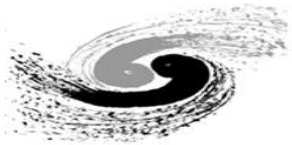


中国科学院高能物理研究所
Institute of High Energy Physics
Chinese Academy of Sciences

高能所計算中心
IHEP Computing Center

大数据驱动的Lustre I/O异常检测

Qingbao Hu(huqb@ihep.ac.cn), Lu Wang (wanglu@ihep.ac.cn)

Computing Center, Institute of High Energy Physics, CAS

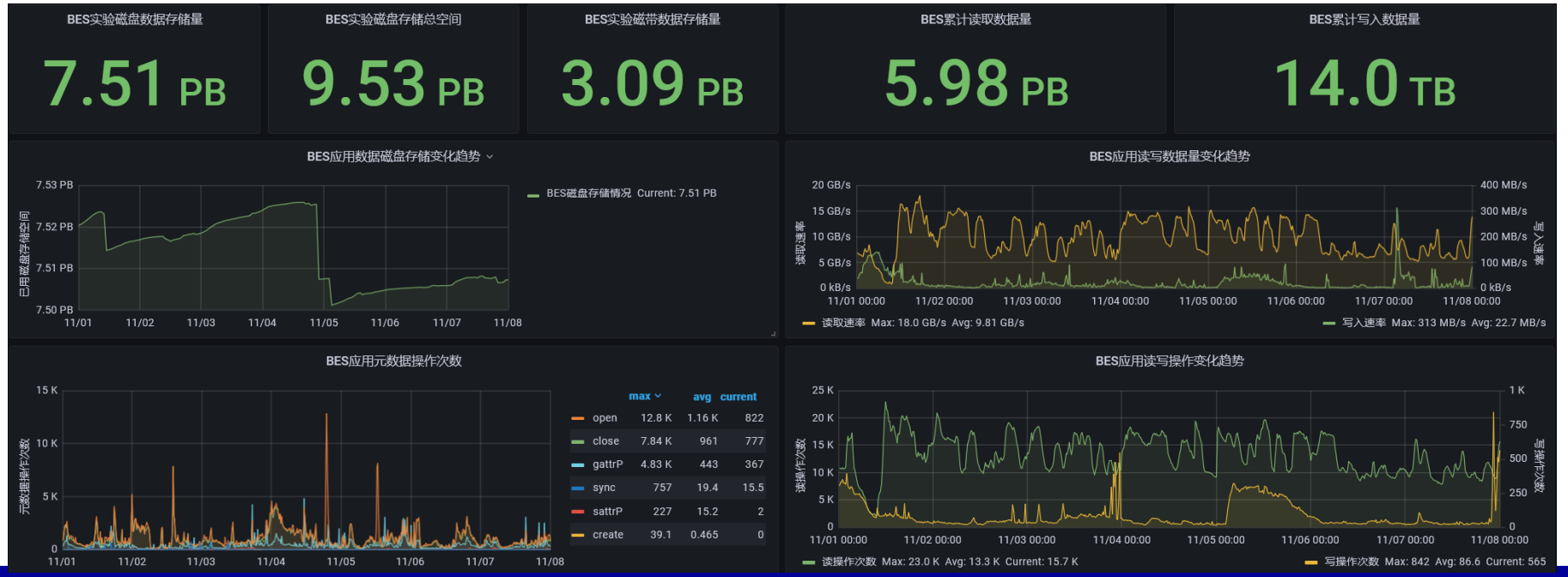
2021CLUG

Outline

1 背景需求
2 大数据运维平台
3 数据流支撑
4 异常检测
5 案例展示

背景需求

- 个别用户作业的数据异常I/O访问行为，会影响Lustre服务稳定性，从而降低整体作业运行效率。传统的异常I/O访问行为追溯，需要监控存储服务器负载、溯源负载异常的来源。平均故障恢复时间数小时左右。
- 为了缩短故障恢复时间，基于无监督学习和大数据技术实现了一种快速、自动的异常I/O访问行为检测方法。

