Applying AI capabilities to address Operations challenges in ECMWF Products Team

~ Aditya Ahuja, Adithya Niranjan

Mentors: Matthew Manoussakis, Peter Dueben

22nd July

Motivation:

- ECMWF's data services (MARS and Web-API) provide tailored data to Members State, commercial and public users.
- The scale and complexity of the services makes monitoring and troubleshooting an increasing challenge.

Aim of the project:

- To investigate how the application of AI/ML techniques can be used to diagnose problems during operations such as crashes of data servers and to pre-empt developing issues.
- To create predictive insights (e.g., to predict coming problems and to classify root causes etc.) based on predictive analytics.
- To help ECMWF to raise alarms early and take immediate action in order to improve their proactivity.

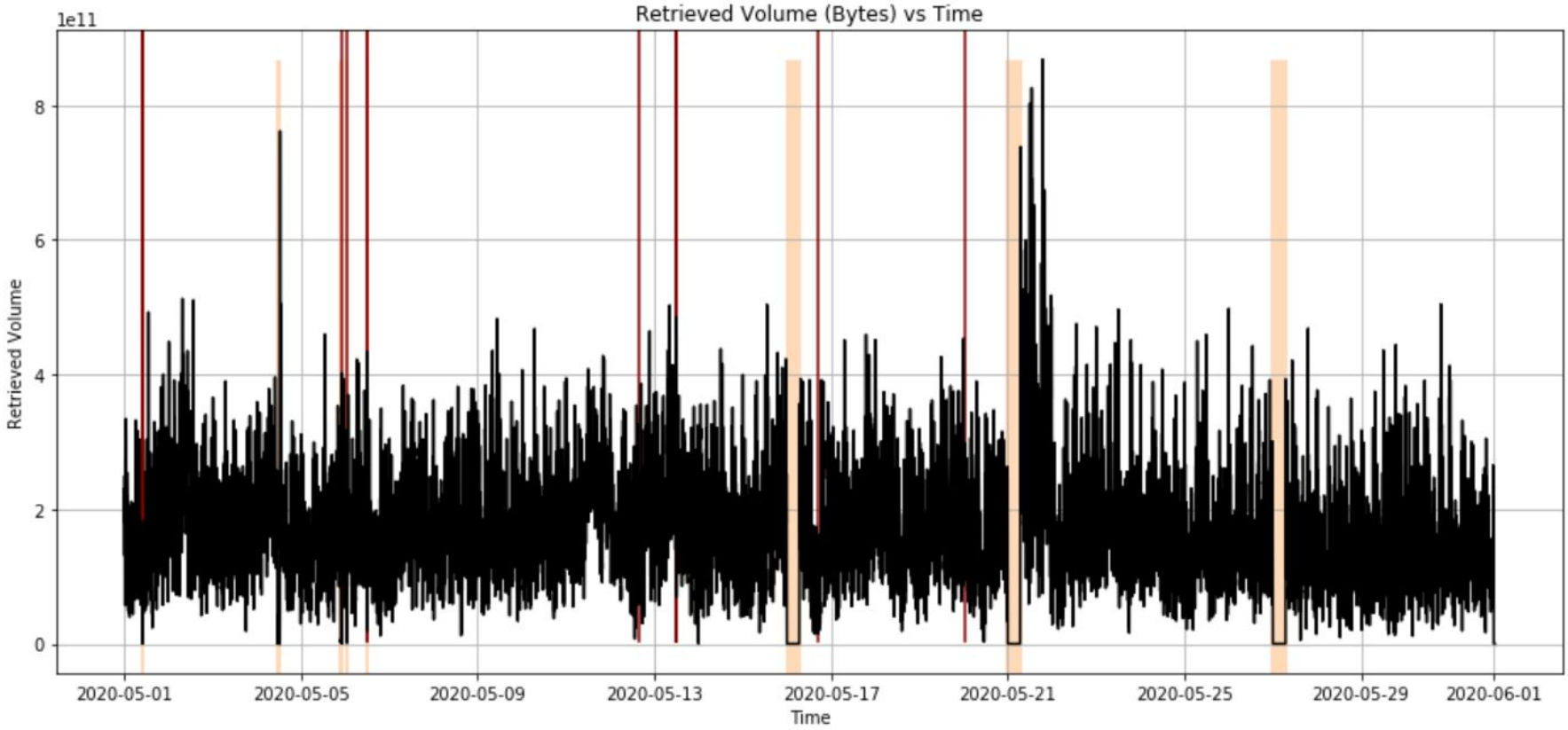
If the project is successful the ideas could be used in other operational services in ECMWF

What does the data look like?

