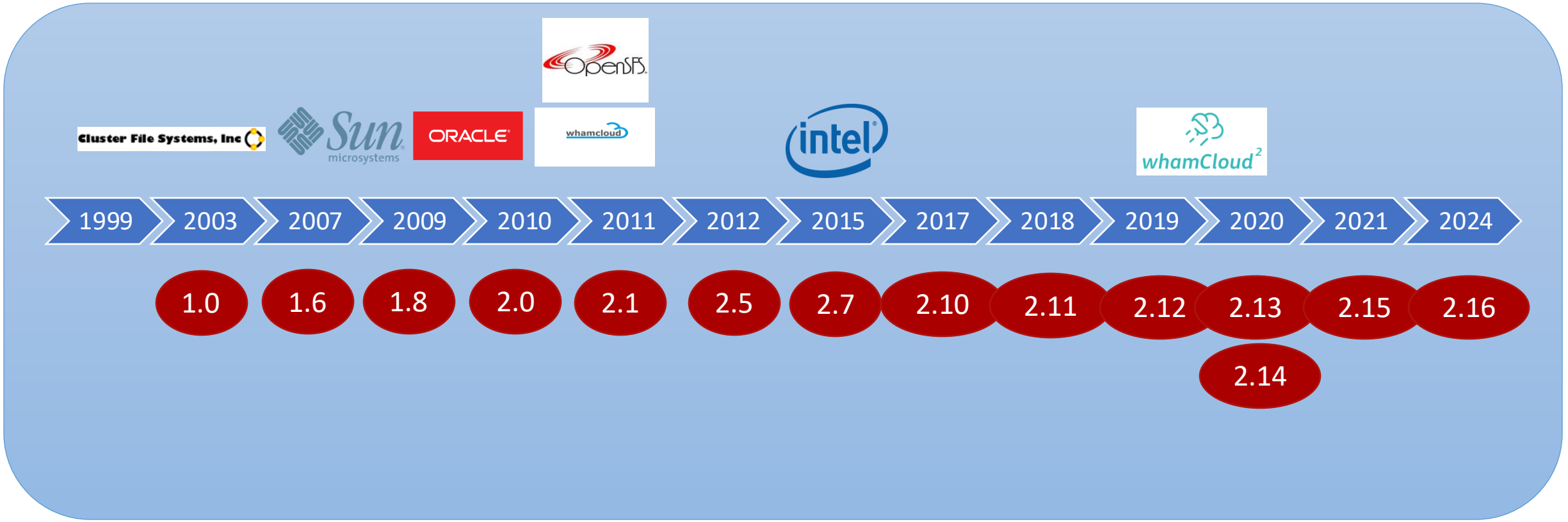
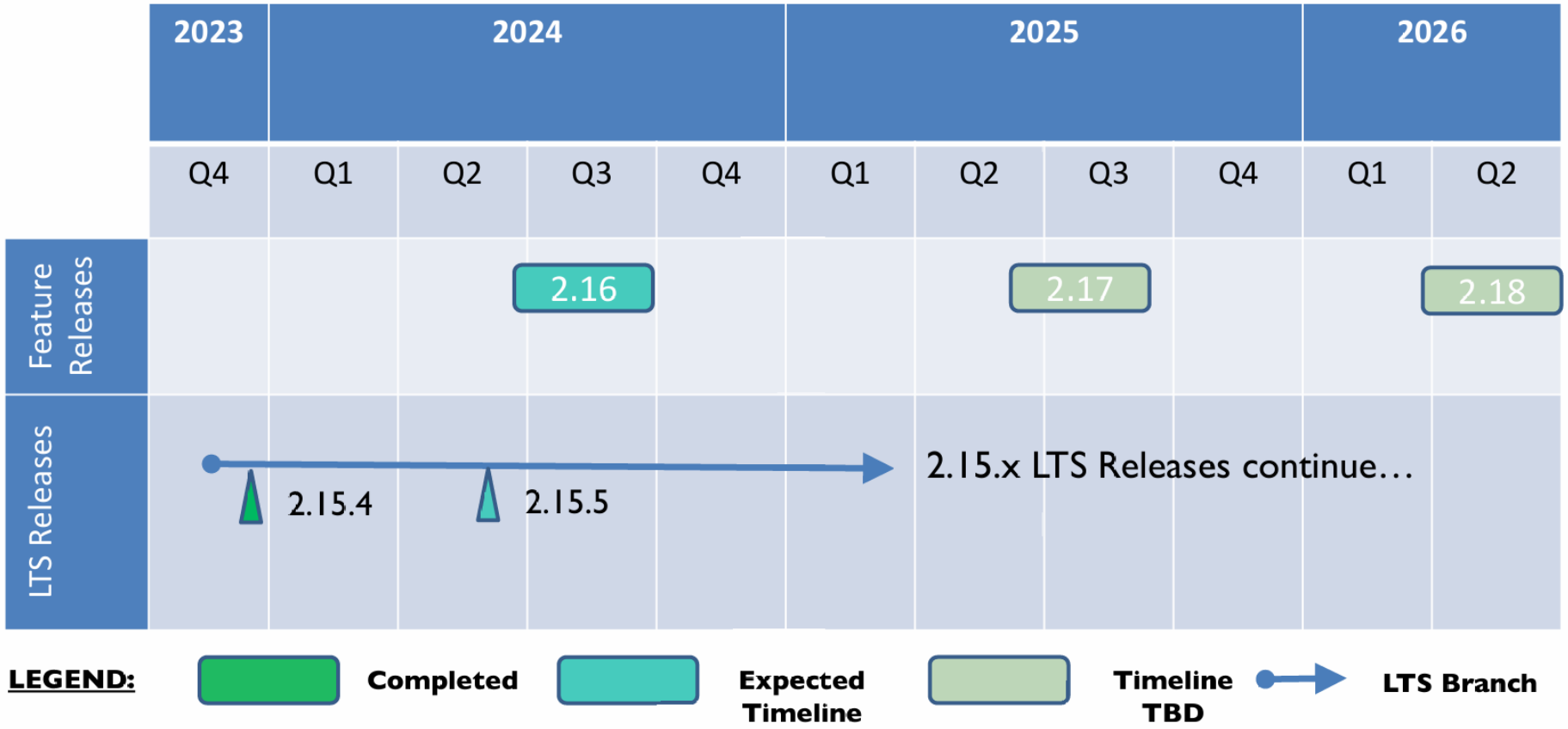
Coral： Lustre集成软件栈的一种开源实践