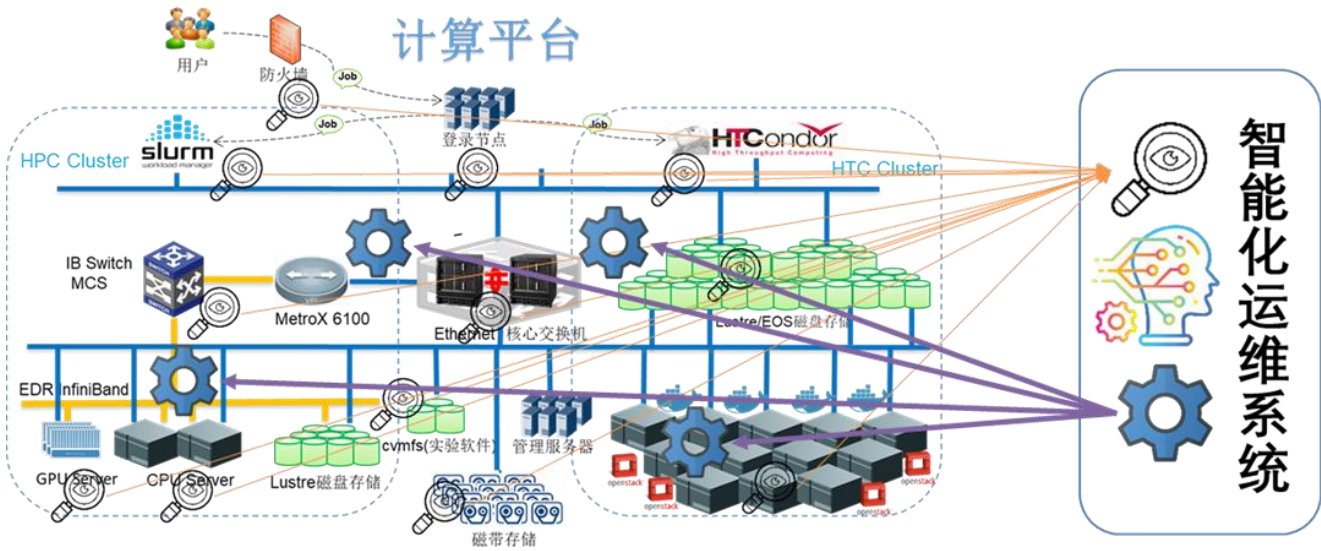


大数据运维平台（1/2）

- 基础数据管理平台（稳定提供“丰富、好用”的数据）
  - 经济性：复用已有运维工具、节省成本
  - 统一性：统一数据汇聚、统一数据共享
  - 可靠性：精确性(过滤脏数据)、完备性(数据源不中断)、及时性(快速预处理)
  - 共享性：多应用、多业务均可通过Data API使用数据
  - 实用性：数据各种维度属性信息确实能支持上层应用的分析
  - 易用性：支持多种模式的数据搜索、提供用户可接受的查询效率和响应时间，易用的查询接口和和灵活多样的展现方式。
  - 安全性：数据敏感信息访问权限控制，数据副本存储和归档存储

大数据运维平台（2/2）

- 平台覆盖
  - 机房动力环境
  - 硬件设备、虚拟机、容器
  - 系统性能、存储性能、网络性能
  - 作业调度、作业数据访问行为、资源管理
  - 安全认证、用户行为、网络攻击



- 平台性能
  - 数据最大采集能力 ~150k doc/s
  - 数据平均处理能力 ~60k doc/s 数据处理延迟 ~2s（Spark Streaming on K8s）
  - 数据最大索引能力 ~180k doc/s （44节点 ES集群 on K8s）
  - 覆盖400+运维数据指标种类
  - 累计存储原始数据435+亿条 & 拥有11k+加工汇总后关键运维指标数据库

数据流支撑（1/4）

- 数据I/O访问可以分为面向MDS的访问和面向OSS的访问
- Lustre Job_stats plugin可以关联访问行为和作业信息
  - Identity can be process name, client node name, process id or any ID which can be read from a process's environment variables
  - We use HT-Condor job ID as job identity `lctl set_param jobid_var=_CONDOR_IHEP_JOB_ID`
  - On login nodes without HT-Condor ID, Lustre will use the default identity as "procname+uid"

数据流支撑（2/4）

- On Metadata server

```
[redacted]# cat /proc/fs/lustre/mdt/lhaasofs-MDT0000/job_stats |grep job_id |wc -l
1529
[redacted]# cat /proc/fs/lustre/mdt/lhaasofs-MDT0000/job_stats |head -n 22
job_stats:
- job_id:          lfs.0
  snapshot_time:   1638074772
  open:            { samples:           0, unit:  reqs }
  close:           { samples:           0, unit:  reqs }
  mknod:           { samples:           0, unit:  reqs }
  link:            { samples:           0, unit:  reqs }
  unlink:          { samples:           0, unit:  reqs }
  mkdir:           { samples:           0, unit:  reqs }
  rmdir:           { samples:           0, unit:  reqs }
  rename:          { samples:           0, unit:  reqs }
  getattr:         { samples:          64, unit:  reqs }
  setattr:         { samples:           0, unit:  reqs }
  getxattr:        { samples:           0, unit:  reqs }
  setxattr:        { samples:           0, unit:  reqs }
  statfs:          { samples:        7234, unit:  reqs }
  sync:            { samples:           0, unit:  reqs }
  samedir_rename:  { samples:           0, unit:  reqs }
  crossdir_rename: { samples:           0, unit:  reqs }
  read_bytes:      { samples:           0, unit:  reqs, min:       0, max:       0, sum:
             0 }
  write_bytes:     { samples:           0, unit:  reqs, min:       0, max:       0, sum:
             0 }
  punch:           { samples:           0, unit:  reqs }
```