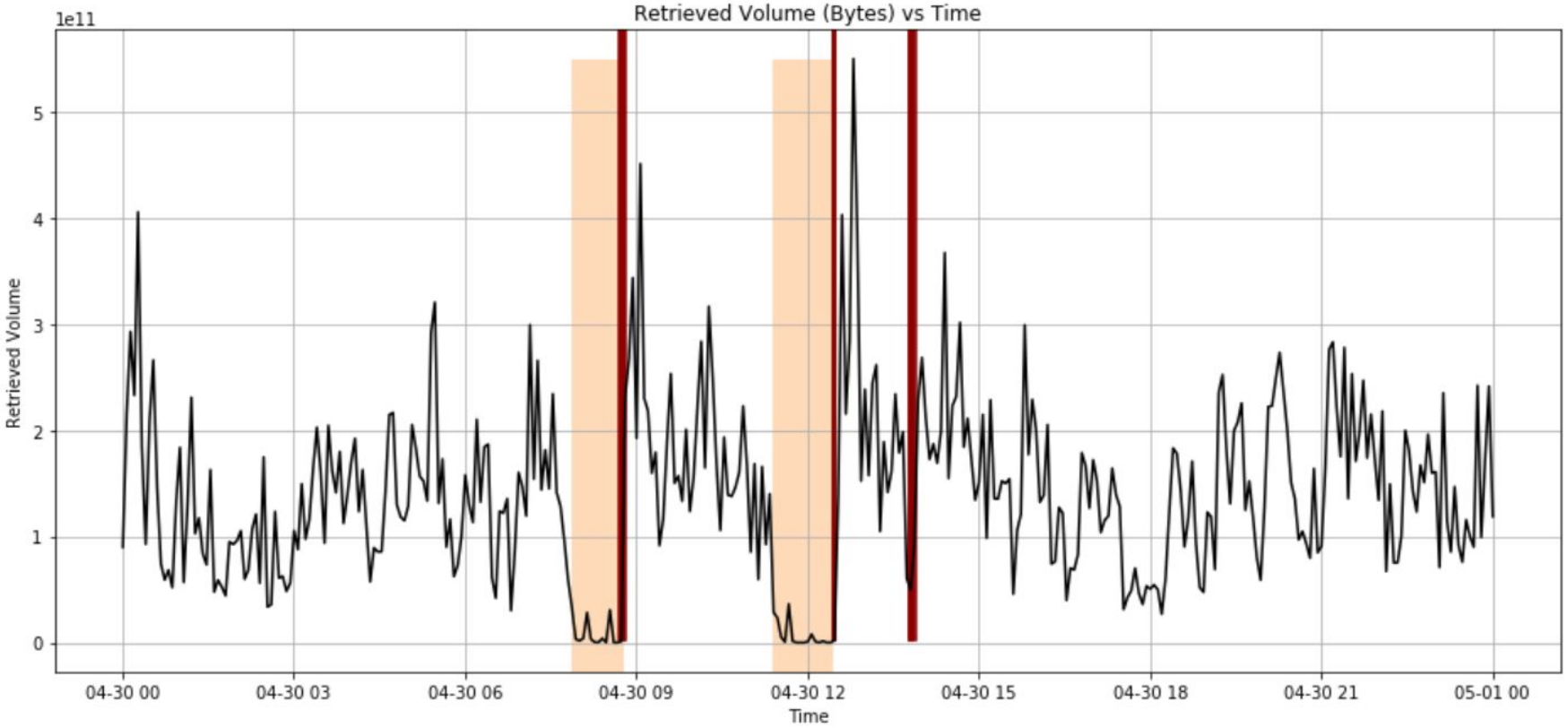
A Sample Log File from MARS

$startdate**='20200521';**$starttime='23:59:59';**$verb**='retrieve';**$version**='20200506';**$application**='mars';
$class**='od';**$type='fc';**$stream**='oper';**$expver**='0001';**$retdate**='20200521';**$age**='0';**$nbdates**='1';**$bon**
ddataset='0';**$reqno**='1';**$multitarget**='new';**$fields**='1';**$database**='fdb5_prod';**$bytes**='13204588';**$writt**
en='13204588';**$interpolated**='0';**$transfertime**='0';**$writetarget**='0';**$cpu**='0';**$elapsed**='0';**$status**='ok';
$stopdate**='20200521';**$stoptime='23:59:59';**$user**='o#m#9';**$category**='l*a*#*e';**$account**='d*e*#o';**$ab**
c='c#*e*o';**$environment**='batch';**$host**='0*l#*#0';**$pid**='40788';**$class**='od';**$type**='fc';**$stream**='oper';**$e**
xpver='0001';**$repres**='gg';**$levtype**='sfc';**$param**='129.128';**$date**='20200521';**$time**='0000';**$step**='0';**$**
domain='g';**$target**='/*#a#*b';**$resol**='1279';**$grid**='o1280';