李希

Lustre文件系统的发展历史



Lustre社区技术路线



2.16
- RHEL9 Server Support
- IPv6
- Optimized Directory Traversal

2.17
- Client Data Compression
- Metadata Writeback Cache
- Erasure Coding

2.18
- RHEL10 Server Support
- Metadata Redundancy v1

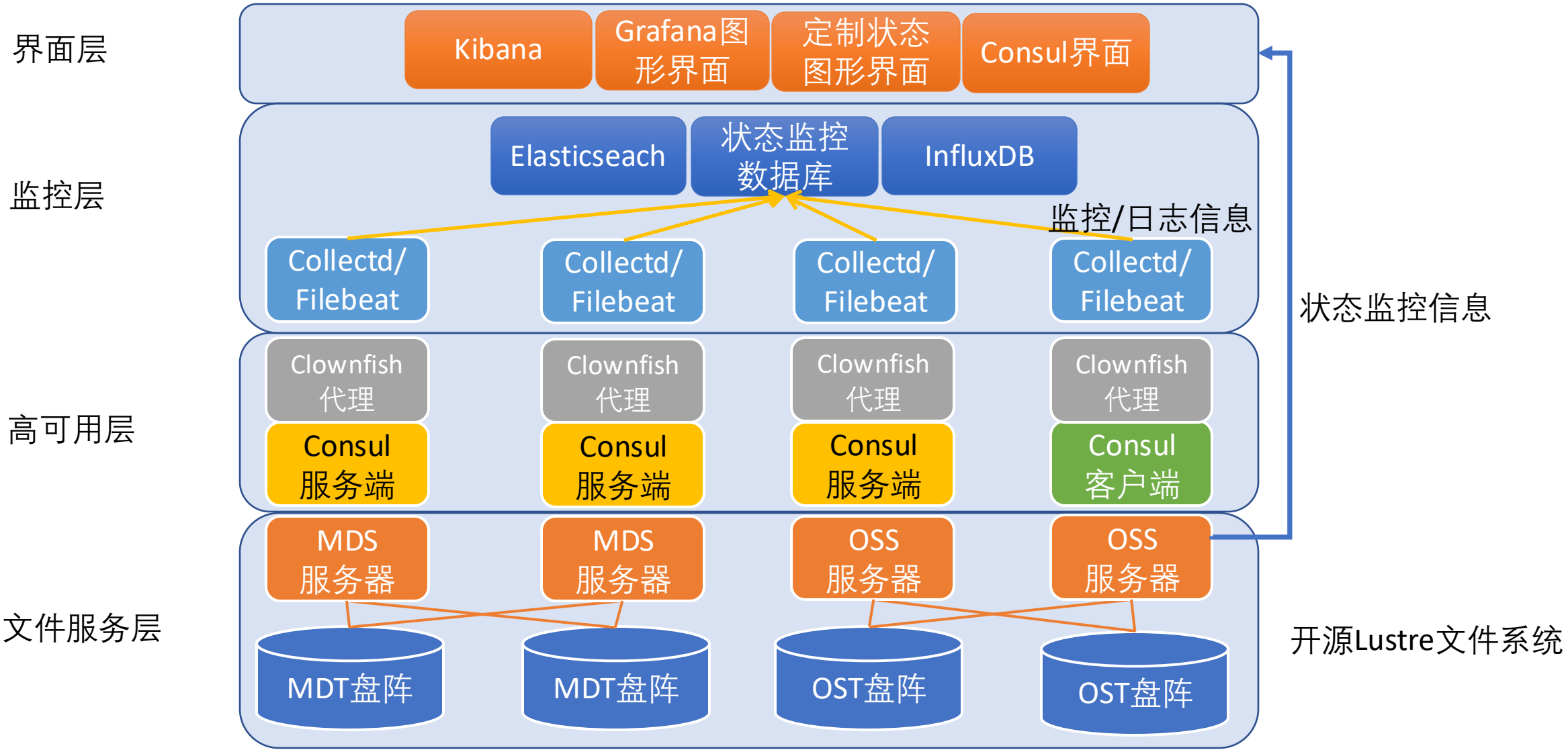
Lustre系统管理与存储管理面临的挑战

- Lustre本身只提供文件系统功能，未绑定支撑软件
- 构建Lustre系统涉及复杂的软件栈
- 周边软件庞杂，普通用户难以全部技术细节
  - 基础系统管理
    - 系统管理工具集、高可用功能、状态监控、性能监控、限额管理等
  - 数据访问接口需要对接其他系统软件
    - S3、NFS、Samba、K8S、OpenStack Cinder
  - 各类功能的实现和集成需要定制的软件支撑
    - 如安全保护、多租户、审计等
  - 数据管理
    - 数据迁移、元数据迁移、用量均衡、冷热池管理、分级存储管理等
  - 生产工具
    - 回归测试、功能验证、性能瓶颈分析等

Coral：Lustre集成软件栈的开源尝试

- Coral：集成化的Lustre发行版