  - Total running jobs accessing this device: 1529
  - Number of metadata: 22

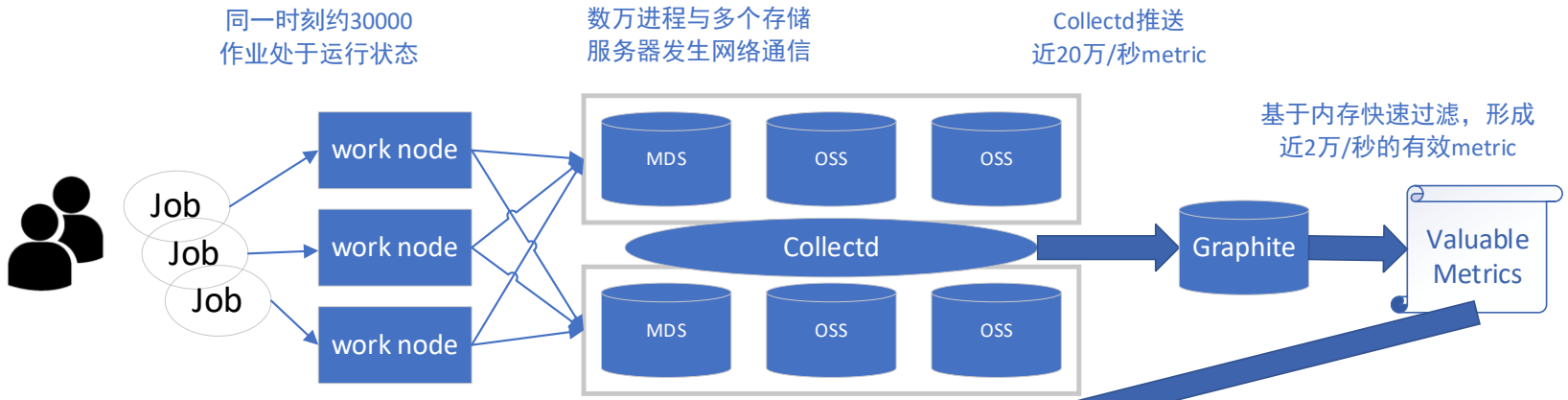
- On data server

```
[redacted]# cat /proc/fs/lustre/obdfilter/lhaasofs-OST0000/job_stats |grep job_id|w
c -l
488
[redacted]# cat /proc/fs/lustre/obdfilter/lhaasofs-OST0000/job_stats |head -n 15
job_stats:
- job_id:          lfs.0
  snapshot_time:   1638075012
  read_bytes:      { samples:           0, unit: bytes, min:       0, max:       0, sum:
             0 }
  write_bytes:     { samples:           0, unit: bytes, min:       0, max:       0, sum:
             0 }
  getattr:         { samples:           0, unit:  reqs }
  setattr:         { samples:           0, unit:  reqs }
  punch:           { samples:           0, unit:  reqs }
  sync:            { samples:           0, unit:  reqs }
  destroy:         { samples:           0, unit:  reqs }
  create:          { samples:           0, unit:  reqs }
  statfs:          { samples:        7236, unit:  reqs }
  get_info:        { samples:           0, unit:  reqs }
  set_info:        { samples:           0, unit:  reqs }
  quotactl:        { samples:           0, unit:  reqs }
```

  - Total running jobs accessing this device: 488
  - Number of data operations: 15