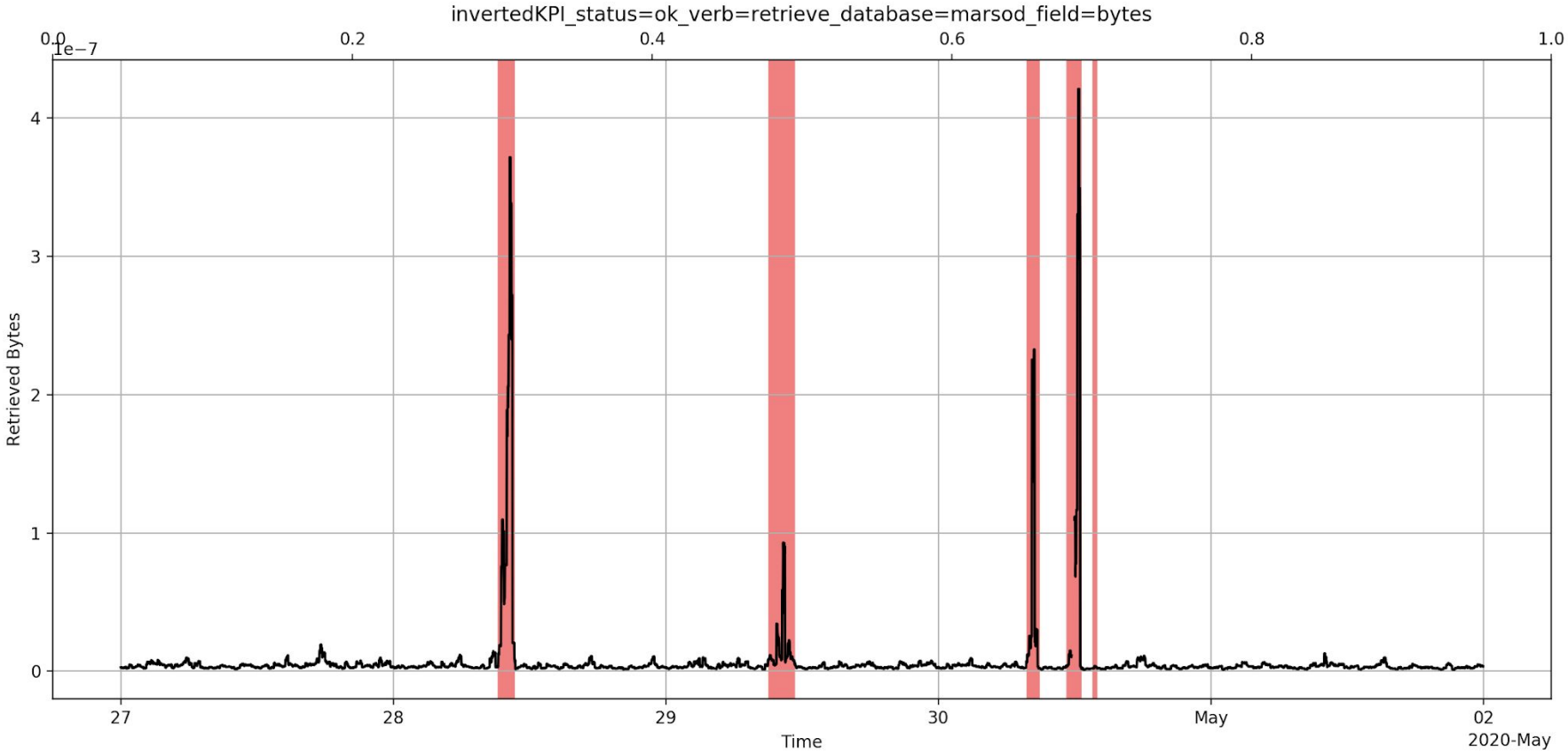
A sample Time Series



A sample Time Series - Zoomed In



A sample KPI


invertedKPI_status=ok_verb=retrieve_database=marsod_field=bytes
1e−7
Retrieved Bytes
Time
2020-May
27
28
29
30
May
02
0.0
0.2
0.4
0.6
0.8
1.0
0
1
2
3
4

Discussion

Anomaly Detection Methods

- Types of anomalies
  - Point anomalies
  - Contextual anomalies
- We need different algorithms to detect both!
- Overall anomaly system will be based on a conditional logic/threshold based combination of algorithms

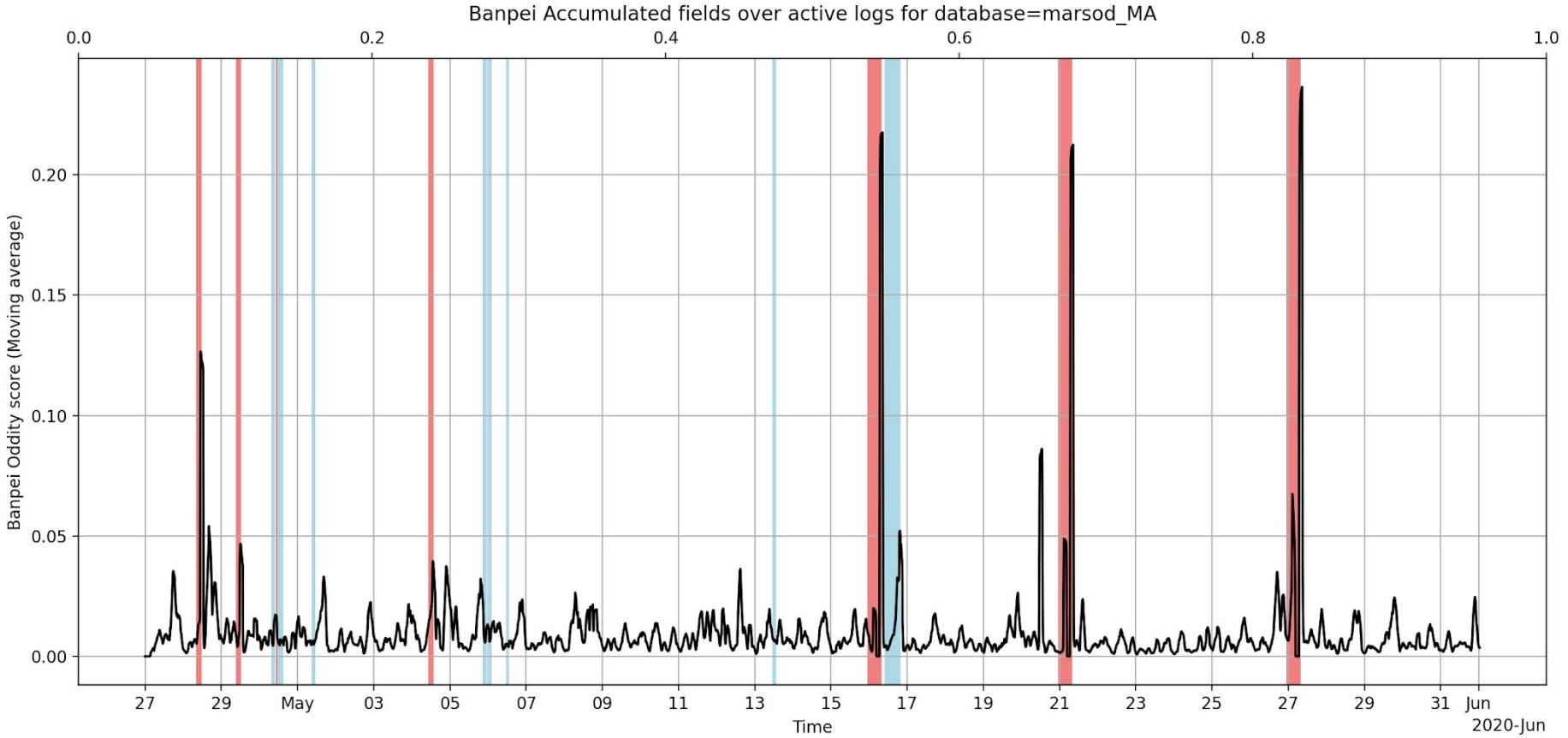
- How these algorithms work -
  - Compare with other time-series windows
  - Reconstruct and compute error
  - Forecast and compute error

Why is this difficult?