  - 集成Lustre文件系统开源发行版（2.12/2.15）
  - 集成其他各类Lustre周边软件功能
  - 开源： https://gitee.com/filesystem/barreleye
- 简化相关软件系统的安装、配置与使用
  - 例如，LDAP的安装配置较复杂，Lustre的并行计算环境需要它
- 组合Lustre内部各功能，性能完善的功能体系
  - 例如，如何综合利用subdir mount、project quota、nodemap、root squash实现多租户功能
- 组件化设计与实现
  - 每个组件各自实现一套功能
  - 各组件间有机结合，避免功能和代码重叠
  - 各组件可独立安装部署，也可组合使用
  - 软件体系结构可扩展，新组件不断丰富和拓展

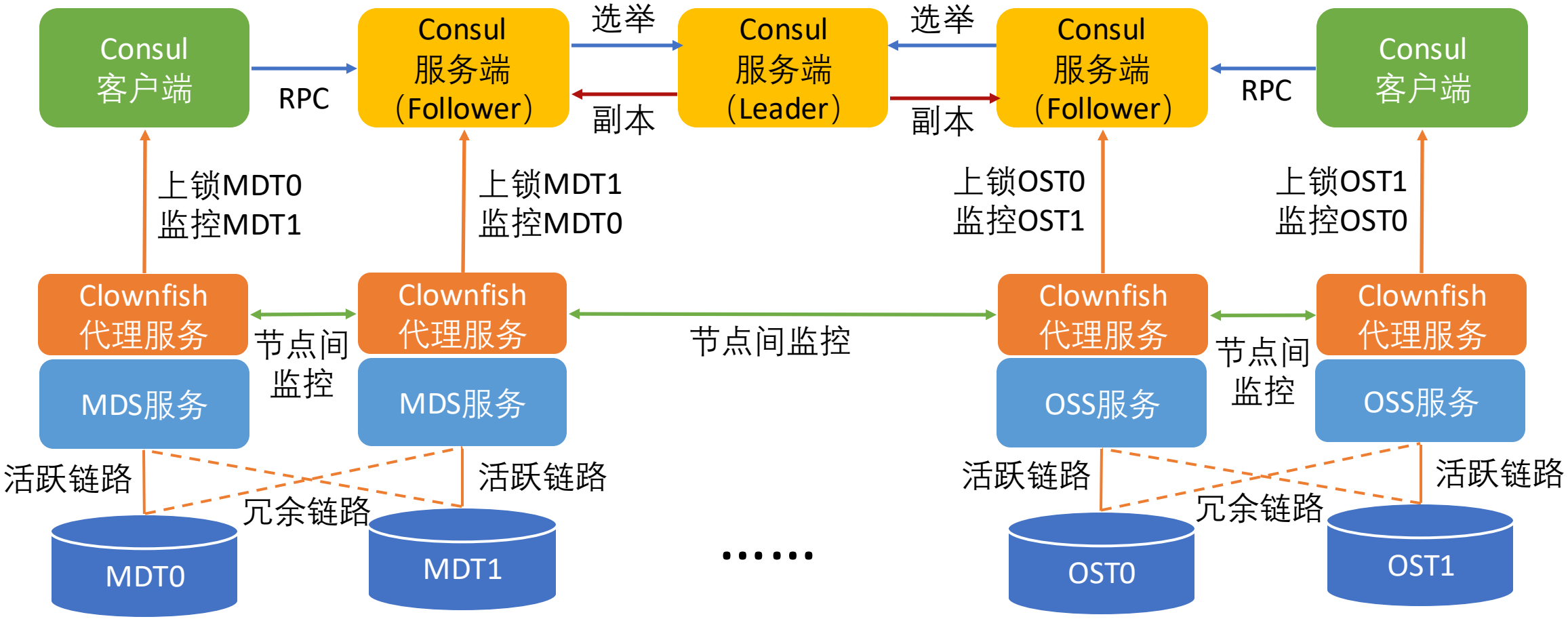
Coral基础软件栈的结构原理



Coral-Clownfish组件

- 利用Consul(ZooKeeper/etcd)新型高可用软件替换Corosync/Pacemaker
- 利用Consul的Key/Value存储，对Lustre存储系统进行动态配置
- 利用Consul的图形界面实现配置的图形化管理
- Clownfish组件特点
  - 一体化：clownf系列命令
  - 分布式：适合支持包含众多服务节点的Lustre集群
  - 高可用：可支撑Lustre高可用机制的各方面需求
  - 可横向扩展：可支撑Lustre添加服务节点和存储目标的需求