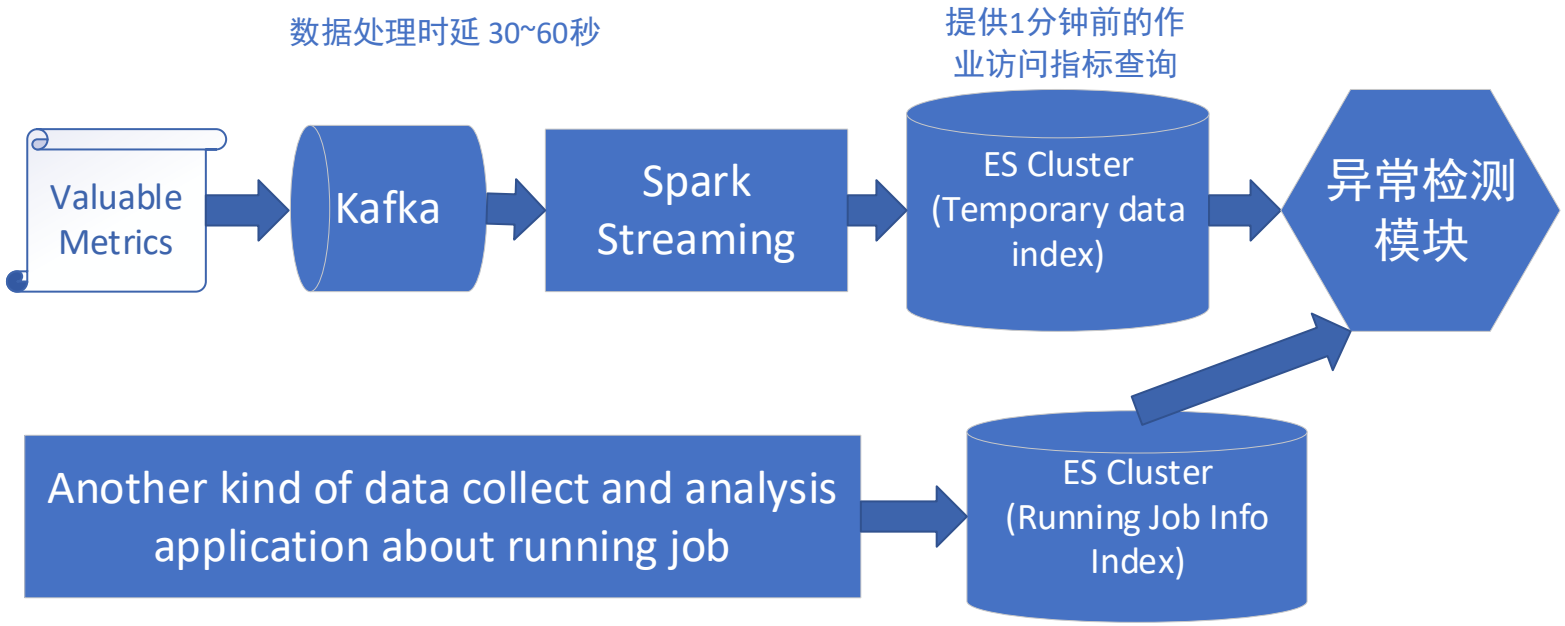
数据流支撑（3/4）

- 19 MDS, 135 OSS @IHEP
- Lustre collectD plugin 用于采集监控指标
  - GitHub - LiXi-storage/barreleye: Lustre Monitoring System




```
collectd.lhoss02_ihep_ac_cn.lhaasofs-OST0007-jobstat_3523795_0.derive-sum_write_bytes,1638022641.000000,98.800000
collectd.lhoss02_ihep_ac_cn.lhaasofs-OST0007-jobstat_35236870_0.derive-max_write_bytes,1638022641.000000,26.433333
collectd.lhoss02_ihep_ac_cn.lhaasofs-OST0007-jobstat_35236870_0.derive-write_samples,1638022641.000000,0.016667
collectd.lhoss02_ihep_ac_cn.lhaasofs-OST0007-jobstat_35237785_0.derive-sum_write_bytes,1638022641.000000,98.900000
collectd.lhoss02_ihep_ac_cn.lhaasofs-OST0007-jobstat_35237785_0.derive-min_write_bytes,1638022641.000000,48.766667
collectd.lhoss02_ihep_ac_cn.lhaasofs-OST0007-jobstat_35237785_0.derive-write_samples,1638022641.000000,0.033333
collectd.lhoss02_ihep_ac_cn.lhaasofs-OST0007-jobstat_35235676_0.derive-sum_write_bytes,1638022641.000000,174659.615435
collectd.lhoss02_ihep_ac_cn.lhaasofs-OST0007-jobstat_35235676_0.derive-write_samples,1638022641.000000,0.216539
```

