Rucio
& ATLAS

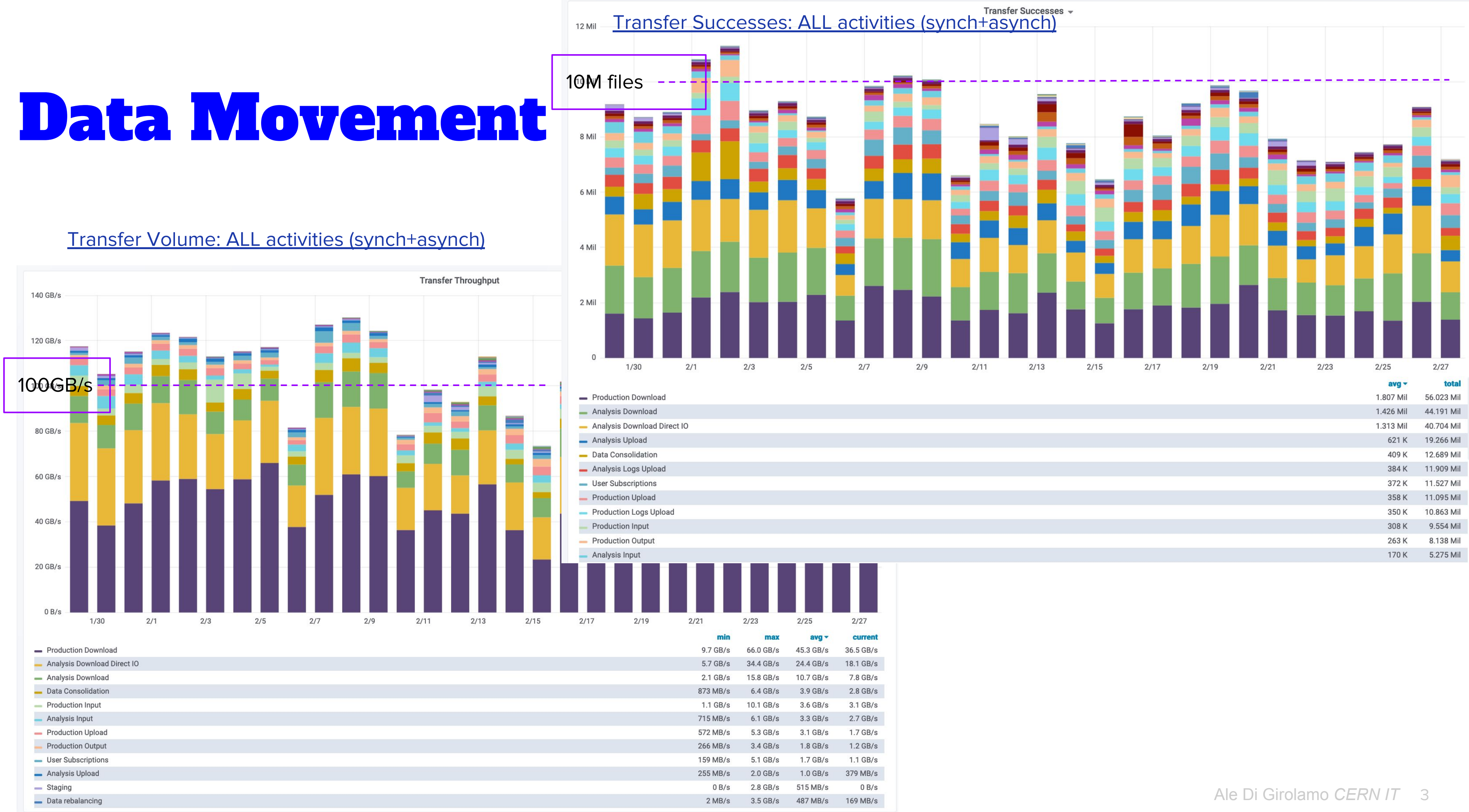
Ale Di Girolamo on behalf of ATLAS

28 February 2019

ATLAS Distributed Data Management

- I am not going to repeat what we already presented ~1y ago in the 1st Rucio Community Workshop ATLAS & Rucio talk
- ATLAS DDM is basically Rucio
  - N.b. ATLAS DDM =! Rucio , Rucio is part of ATLAS DDM
  - not just Rucio as a service, Rucio as ecosystem
  - What is ATLAS DDM that is not Rucio: right now not so much, but we have activities like "Smart Data Placement" and R&D activities like WLCG DOMA ones which are done through Rucio but require ad-hoc investment

Data Movement



Transfer Volume: ALL activities (synch+asynch)