Coral-Clownfish的高可用机制的原理



采用Coral-Clownfish简化Lustre管理流程

COFS China Open File System

- 安装Coral软件包

```
# mount -o loop coral-*.iso /mnt/iso/
# rpm -ivh /mnt/iso/Packages/coral-*
```

- 编辑Clownfish配置
  - /etc/coral/clownfish.conf单文件存储了关于Clownfish集群的所有配置信息，包含：Lustre文件系统名、各存储设备、集群内各服务节点、存储网络配置等
  - /etc/coral文件下有配置实例供参考

```
# cp /etc/coral/clownfish.conf.example /etc/coral/clownfish.conf
```

- 安装和配置Clownfish集群
  - 在集群所有节点上安装Coral软件包、同步Clownfish配置、安装和配置Consul软件、构建Clownfish高可用集群
  - 不包含Lustre软件的安装

```
# clownf cluster install --iso coral-*.iso
```

- 在Clownfish集群上安装Lustre软件
  - 在集群内所有Lustre节点上安装Lustre服务器内核、启动内核、安装Lustre软件包

```
# clownf cluster prepare
```

- 格式化Lustre文件系统
  - 按/etc/coral/clownfish.conf的配置格式化Lustre文件系统的MGS、MDT和OST

```
# clownf cluster format
```

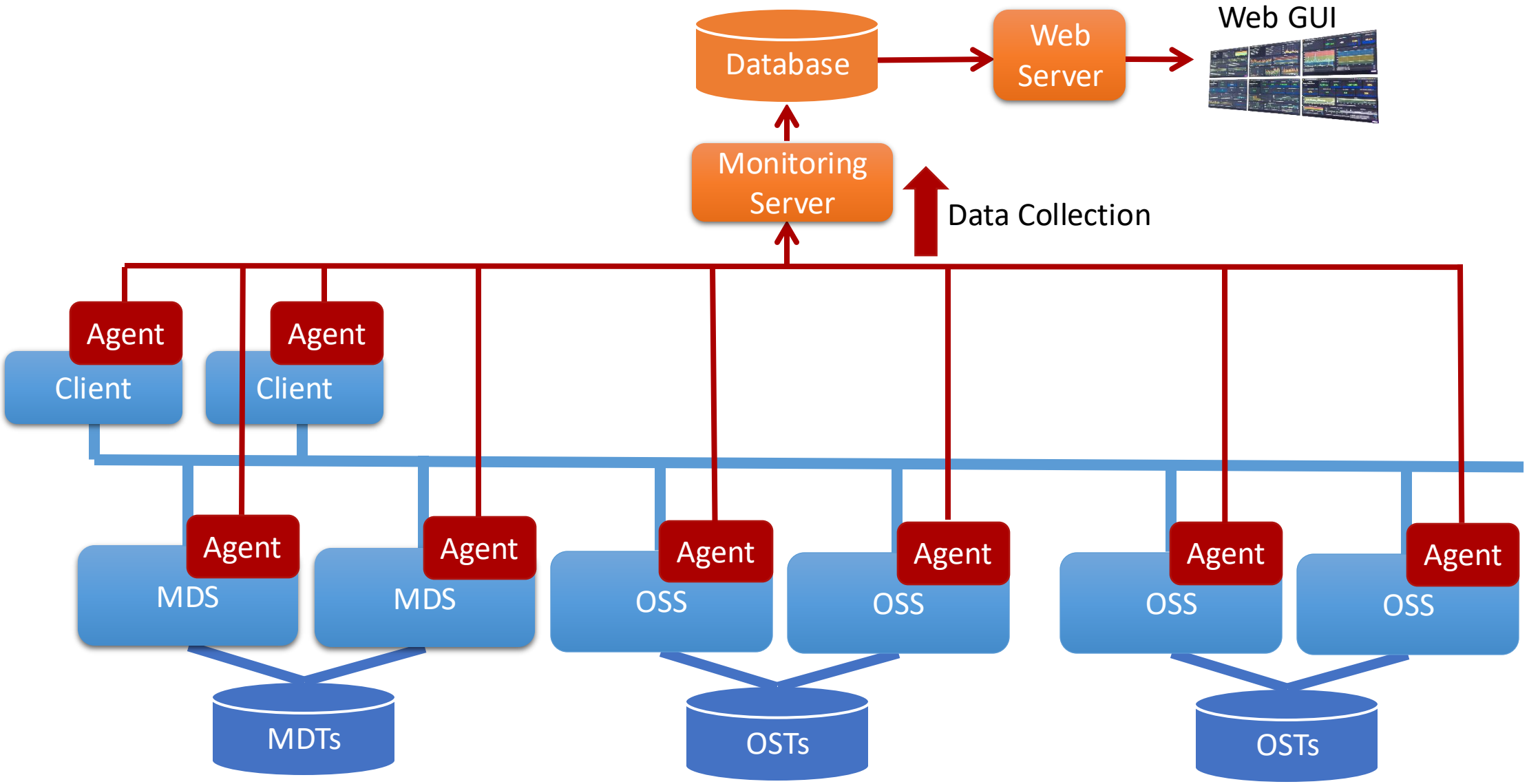
- 挂载Lustre文件系统

```
# clownf cluster mount
```