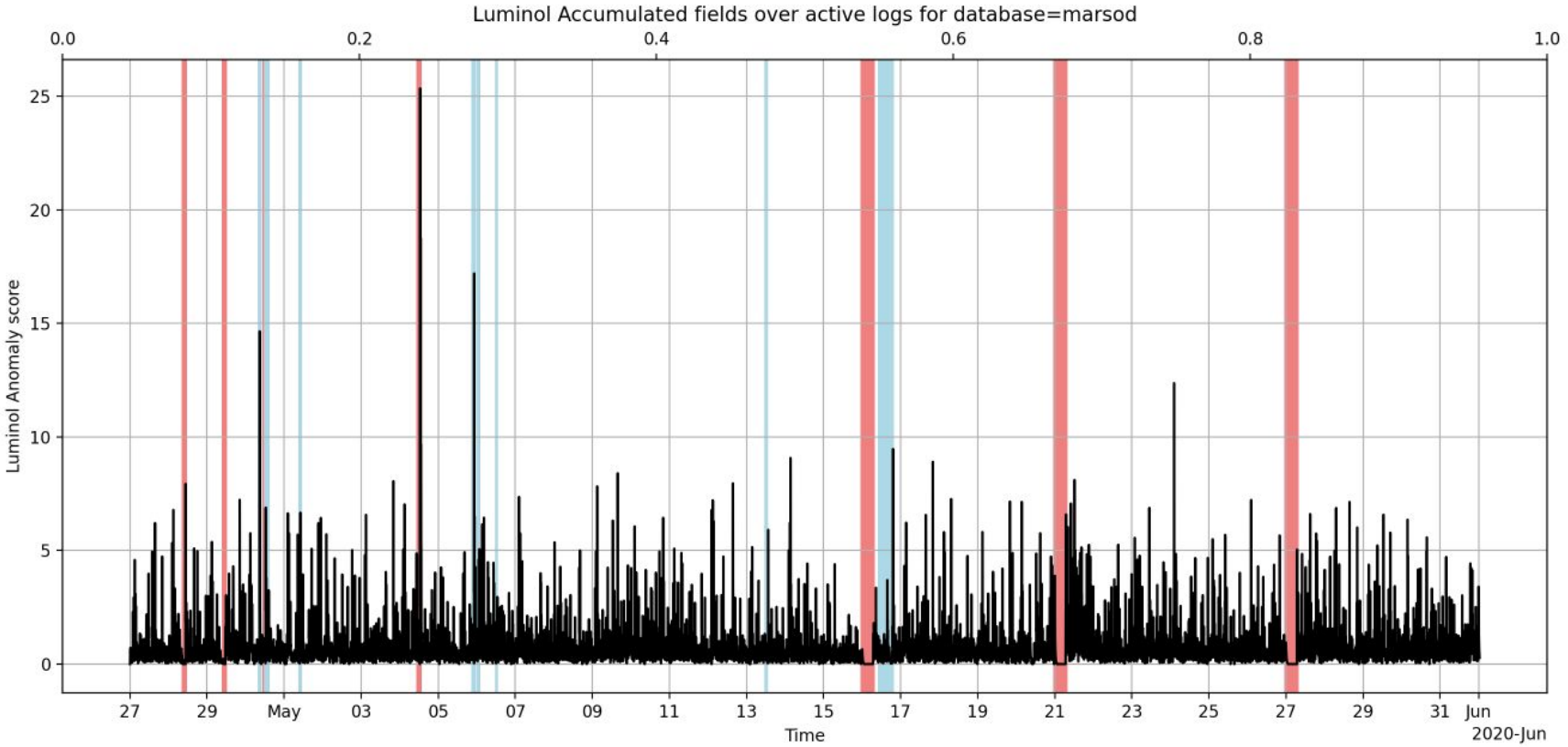
- We only have information about completed logs.
- Anomaly distribution in the dataset is very sparse.
- Not every low-likelihood point is an anomaly!
  - Detecting positive anomalies (like a massive spike) is undesirable (false alarm).
  - Some drops might not be anomalies (planned shutdowns, system maintenance).
- Need to choose the right window length for binning logs.
  - Too small : Lots of false positives generated - Unable to distinguish between temporary sparsity of logs and a genuine downtime.
  - Too large : Lots of false negatives - Genuine downtimes are masked by neighbouring points.