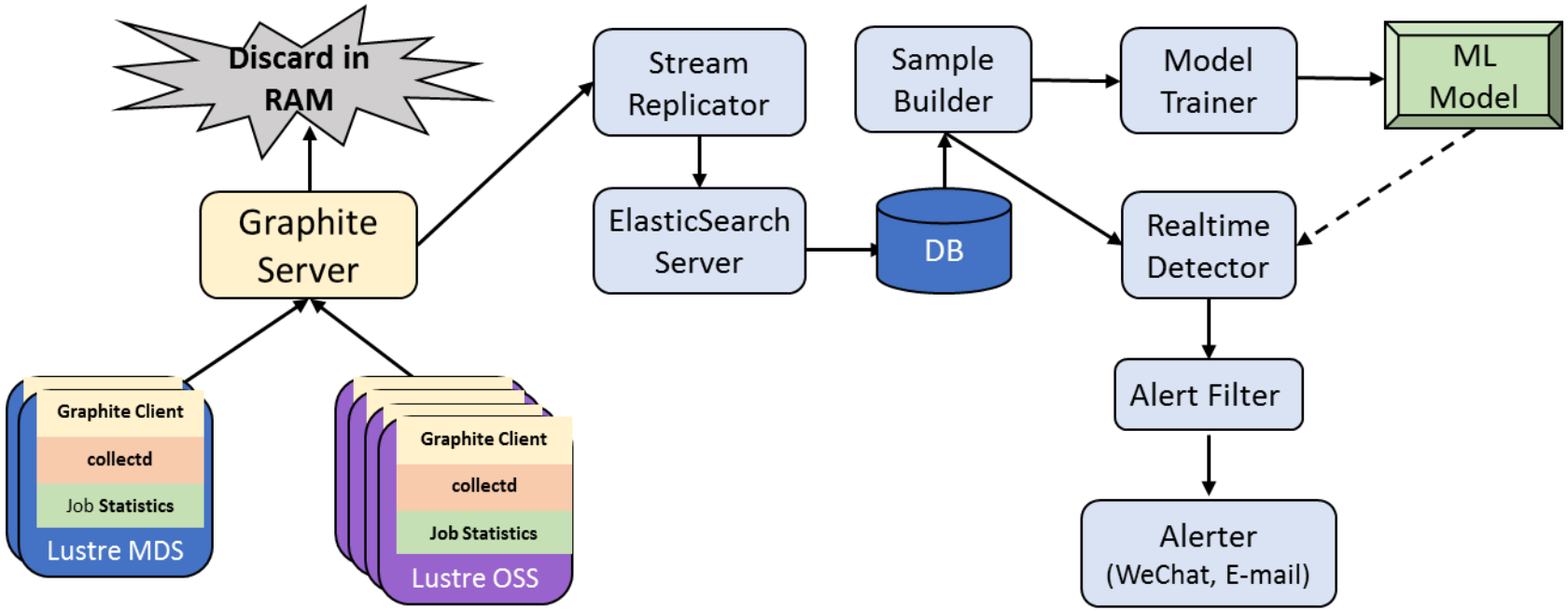
数据流支撑（4/4）



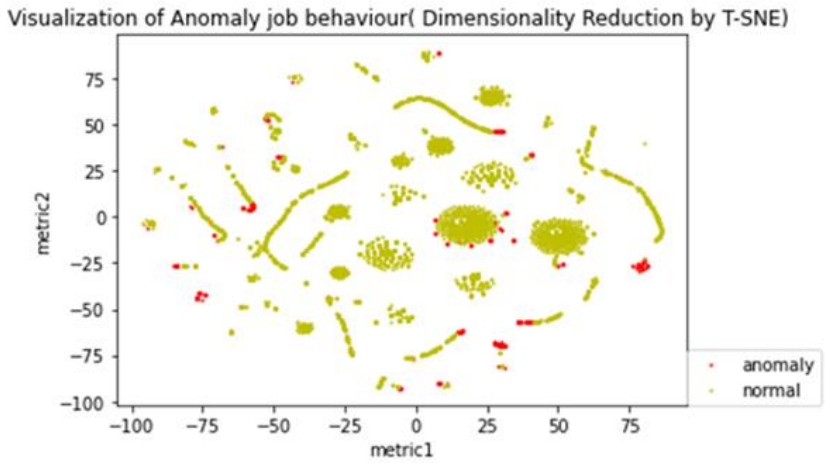

```
subdisk trigger value   timev   operation   fsname
0   OST0000 122951105_0 1.100000    1617465627  write_samples   lhaasofs
1   OST0000 122951129_0 68.266667   1617465627  min_read_bytes  lhaasofs
3   OST0000 123660742_0 68.266667   1617465627  max_read_bytes  lhaasofs
27  OST0006 rsync_12142 4437.333333 1617465715  max_read_bytes  lhaasofs
28  OST0006 rsync_12142 0.750000    1617465715  setattr lhaasofs
45  OST000b 123647284_0 1.050000    1617465715  write_samples   lhaasofs
54  OST000b rsync_12142 68.266667   1617465715  min_read_bytes  lhaasofs
```

```
# grep "Query" 2021-05-09_2021-05-10_*
2021-05-09_2021-05-10_besfs4_gensample.log:Query used: 1972.941382 seconds
2021-05-09_2021-05-10_besfs5_gensample.log:Query used: 1223.382304 seconds
2021-05-09_2021-05-10_besfs6_gensample.log:Query used: 705.211348 seconds
2021-05-09_2021-05-10_bprofs_gensample.log:Query used: 1013.886942 seconds
2021-05-09_2021-05-10_lhaasofs_gensample.log:Query used: 1364.852332 seconds
```

