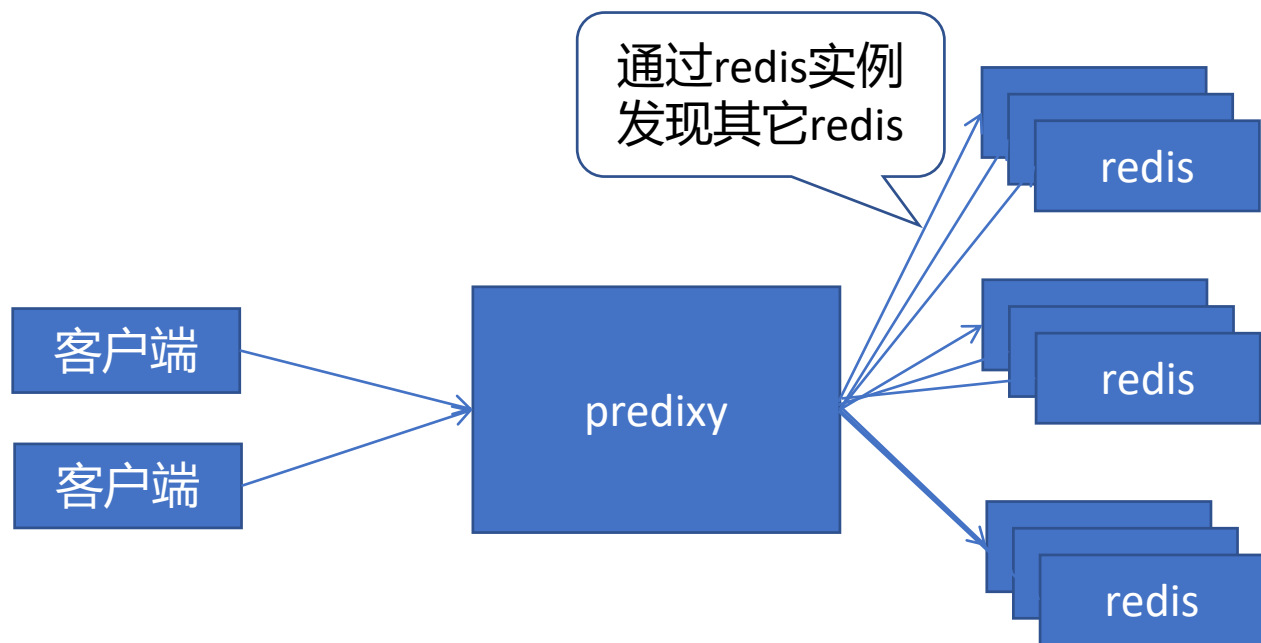


- 线程间独立的
 - 每个工作线程都有自己的事件处理循环（对应epoll的话就有一个epoll文件描述符）
 - 客户端与predixy建立连接后，该连接永远都在同一个工作线程中处理
 - 每个工作线程都与后端redis实例有独立的连接
- 线程间共享的
 - 后端redis实例信息
 - 日志记录

- 适用于redis单一主从部署结构
- predixy提供完整的redis功能(针对server的命令除外)



- predixy支持的命令同redis cluster一样（针对server的命令除外）
- MSET/MSETNX/MGET/DEL/TOUCH/EXIST/UNLINK聚合支持
- SCAN支持，全局扫描所有主节点