Coral-Barreleye组件

- Coral提供的Lustre性能监控系统
- Collectd-5.12 + 深度定制的Lustre数据收集插件
- Lustre数据格式的XML描述文件
- Grafana-7.3.7 + 定制的Lustre仪表盘
- 时序数据库Influxdb-1.8.4
- 一体化管理命令行： barrele
- 支持Lustre版本2.7、2.10、2.12、2.15
- 支持CentOS7、Rocky8、Ubuntu
- 支持X86_64和ARM架构

Coral-Barreleye性能监控系统的架构



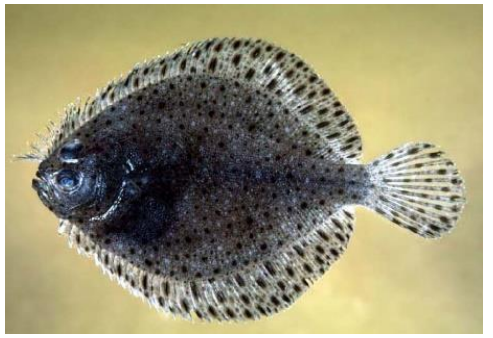
Coral-Reaf组件

- Reaf：Coral基础组件，提供其他组件所需的通用基础功能
  - Lustre版本的检测及管理
  - E2fsprogs版本的检测及管理
  - 通用型接口及命令，弥补Lustre现有接口的不足
    - 查询单个OST/MDT的状态
- Coral-Reaf组件的特点
  - 可灵活拓展对Lustre各版本的支持
  - 支持基于配置文件的自定义版本，可自由添加对自有版本的支持

Coral-Reaf组件命令一览

- Lustre版本管理
  - reaf lustre create：创建Lustre发行版
  - reaf lustre ls：列出所有Lustre发行版
  - reaf lustre status：查看某个Lustre发行版的状态
  - reaf lustre copy：复制Lustre发行版
  - reaf lustre update：更新Lustre发行版的元数据
  - reaf lustre download：下载某个Lustre发行版下的文件
  - reaf lustre file_add：向某个Lustre发行版添加文件
  - reaf lustre file_remove：删除某个Lustre发行版添加文件
  - reaf lustre files：列出某个Lustre发行版的所有文件
- E2fsprogs版本管理
  - 与Lustre版本管理命令类似

Coral-Flatfish组件



- Flatfish：Coral提供的Lustre配置管理机制
- 包含对百余个Lustre参数的配置方法
- 简化参数的配置流程
  - 传统流程，以设置jobid_var为例：
    - 1. 登录某个可能运行MGS的节点io00，运行`lctl conf_param $FSNAME.sys.jobid_var=SLURM_JOB_ID`命令；
    - 2. 由于MGS服务并未运行在io00上，而是运行在io01上，登录io01重新执行第1步；
    - 3. 登录某个客户端，运行`lctl get_param jobid_var`，检查输出结果是否为`SLURM_JOB_ID`；
    - 4. 因为参数设置延迟，第2步尚未成功，反复尝试第3步直至成功。
  - Flatfish流程：在任意节点上运行`flatf param change jobid_var --kv jobid_var=LURM_JOB_ID`即可完成所有设置及检查操作
- 支持定义一组参数设置链
  - 实现性能优化目标的一组参数链，如设置RPC大小为64MB
  - 实现某功能参数链，如设置NRS TBF完成QoS功能需要改变多个参数

Flatfish操作实例


```
# flatf param man jobid_var
NAME
    jobid_var - The environment variable that holds the JobID for the process.

SYNOPSIS
    flatf param change jobid_var --kv jobid_var=JOBID_VAR

DESCRIPTION
    Any environment variable can be specified for the JobID of the process. The
    Lustre jobstats code on the client extracts the unique JobID from an
    environment variable within the user process, and sends this JobID to the
    server with the I/O operation. The server tracks statistics for operations
    whose JobID is given, indexed by that ID.
…
# flatf param change jobid_var --kv jobid_var=LURM_JOB_ID
INFO: changing parameter [jobid_var] with KV [jobid_var=LURM_JOB_ID]
INFO: running command [lctl set_param jobid_var="LURM_JOB_ID"] on host [mds0], expected value: [LURM_JOB_ID]
INFO: running command [lctl set_param jobid_var="LURM_JOB_ID"] on host [mds1], expected value: [LURM_JOB_ID]
INFO: running command [lctl set_param jobid_var="LURM_JOB_ID" -P] on host [mds0], expected value: [LURM_JOB_ID]
INFO: running command [lctl get_param jobid_var] on host [client0], expected value: [LURM_JOB_ID]
INFO: running command [lctl get_param jobid_var] on host [client1], expected value: [LURM_JOB_ID]
```