异常检测（1/6）




```
ostoplist = ["read_samples", "write_samples", "destroy",
             "create","get_info", "set_info", "quotactl",
             "sum_read_bytes","sum_write_bytes"]
mdtoplist = ["open", "close", "mknod", "link", "unlink",
             "mkdir","rmdir", "rename", "getattr", "setattr",
             "getxattr", "setxattr", "statfs", "sync",
             "samedir_rename", "crossdir_rename", "punch"]
```



Since we keep the timestamp, we can also made training samples as time series, and train more complicated models such as recurrent neural networks in the future.

异常检测（2/6）

- With above I/O metrics, we can simply make detection by heuristic policies
  - The threshold of problematic job for each operation is difficult to determine
  - can not adapt to changes of hardware and software
  - can not detect new
- Machine learning provides a data driven way to do this task
- In this case, we used a ML algorithm called Isolation Forest
  - Unsupervised, do not need tagged data set for model training
  - linear time complexity, low memory requirement, works well in high dimensional problems that have a large number of irrelevant attributes, and in situations where training sets do not contain any anomalies

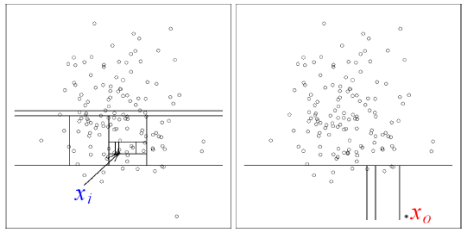
异常检测（3/6）

- Isolation Forest builds a set of proper binary trees from dataset: X = {x1, ..., xn} of n instances from a d-variate distribution.
  - to build one isolation tree, recursively divide X by randomly selecting an attribute q and a split value p, until either: (i) the tree reaches a height limit, (ii) |X| = 1 or (iii) all data in X have the same values.
  - For a new data point, its path length h(x) of a iTree is the the number of edges x traverses an iTree from the root node until the traversal is terminated at an external node.
  - Than its anomaly score is

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}}$$

$$c(n) = 2H(n-1) - (2(n-1)/n)$$

$$H(i) = \ln(i) + 0.5772156649$$

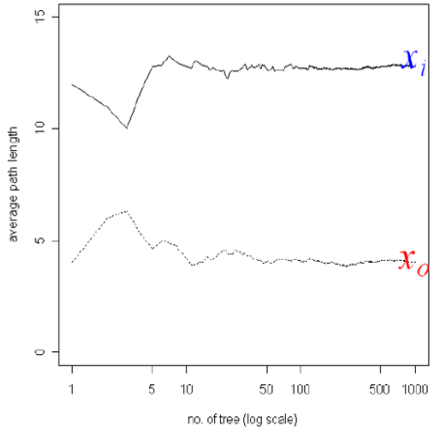


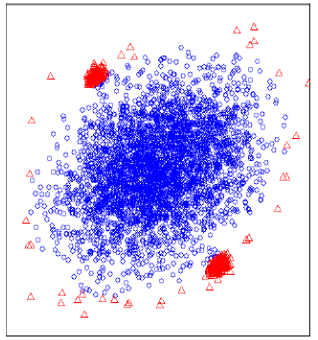
(a) Isolating x_i (b) Isolating x_o

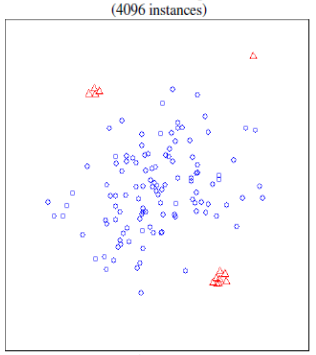
异常检测（4/6）

- Two parameters of Isolation Forest
  - number of trees (t)
  - number of samples used to build a Forest (ψ)
- Experiments show that
  - h(x) converges with a small number of trees
  - Training with sub sampling gives comparable result with training on full dataset
  - Default setting of ψ is *256*
- very effective on large, high dimensional data
- Widely used, included in scikit-learn package

training stage: $O(t\psi \log \psi)$
evaluating stage $O(nt \log \psi)$




(a) Original sample
(4096 instances)


(b) Sub-sample
(128 instances)