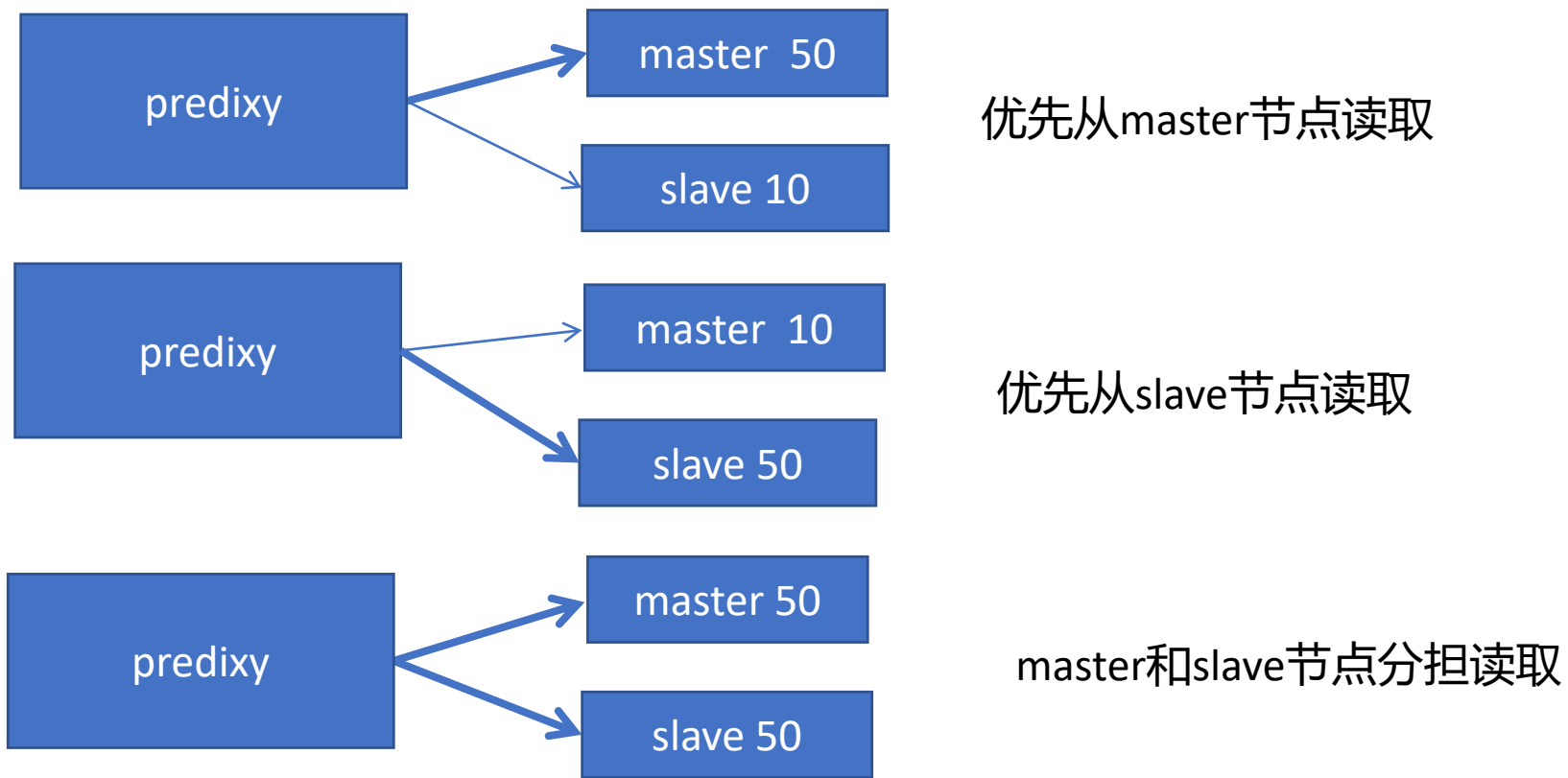


- 权限控制
 - 只读权限
 - 读写权限
 - 管理员权限
- 键空间限制
 - 可读键空间
 - 可写键空间

密码	权限	可读键空间	可写键空间
p1	read		
p2	write		
p3	read	user	
p4	write	blog	stats
p5	admin		