A few results

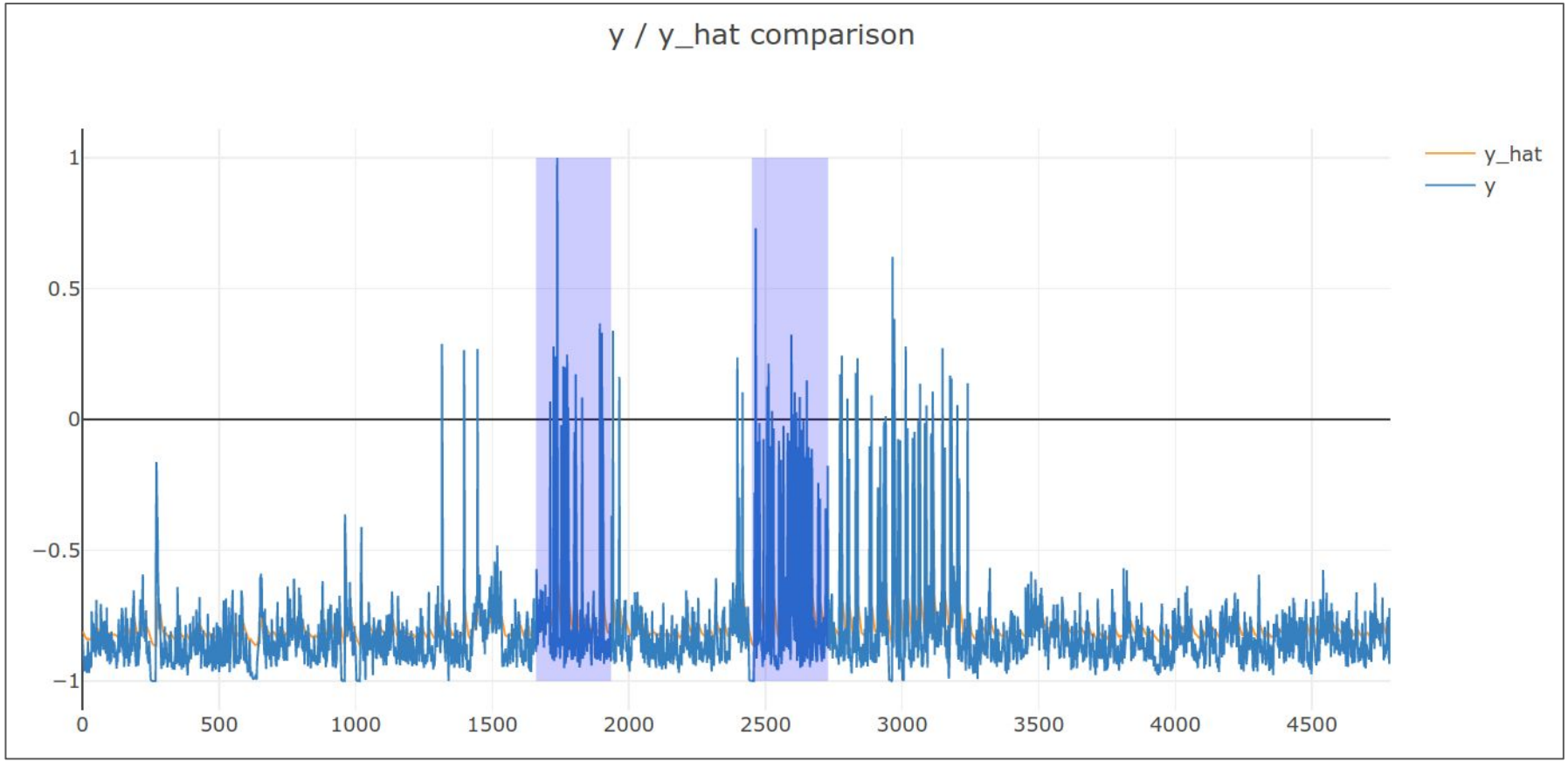
Banpei


Banpei Accumulated fields over active logs for database=marsod_MA
0.0
0.2
0.4
0.6
0.8
1.0
Banpei Oddity score (Moving average)
0.20
0.15
0.10
0.05
0.00
27
29
May
03
05
07
09
11
13
15
17
19
21
23
25
27
29
31
Jun
Time
2020-Jun

Luminol

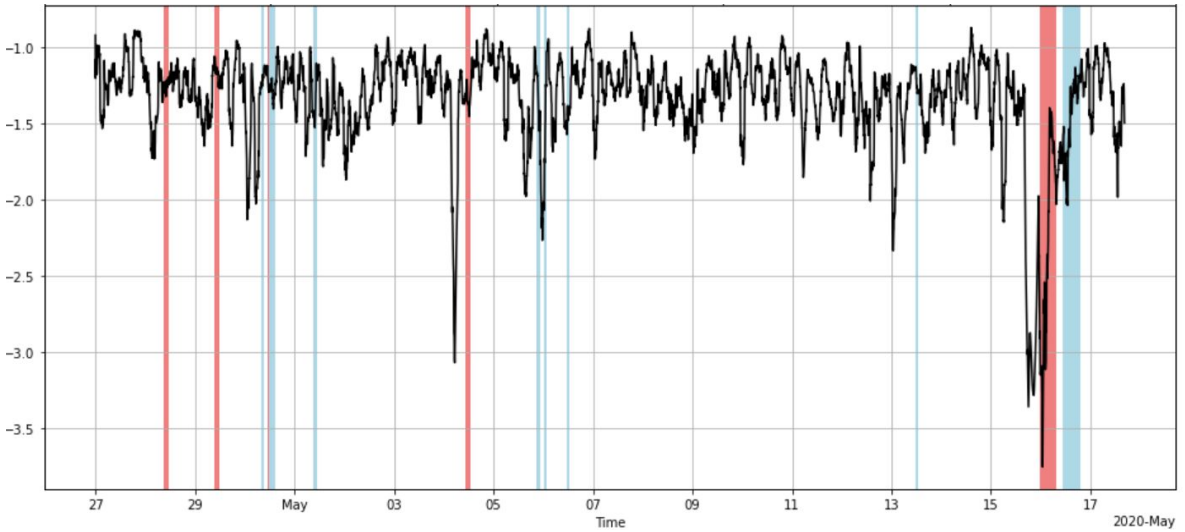

Luminol Accumulated fields over active logs for database=marsod
0.0
0.2
0.4
0.6
0.8
1.0
25
20
15
10
5
0
Luminol Anomaly score
27
29
May
03
05
07
09
11
13
15
17
19
21
23
25
27
29
31
Jun
Time
2020-Jun