100GB/s

Transfer Throughput

	min	max	avg ▾	current
Production Download	9.7 GB/s	66.0 GB/s	45.3 GB/s	36.5 GB/s
Analysis Download Direct IO	5.7 GB/s	34.4 GB/s	24.4 GB/s	18.1 GB/s
Analysis Download	2.1 GB/s	15.8 GB/s	10.7 GB/s	7.8 GB/s
Data Consolidation	873 MB/s	6.4 GB/s	3.9 GB/s	2.8 GB/s
Production Input	1.1 GB/s	10.1 GB/s	3.6 GB/s	3.1 GB/s
Analysis Input	715 MB/s	6.1 GB/s	3.3 GB/s	2.7 GB/s
Production Upload	572 MB/s	5.3 GB/s	3.1 GB/s	1.7 GB/s
Production Output	266 MB/s	3.4 GB/s	1.8 GB/s	1.2 GB/s
User Subscriptions	159 MB/s	5.1 GB/s	1.7 GB/s	1.1 GB/s
Analysis Upload	255 MB/s	2.0 GB/s	1.0 GB/s	379 MB/s
Staging	0 B/s	2.8 GB/s	515 MB/s	0 B/s
Data rebalancing	2 MB/s	3.5 GB/s	487 MB/s	169 MB/s

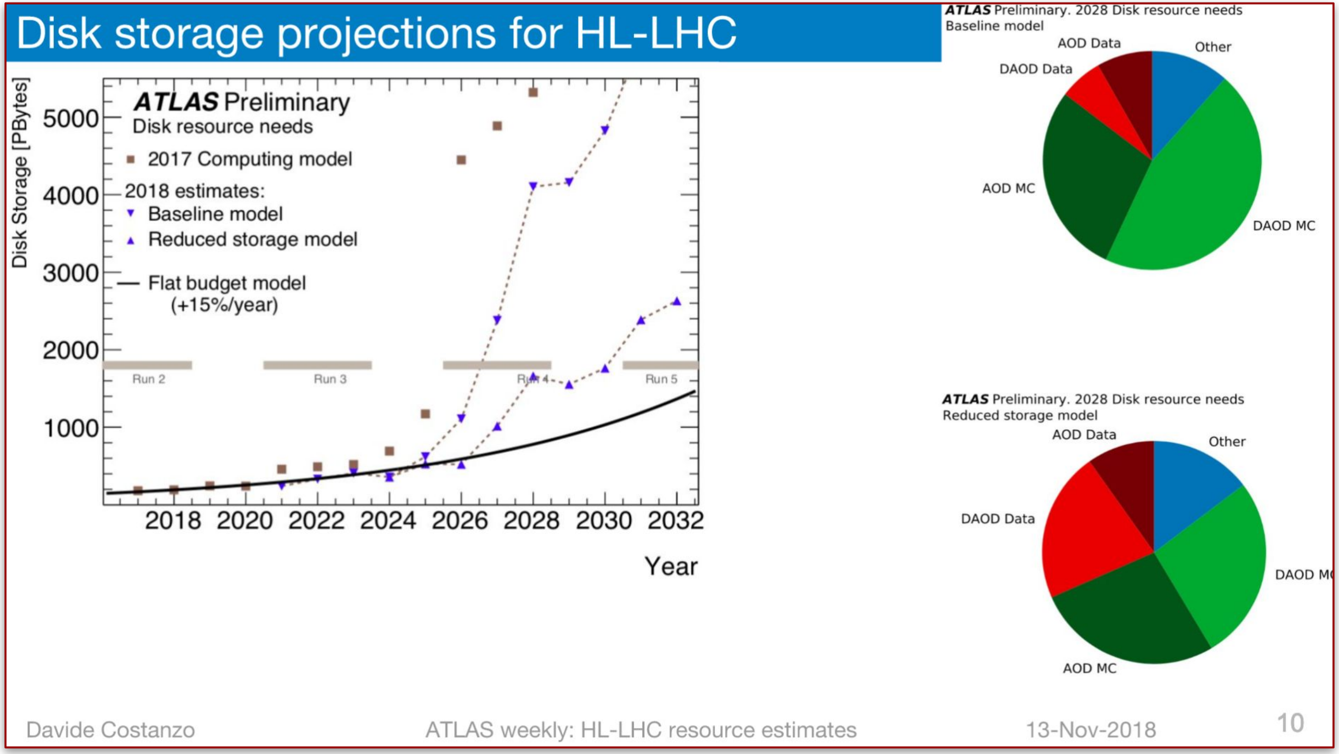
Transfer Successes: ALL activities (synch+asynch)

10M files

Transfer Successes ▾

	avg ▾	total
Production Download	1.807 Mil	56.023 Mil
Analysis Download	1.426 Mil	44.191 Mil
Analysis Download Direct IO	1.313 Mil	40.704 Mil
Analysis Upload	621 K	19.266 Mil
Data Consolidation	409 K	12.689 Mil
Analysis Logs Upload	384 K	11.909 Mil
User Subscriptions	372 K	11.527 Mil
Production Upload	358 K	11.095 Mil
Production Logs Upload	350 K	10.863 Mil
Production Input	308 K	9.554 Mil
Production Output	263 K	8.138 Mil
Analysis Input	170 K	5.275 Mil

ATLAS DDM - the future



Disk storage projections for HL-LHC

Davide Costanzo ATLAS weekly: HL-LHC resource estimates 13-Nov-2018 10

- Run-3: Rucio
- Run-4: Rucio
- *N.b.: Not without work to satisfy the future needs*
- FCC: a bit too far right now to comment on it today.