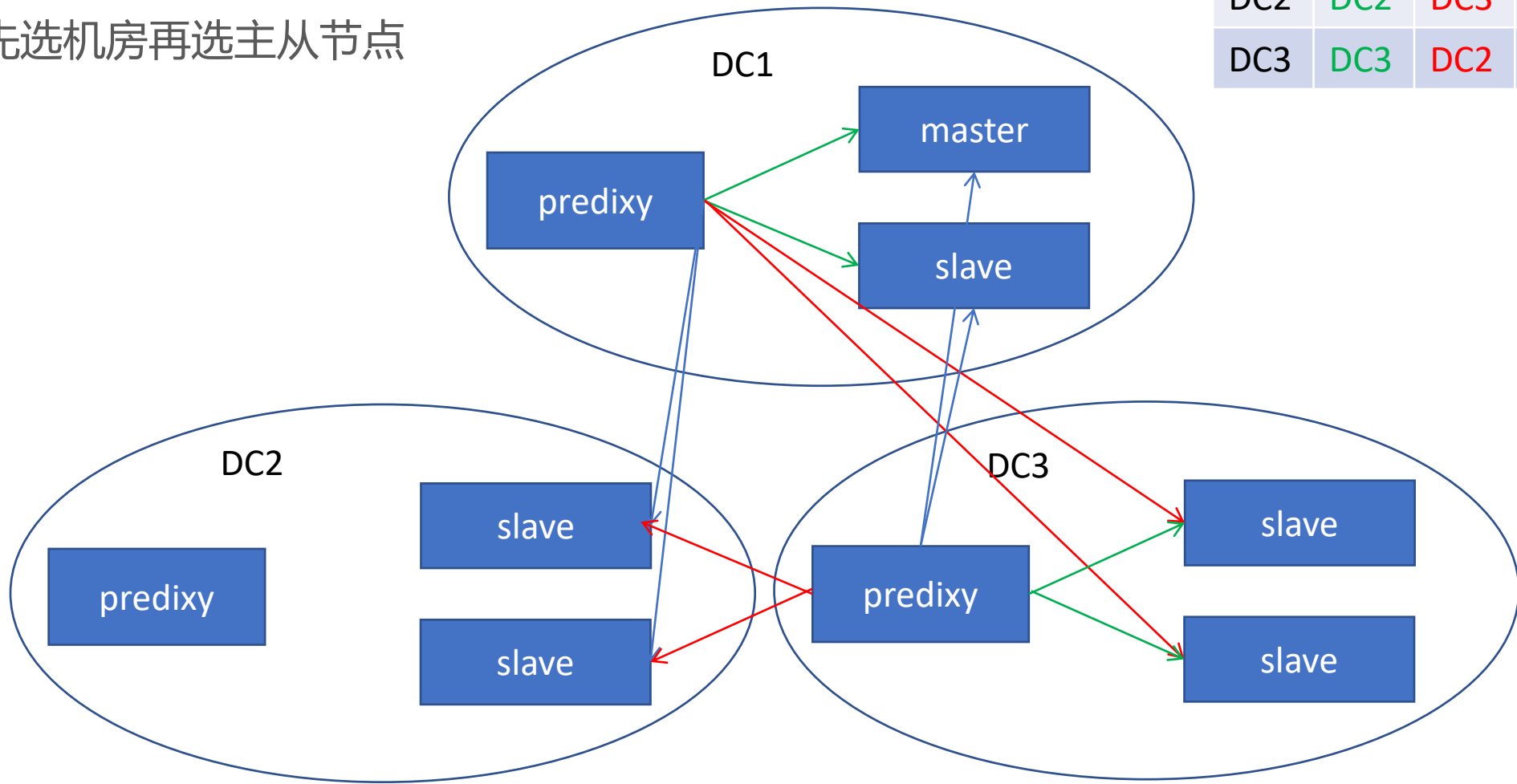
命令/密码	p1	p2	p3	p4	p5
get user.123	√	√	√	×	√
set user.123 value	×	√	×	×	√
get blog.123	√	√	×	√	√
set blog.123 value	×	√	×	×	√
set stats.123 value	×	√	×	√	√
config set LogInfoSample 10	×	×	×	×	√

