DDM Op Int

Mario.Lassnig@cern.ch
DDM Ops?

- Primary task is to keep our data under control
- 80% of DDM ops is automated — the rest requires manual effort
  - lifetime model updates, exceptions, secondarisation, dark data verification & cleanup
  - rebalancing between sites, comparison of pledged vs available, etc...
- Lost files — automation very tricky
- Transfers not progressing
- Reports for diverse mgmt groups
- Technology evaluations
  - Network tests, Tape tests, ...
- Helping users with their data tasks
- Heroic efforts by Cedric, Dimitrios & Tomas
Why is it so difficult?

- Decision making by DDM Ops people is based on two decades of experience
  - The recipes are in Wikis and Docs, …
  - … but when you need to touch 10 Petabytes of data you want to be really sure what you're doing
  - Simple changes can have large effects on other parts of the distributed computing environment

- We have tried in the past to "encode" this experience
  - Fashionable again due to fancy new technologies

- The expectation is that the more you know about the steady-state of the system
  the better you can react to problems, or anticipate potential improvements

- This is an inventory of things that we tried to ease our operational challenges
Anomaly detection in multi-dimensional time series

- "Can we predict transfer problems from our instrumentation data?"
  - Ship data somewhere else, find different routes, delay, throttle, etc…
- Presumption that a potential steady-state exists
- No discernible difference between NNs and F/SARIMA models
- Approach using wavelets was more promising
  - Find the spatio-temporal dependencies of the signal
- All models beaten by short history approaches
  - Last-n values always a better predictor
- Eventual conclusion
  - → reactive better than predictive
Hybrid simulation

- "Can we evaluate different dataflow models?"
- Flow-based discrete events based on SimGrid
- Analysis of the full 2015 dataflow
- Every component modelled with a technique that was best suited on the available data
  - CART, SVMs, ffNN, …
- Median relative error of full sim at 33%
  - State of the art (GloBeM) before was 73% error!
- Still way too far from anything remotely close to what we were aiming for
Anomaly detection revisited — Deep NN

- Vyom Sharma, GSoC'17
- Deep learn steady-state with LSTM
- Be able to trigger on steady-state violation for alerting
  - Training time 30min on 1mon history
- Good hitrates, but
- On our scale still thousands of wrong anomalies every day
- Can be solved through proper labelling of anomalies
  - But who will manually label thousands of anomalies every day?

https://github.com/vyomshm/DeepAnomaly
Automating labelling?

- Simen Hellesund, QT
- **Evaluation** of classifiers for problematic transfers based on events
- Looked very promising, but no follow-up due to lack of persons
Data popularity

- Thomas Beermann, Thomas Maier
- Predict data **popularity** using NNs and use this to improve job throughput
- Built new Rucio component (C3PO) to exploit this popularity prediction
- Slight but noticeable effect especially on the long tail confirmed via A/B testing
Transfer time estimation — the early days

- Wesley Toler, Summie
- Came from a particular use-case "where should we place Heavy Ion data?"
  - Estimate distance in terms of throughput instead of longitude/latitude
  - Place or rebalance data close in distance to the few HIMEM queues
- Beginning of the "transfer duration estimation" efforts (TTC)
- Used decision trees
  - Good in some cases
  - Bad in others
- Decided we need a more dedicated effort
Transfer time estimation — Next Generation

- Joaquin Bogado
- Client-focused **approach**
  - "How long will it take for my rule to finish?"
- User / WFMS can react accordingly

- Rucio is holding some transfers back to avoid FTS queues to be overfed.
- Not clear yet why this overfeed occurs.

![Graph showing Rucio and FTS queues for CERN to BNL](image)

![Graph showing rule 931337eaba74f628892e4cc7f594c07 [Production_Output]](image)
Where to go from here?

- Lots of varied activities, very dispersed focus, but promising intermediary results!
  - We have suffered always from time-constraints of people (10% here, 20% there, …)

- Very difficult to bring analytics results back into operations
  - We understand a lot of our infrastructure and dataflows much better now
  - But it still doesn't take the load off our operations people!

- Cross-experiment operational effort needs to go beyond analytics
  - Needs to be well-communicated, automated and verifiable
  - Share the recipes and home-made scripts that already exist
  - Start small and identify the boundaries of a problem to solve (e.g., "which data to rebalance")
  - Build trust in the tools and verify them with real workloads

- Can we envision a cross-experiment operational team?