Telemanon

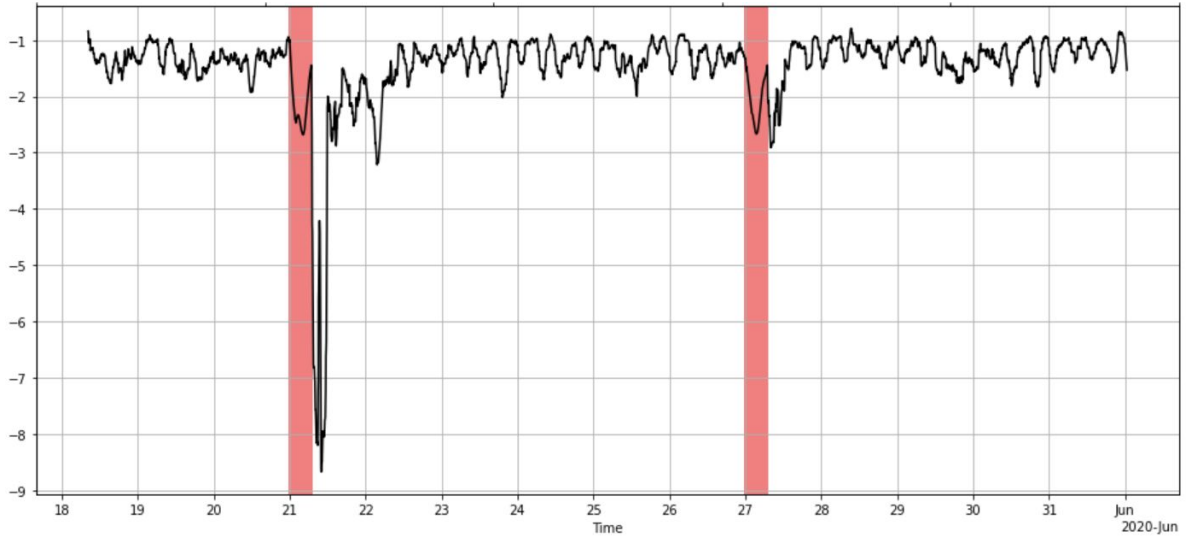

y / y_hat comparison
y_hat
y
1
0.5
0
−0.5
−1
0
500
1000
1500
2000
2500
3000
3500
4000
4500