Coral-Flatfish组件命令一览

- flatf param ls：列出所有支持的参数名字
- flatf param man：查看某参数的详细解释
- flatf param change：在Lustre文件系统上设置某参数的值
- flatf chain create：创建一个空的参数链
- flatf chain declare：在参数链中添加一个参数键值对
- flatf chain ls：列出已定义好的所有参数链
- Flatf chain params：列出参数链中的所有参数
- flatf chain remove：删除某个参数链
- flatf chain undeclare：删除一个参数链中的一个参数键值对
- flatf chain apply：在Lustre文件系统上设置参数链中的所有参数

Coral-Leaffish组件

- Leaffish：Coral提供的LDAP（轻量级目录访问协议）管理机制
- LDAP广泛应用于用户统一登陆管理
- 有多种高可用模式，配置 较复杂
- Coral-Leaffish组件
  - 简化LDAP的安装配置
  - 配置高可用镜像模式，适合生产系统
  - 简化用户和用户组的日常管理维护

Coral-Leaffish组件命令一览

- leaff cluster install：在集群中安装所有LDAP服务器和客户端
- leaff client install：安装和配置LDAP客户端
- leaff group ls：列出已配置的LDAP用户组
- leaff group add：向LDAP添加用户组
- leaff group delete：从LDAP删除用户组
- leaff user ls：列出已配置的LDAP用户
- leaff user add：添加LDAP用户
- leaff user delete：从LDAP删除用户

Coral-Lionfish组件

- Lionfish：Coral提供的LVM（Logical Volume Manager）管理工具
- LVM可提供简单高效的系统级快照功能
- LVM可提供灵活方便的存储设备管理
  - 存储设备故障替换
  - 逻辑卷大小调整
  - 逻辑卷迁移
- LVM的快照功能实现文件系统级备份中
  - 防止备份中的文件系统数据不一致
- 后续将基于Coral-Lionfish实现Lustre文件系统备份功能

Coral-Lionfish组件命令一览

- lionf cluster install：在集群内安装逻辑卷管理工具
- lionf cluster reload：在集群内重新加载逻辑卷
- lionf volume create：创建一个逻辑卷
- lionf volume ls：列出所有的逻辑卷
- lionf volume remove：删除某个逻辑卷
- lionf volume status：查看某个逻辑卷的状态
- lionf volume instances：列出某个逻辑卷在多个节点上的实例
- lionf host ls：列出集群中配置的所有节点
- lionf host instances：列出某个节点上的所有逻辑卷实例

Coral-Tilefish组件

- Coral提供的Lustre日志分析及错误报警机制
- knowledge_lustre-{version}.yaml
  - 用于定位日志的生成位置
  - 解析并分析源代码，生成日志的知识信息
- expertise_lustre-{version}.yaml
  - 人工分析了数百个常见日志，确定了其所对应的事件
  - 定义了数十个常见事件及其含义
- 日志收集、解析、存储、分析
  - Filebeat：轻量级日志分析
  - Logstash：日志的收集、解析和转换
  - Elasticsearch：日志的存储
  - Kibana：日志的分析和可视化
- 每日状态汇报
  - 自动生成汇报文档
  - 自动向邮件列表发送汇报邮件

knowledge_lustre-{version}.yaml实例


```
lustre_src: /git/lustre-release.git/
messages:
 - codes: "\t\tCWARN(\"%s: write to upcall name
%s\\n\",\n\t\t      mdt_obd_name(mdt),\
      \ hash->uc_upcall);"
    codes_in_parentheses: "\"%s: write to upcall
name %s\\n\",\n\t\t      mdt_obd_name(mdt),\
      \ hash->uc_upcall"
    codes_in_quotes: '%s: write to upcall name %s'
    codes_one_line: '%s: write to upcall name %s'
    file_path: lustre/mdt/mdt_lproc.c
    keywords:
      - ': write to upcall name '
    line_number: 312
```

messages：多条日志信息的列表

codes：打印日志的Lustre源码

codes_in_parentheses：打印函数（宏）括号中的源码

codes_in_quotes：打印函数（宏）引号中的源码

codes_one_line：将可能的多行缩略为单行

file_path：Lustre源码文件的路径

keywords：匹配日志的关键字列表

linenumber：打印代码在源码中的行数

expertise_lustre-{version}.yaml实例


```
disabled_messages:
  - codes_one_line: '%#x < %#x’
  - codes_one_line: 'Short %s: %d(%d)'
events:
  - type: connect
    description: 'The status of connection between
peers is moving from disconnected to connected.'
  - type: disconnect
    description: 'The status of connection between
messages:
  - codes_one_line: '%s: Connection restored to %s
(at %s)'
    events:
      - connect
```