异常检测（5/6）

- Hyper parameters of IF
  - n_estimators=100, max_samples=256, contamination=0.1
- Separate models for Metadata and data operations
- Separate models with samples collected last day and week
- New job samples are tagged by cron job every 10 mins
- Since we have detailed information of HT-Condor job in Elasticsearch, we can display details of abnormal jobs
  - uid, submission time, procname etc.,

异常检测（6/6）

- Tagged data samples for last 10 mins are reshuffled for visualization with matplotlib
- The first 5000 samples are reduced to 2-D with PCA and t-SNE

```
In [61]: adf_sorted
```

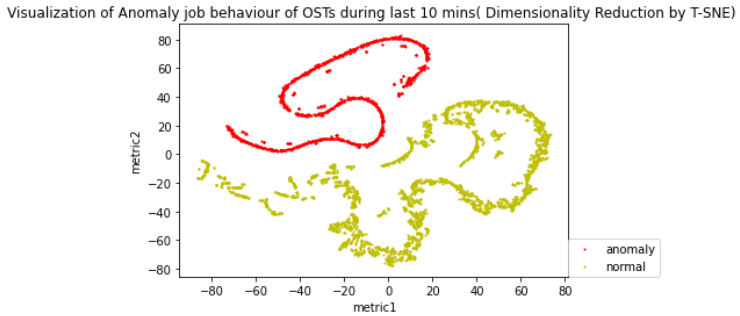
Out[61]:

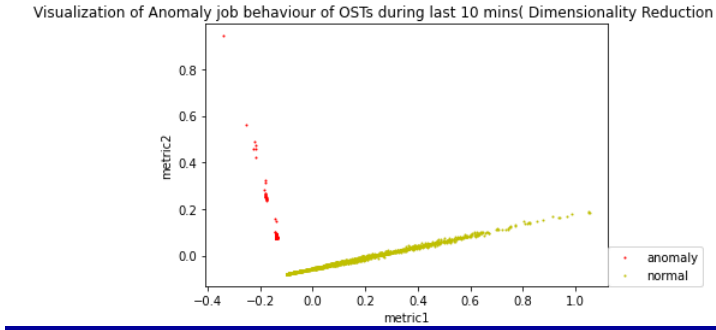
	Unnamed: 0	trigger	fsname	read_samples	write_samples	destroy	create	get_info	set_info	quotactl	sum_read_bytes	sum_write_bytes	sta
18070	3633	45626603_180	bprofs	0.000000	0.016658	0.0	0.0	0.0	0.0	0.0	0.000000	78.475183	16375
22816	3674	45626603_1504	bprofs	0.000000	0.016658	0.0	0.0	0.0	0.0	0.0	0.000000	68.363788	16375
23729	4587	45626605_593	bprofs	0.000000	0.016662	0.0	0.0	0.0	0.0	0.0	0.000000	68.596432	16375
20373	1231	45626603_971	bprofs	0.016656	0.016656	0.0	0.0	0.0	0.0	0.0	68.224225	68.257538	16375
4136	4136	45626604_27	bprofs	0.000000	0.016650	0.0	0.0	0.0	0.0	0.0	0.000000	67.798340	16375
...	...	...	...	...	...	...	...	...	...	...	...	...	
35704	356	45626604_937	bprofs	0.000000	0.016664	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	16375
35481	133	45626603_1479	bprofs	0.000000	0.016664	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	16375
35533	185	45626603_613	bprofs	0.000000	0.016666	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	16375
23911	4769	45626605_694	bprofs	0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	1337.916667	16375
4829	4829	45626605_667	bprofs	0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	1267.866667	16375

1574 rows × 18 columns

```
In [62]: for i in range(0,10):
             jobid=adf_sorted.iloc[i,1].replace("_",".")
             print(jobid)
             job_info_json=get_jobinfo_json(jobid)
             res=search_job("condorpro",job_info_json)
             if (res!=[]):
                 print("%s,%s,%s,%s\n"%(jobid,res["fields"]["execnode"],res["fields"]["user"],res["fields"]["command"]))
```