- 写请求总是发给master节点
- 可分别配置master和slave节点的读优先级



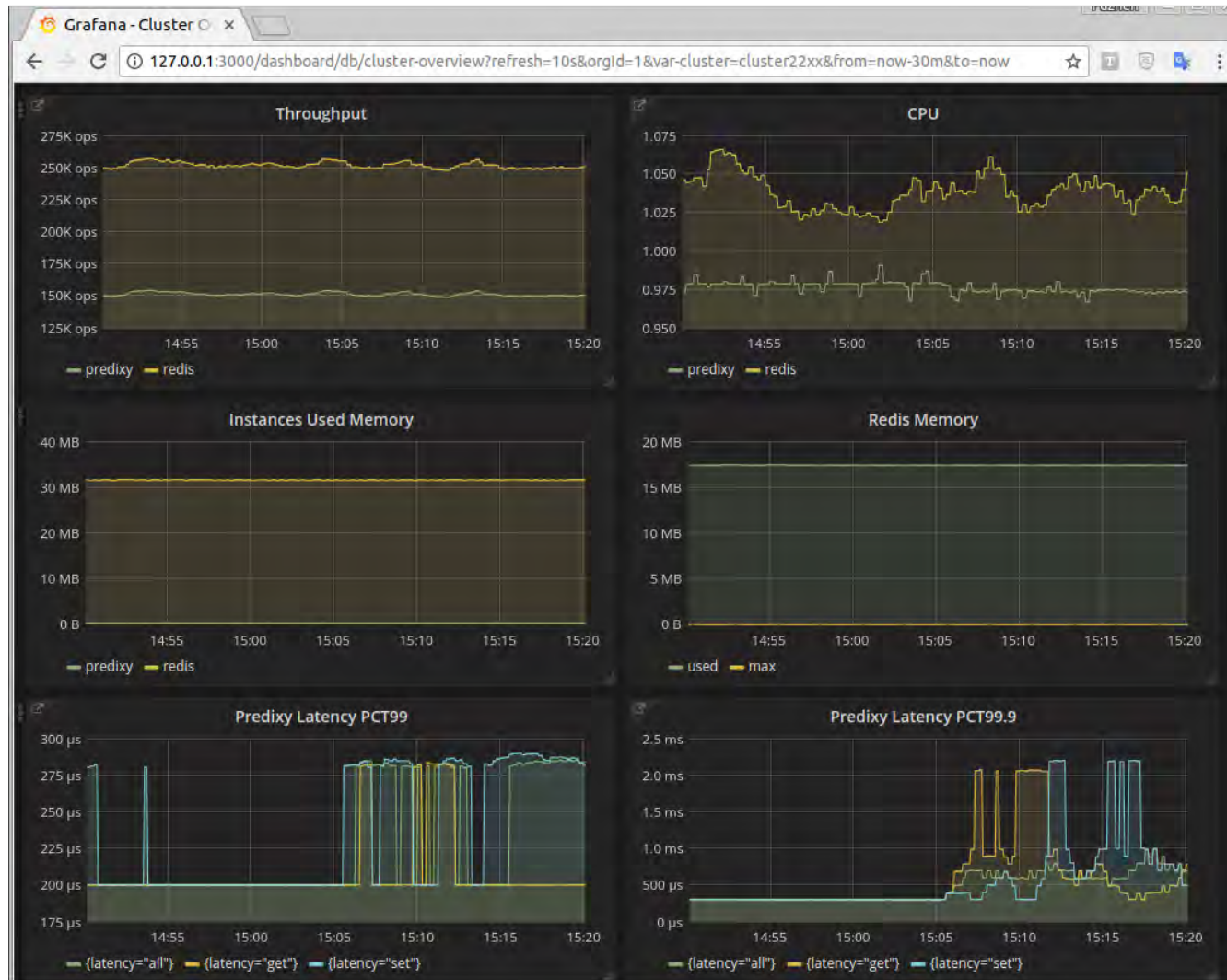
- redis节点主从跨机房部署
- 写请求总是发给master节点
- 读请求先选机房再选主从节点

DC	优先级		
DC1	DC1	DC3	DC2
DC2	DC2	DC3	DC1
DC3	DC3	DC2	DC1



- 日志异步记录
 - 独立的日志记录线程
 - 记录日志要加锁，可配置丢弃日志避免等待锁
 - 避免磁盘卡顿影响服务
- 日志按级别分别采样输出
 - Verb/Debug/Info/Notice/Warn/Error独立配置采样率
 - 可以在线实时修改
- 日志按时间、大小自动切分

- CPU、内存资源使用信息
- 客户端连接数、后端redis连接数
- 请求数统计，包括总量及各后端实例处理量
- 请求延迟统计



环境	详情
CPU	AMD Ryzen 7 1700X Eight-Core Processor 3.775GHz
内存	16GB DDR4 3000
OS	x86_64 GNU/Linux 4.10.0-27-generic #30~16.04.2-Ubuntu

代理	版本	详情
predixy	1.0.1	3个主节点的redis cluster, 平分slot
twemproxy	0.4.1	3个主节点CRC16哈希算法取模分布
codis	3.2.0	3个主节点, 平分slot
cerberus	git8a5d	3个主节点的redis cluster, 平分slot
corvus	0.2.6	3个主节点的redis cluster, 平分slot

单线程SET/GET评测

Single thread proxy SET/GET benchmark :
redis-benchmark -p xxx -t set,get -r 3000 -n 1000000 -d xxx