disabled_messages ：不尝试匹配的日志列表

codes_one_line：不尝试匹配的日志源码

events：事件列表

type：事件类型

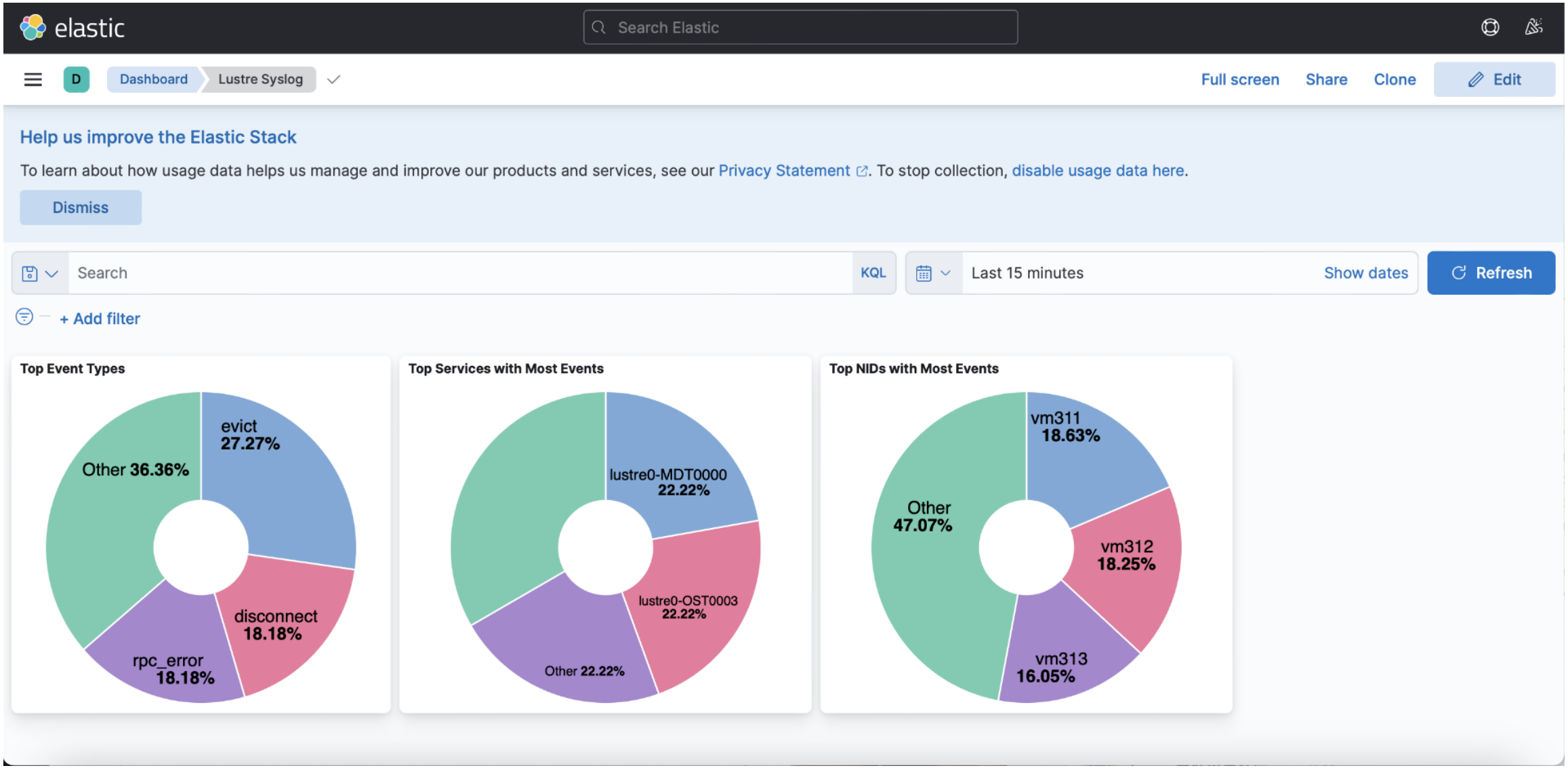
description：对于事件的描述

messages：日志列表

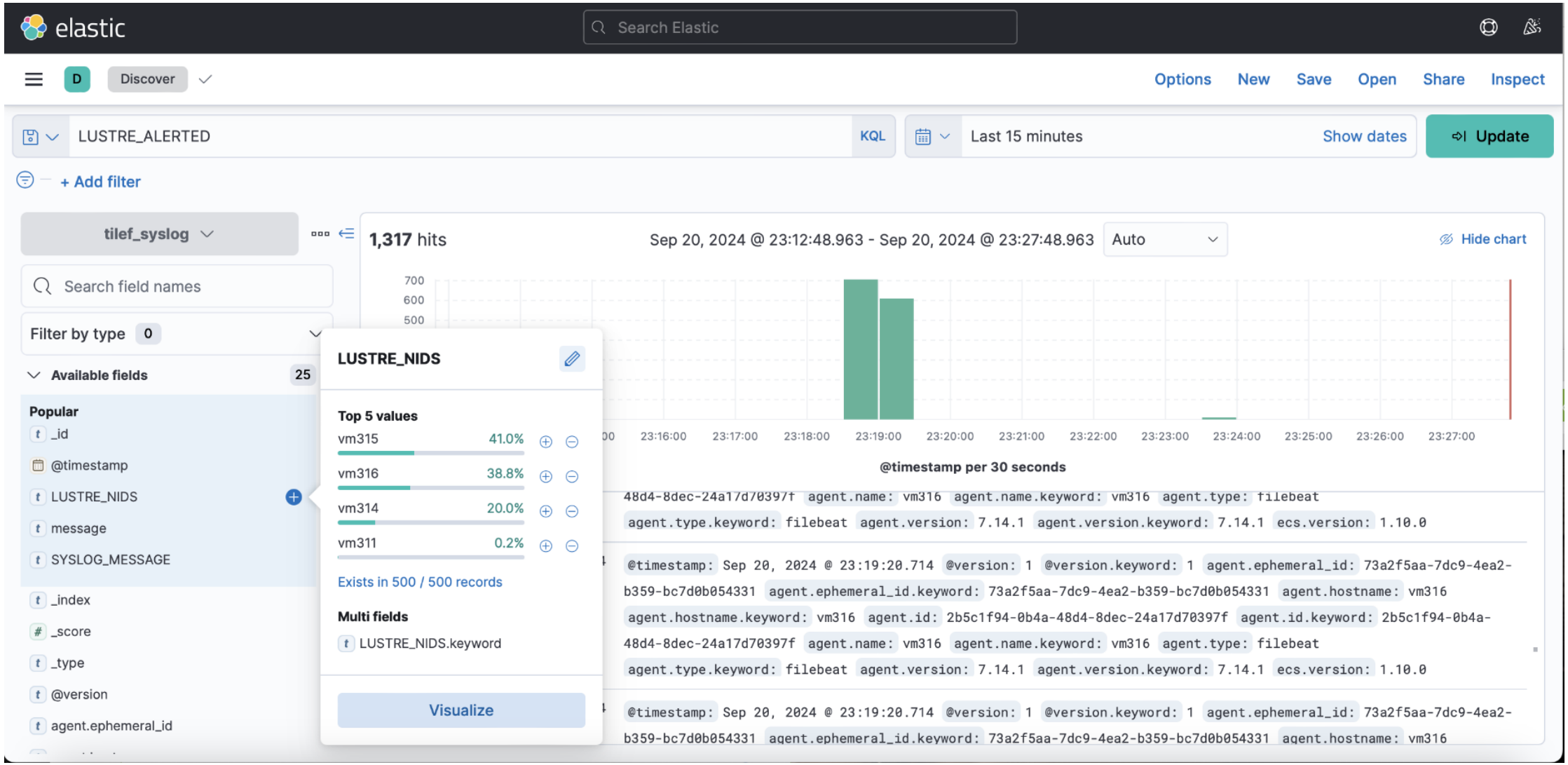
codes_one_line：日志匹配的源码

events：该日志对应的事件列表

Coral-Tilefish组件的Dashboard



Coral-Tilefish组件的Dashboard



Coral-Tilefish组件每日Lustre状态汇总

COFS China Open File System

Report from Tilefish 安全浏览模式

发件人:

收件人:

时 间: 2024年09月20日 23:50 (星期五)

附 件: 1个(tilef_report.docx) 查看附件

翻译成中文

Hello, please check the attachment for the Tilefish report.

附件 (1)

tilef_report.docx

115.31K

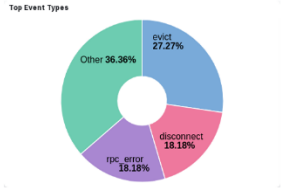
Tilefish Report

Daily Report of Your File System

This is the review of your Lustre file system from Thu Sep 19 2024 23:50:42 to 1 day later.

There is no Lustre error messages in your Lustre file system. The number of Lustre messages increases to 11 from zero. Something happened in your Lustre file system.

The following is the top Lustre events happened during this period:

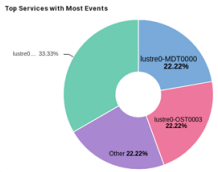


The following is the full list of Lustre events happened:

Event Type	Happen Number	Description
evict	3	Server intentionally evicts client. This is different with [disconnect] event in the sense that client is actually able to connect to server.
similar_messages	2	Several previous similar messages are skipped. Not much information in this message.