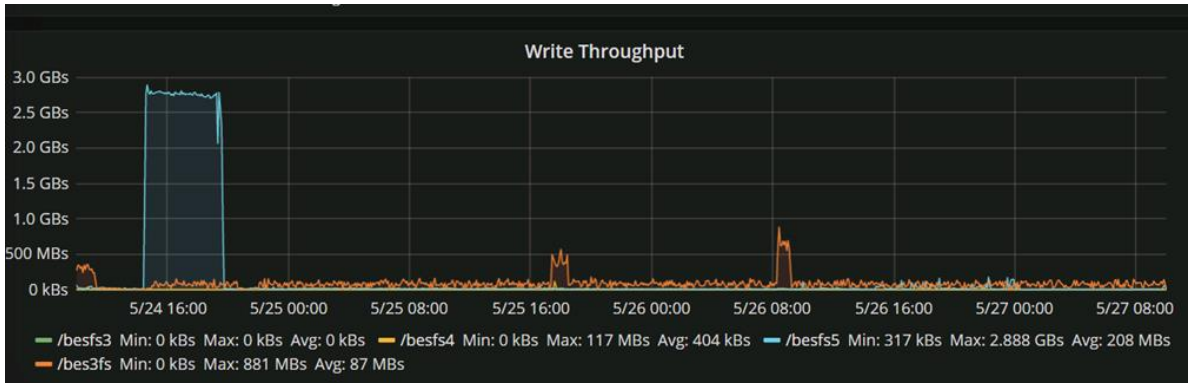
```
45626603.180
job info not ready
45626603.1504
job info not ready
45626605.593
45626605.593,['bws0641.ihep.ac.cn'],['offline'],['boss.exe']

45626603.971
45626603.971,['bws0796.ihep.ac.cn'],['offline'],['boss.exe']

45626604.27
45626604.27,['bws0854.ihep.ac.cn'],['offline'],['boss.exe']

45626603.1284
45626603.1284,['bws0825.ihep.ac.cn'],['offline'],['boss.exe']

45626603.1134
job info not ready
45626603.620
45626603.620,['bws0887.ihep.ac.cn'],['offline'],['boss.exe']
```





案例展示

- 右图所示BESIII存储的监控面板
- 基于前一天的预测模型对该时段用户行为进行异常评分.
- 快速得出异常I/O行为的作业信息



execnode	jobsubstrid	user	jobidsubtime	group
bws0732.ihep.ac.cn	27040249.2		2021-05-24 10:16:18	physics
bws0881.ihep.ac.cn	27003466.1051		2021-05-23 22:27:59	physics
bws0866.ihep.ac.cn	27003466.1255		2021-05-23 22:27:59	physics
lhws169.ihep.ac.cn	27003466.1230		2021-05-23 22:27:59	physics
bws0910.ihep.ac.cn	27040249.81		2021-05-24 10:16:18	physics
...	...		...	...
bws0921.ihep.ac.cn	27003466.468		2021-05-23 22:27:59	physics
bws0910.ihep.ac.cn	27003466.1050		2021-05-23 22:27:59	physics
hxmt010.ihep.ac.cn	27003466.1833		2021-05-23 22:27:59	physics
jnws033.ihep.ac.cn	27003466.1597		2021-05-23 22:27:59	physics
bws0835.ihep.ac.cn	27040249.33		2021-05-24 10:16:18	physics

```
ostdayadf.iloc[:10,[0,1,2,3,4,5,6,7,8,9,10,13,14]]
```

trigger	fsname	read_samples	write_samples	destroy	create	get_info	set_info	quotactl	sum_read_bytes	sum_write_bytes	lastday_score	l
27003466_1255	besfs5	1045.216667	581.616667	0.0	0.0	0.0	0.0	0.0	4.281207e+06	1.770720e+09	0.874030	
27003466_1051	besfs5	1065.733334	603.070160	0.0	0.0	0.0	0.0	0.0	4.365244e+06	1.840973e+09	0.874030	
27003466_139	besfs5	520.402149	562.585470	0.0	0.0	0.0	0.0	0.0	4.240043e+06	8.778144e+08	0.874030	
27003466_825	besfs5	419.933333	611.783333	0.0	0.0	0.0	0.0	0.0	3.428762e+06	5.720865e+08	0.872258	
27003466_1651	besfs5	464.408741	526.808741	0.0	0.0	0.0	0.0	0.0	3.781154e+06	6.993245e+08	0.872258	
27003466_19	besfs5	482.501990	1064.066666	0.0	0.0	0.0	0.0	0.0	3.928747e+06	7.546838e+08	0.872258	
27003466_895	besfs5	398.400000	1346.600000	0.0	0.0	0.0	0.0	0.0	3.250517e+06	1.022176e+09	0.872258	
27003466_1698	besfs5	436.483333	472.370999	0.0	0.0	0.0	0.0	0.0	3.575671e+06	1.235001e+09	0.872258	
27003466_1305	besfs5	352.718972	403.518972	0.0	0.0	0.0	0.0	0.0	2.871842e+06	7.974343e+08	0.872258	
27003466_551	besfs5	378.716667	414.933333	0.0	0.0	0.0	0.0	0.0	3.114325e+06	4.720840e+08	0.872258	

Thanks for your attentions!

谢谢！