Donut

Training Set



Validation Set



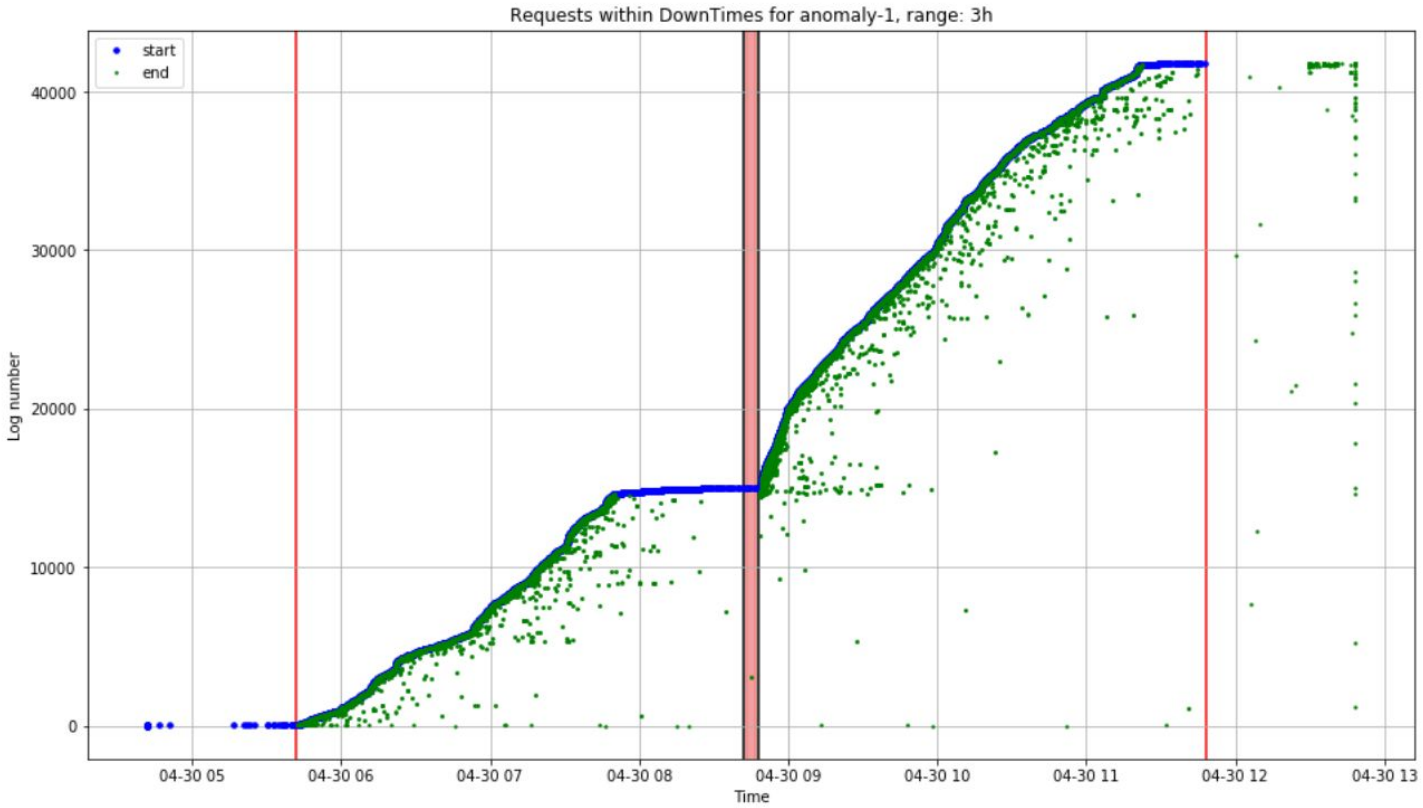
Further Work

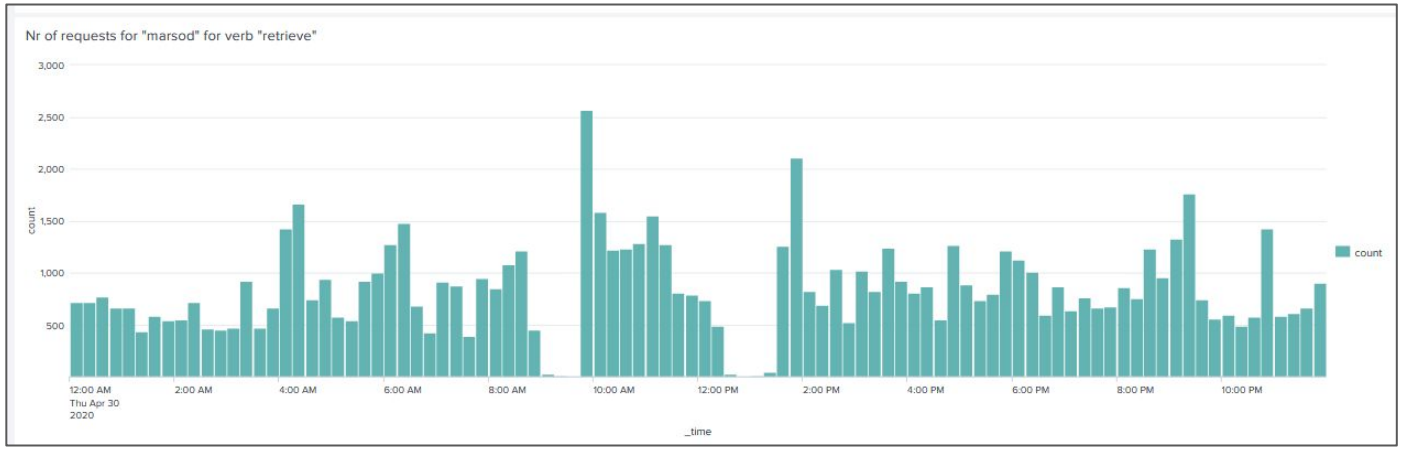
- Implement and train more machine learning methods.
- Tune hyperparameters and thresholds for existing methods.
- Possibly look into metrics which can focus on negative anomalies.
- Integrate anomaly detection with forecasting methods, to help enable near real-time alarm systems.
- Implement a warning system based on a combination/ensemble of anomaly detection methods.
- Package existing code and methods into a directly deployable package.

Questions?

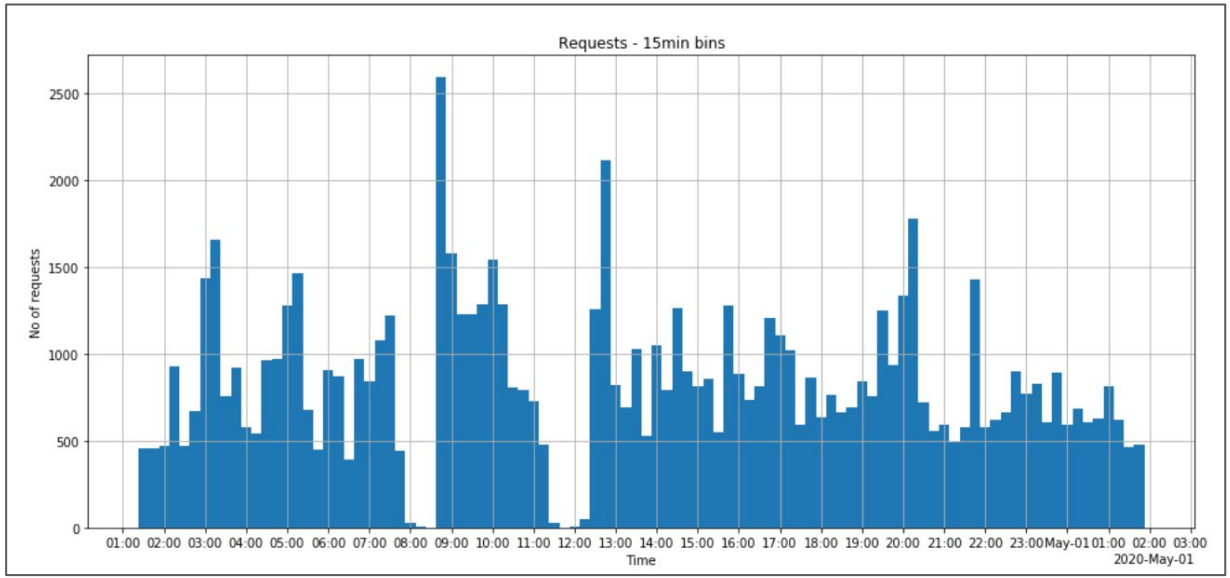
Appendix

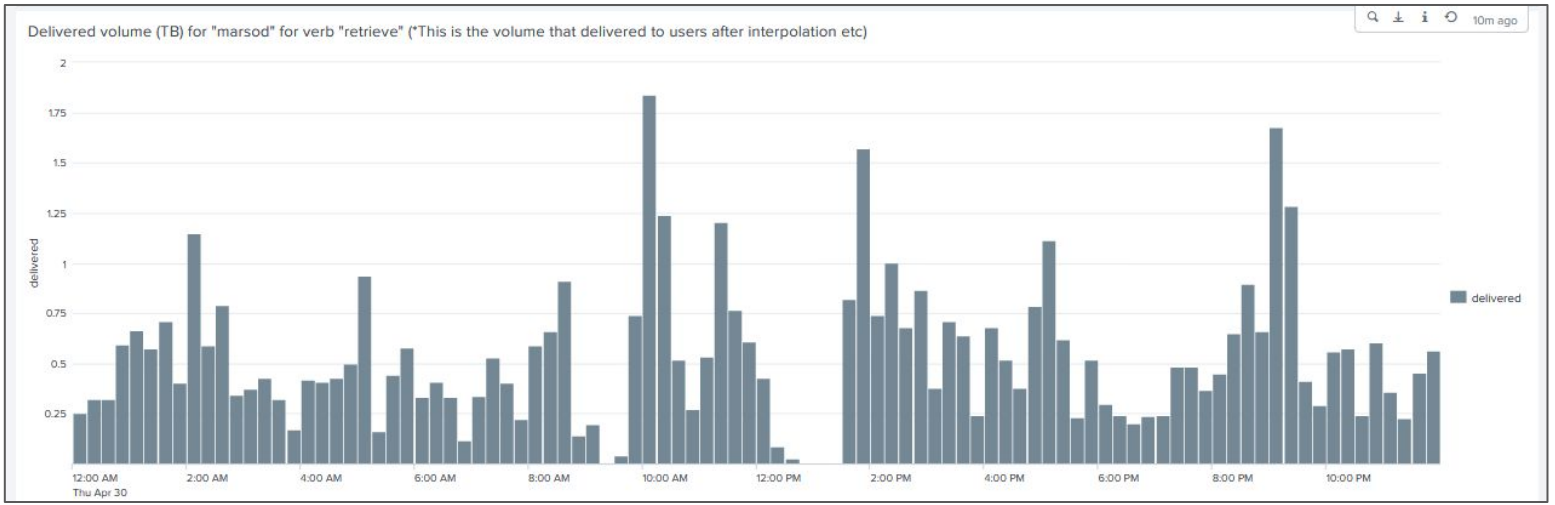
What happens during a downtime?