rpc_error	2	Failure happened when sending/receiving a RPC.
disconnect	2	The status of connection between peers is moving from connected to disconnected.
connect	1	The status of connection between peers is moving from disconnected to connected.
connect_error	1	Failure happened when clients/servers connect.

The following graph shows how many events happened to each of your Lustre services.

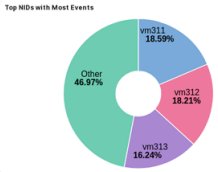
Tilefish Report



The following is the full list of Lustre services with events:

Service	Event Number
lustre0-OST0001-osc-MDT0000	3
lustre0-MDT0000	2
lustre0-OST0003-osc-MDT0000	2
lustre0-OST0003	2
lustre0-OST0001	1
lustre0-MDT0001	1

The following graph shows how many events happened to each of your Lustre NIDs. NID means network ID. Different Lustre service/client hosts have different NIDs



The following is the full list of Lustre NIDs with events:

NID	Event Number
10.0.2.83@tcp	6
10.0.2.88@tcp	2
10.0.2.87@tcp	1
10.0.2.91@tcp	1

总结与展望

- Coral：尝试构建开源的更为全面的Lustre解决方案
  - Coral-Clownfish： Lustre高可用及日常管理
  - Coral-Barreleye： Lustre性能监控
  - Coral-Reaf： Lustre通用基础功能
  - Coral-Flatfish： Lustre配置管理机制
  - Coral-Leaffish： 提供的LDAP管理功能
  - Coral-Lionfish： LVM管理工具
  - Coral-Tilefish： Lustre日志分析及错误报警机制

- 后续计划：
  - 继续实现Lustre所需的其他功能组件
  - 实用案例

谢谢！