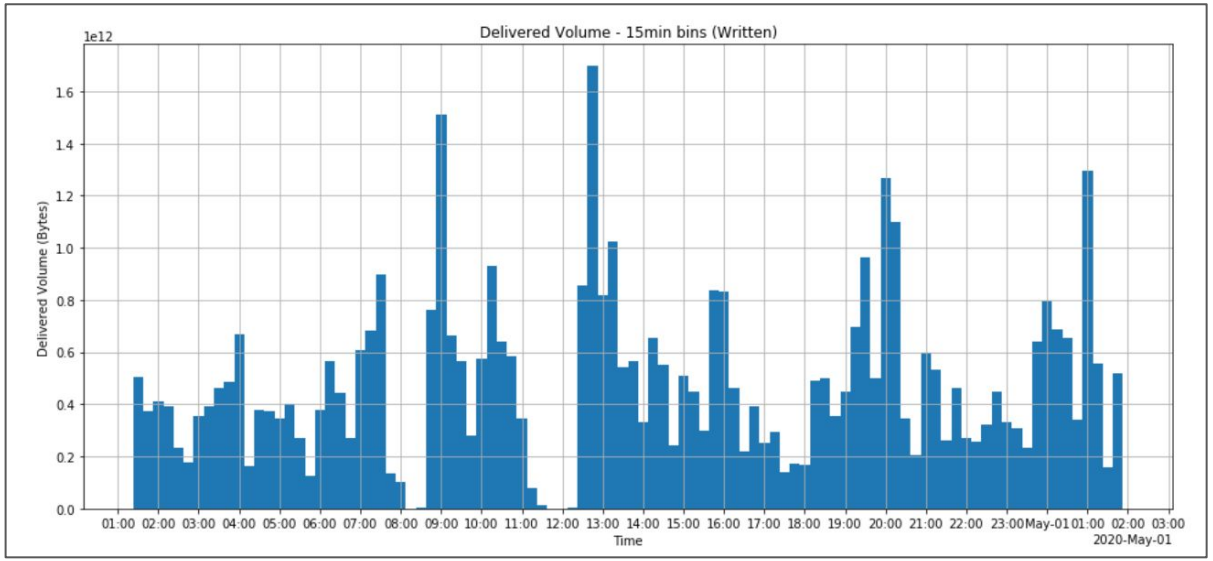


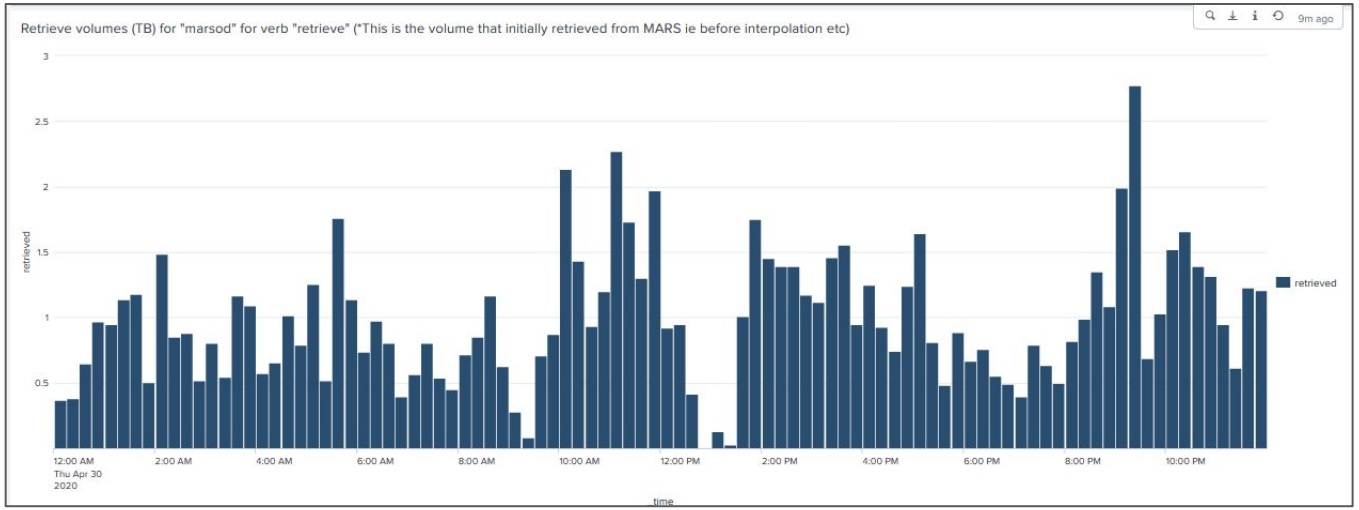
Reconstructing Spunk Logs I





Reconstructing Spunk Logs II





Reconstructing Spunk Logs III