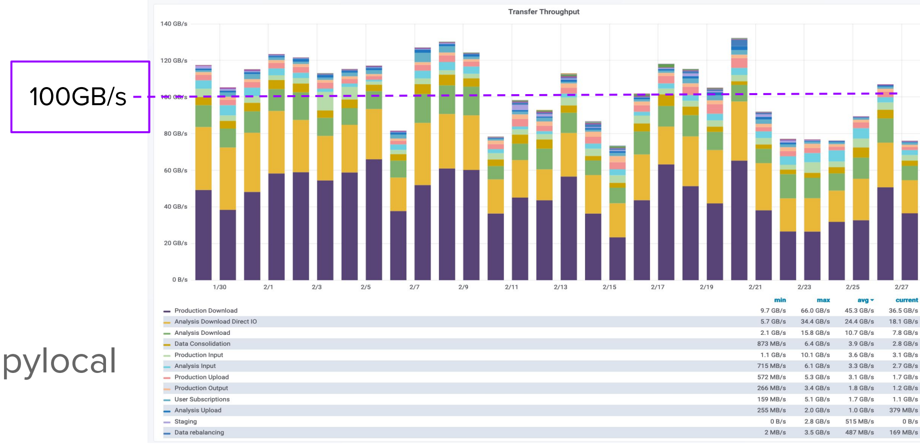
Flexibility and Rucio

- Rucio is flexible in many ways:
  - For instance, you can define in Rucio RSE a bit as you want, you can define your naming convention, you can plugin new transfer methods….
- Flexibility is paramount: good, great, super!
- … when there are lots of possibilities, guidance might be needed
  - I mean that it's good to be flexible, but "we" ("we" to be defined who) need to give guidance, assistance, consultancy to the various collaboration embracing Rucio to do things properly from the beginning, without overdoing but without oversimplifying
  - "Consultancy" is for me the key here
- Rucio team need to take this consultancy aspect into their own responsibilities
  - Cause they have experience on how painful can be when you have too many attributes (or not enough) to do what you want to do
  - And they also know now that often we physicists are very bad in defining what we want from the beginning….
  - Rucio team for me is really an enlarged team of experts coding Rucio and using Rucio

An example of feature req to Rucio

- The Archive business
- Some of our "crazy" guys asked Rucio:
  - We want to be able to use zip files (archive, files concatenated).
  - These zip file should be "transparent" to the users (panda, real users, everyone)
  - Why? Several reasons: cause some workflows produce zillions of small files that create troubles when we move them around, for tape the bigger might be the better, ...
- First estimate:
  - few months of work
- Reality: 1+ year of work
  - Several iterations
  - Requirements changing, evolving, new things discovered that needs fixes to make them working
- Moral of the story:
  - things are complicated.
  - The Rucio development and deployment cycle is solid but somehow requires that Rucio team would take onboard also beta testing (in addition to alpha testing).

Rucio Mover



	min	max	avg ▾	current
▬ Production Download	9.7 GB/s	66.0 GB/s	45.3 GB/s	36.5 GB/s
▬ Analysis Download Direct IO	5.7 GB/s	34.4 GB/s	24.4 GB/s	18.1 GB/s
▬ Analysis Download	2.1 GB/s	15.8 GB/s	10.7 GB/s	7.8 GB/s
▬ Data Consolidation	873 MB/s	6.4 GB/s	3.9 GB/s	2.8 GB/s
▬ Production Input	1.1 GB/s	10.1 GB/s	3.6 GB/s	3.1 GB/s
▬ Analysis Input	715 MB/s	6.1 GB/s	3.3 GB/s	2.7 GB/s
▬ Production Upload	572 MB/s	5.3 GB/s	3.1 GB/s	1.7 GB/s
▬ Production Output	266 MB/s	3.4 GB/s	1.8 GB/s	1.2 GB/s
▬ User Subscriptions	159 MB/s	5.1 GB/s	1.7 GB/s	1.1 GB/s
▬ Analysis Upload	255 MB/s	2.0 GB/s	1.0 GB/s	379 MB/s
▬ Staging	0 B/s	2.8 GB/s	515 MB/s	0 B/s
▬ Data rebalancing	2 MB/s	3.5 GB/s	487 MB/s	169 MB/s

Transfer Volume: ALL activities (synch+asynch)

- WN - Storage data movement
  - jobs: reading/writing, WAN/LAN, directI/O/copylocal
    - Asynch data transfers (FTS): 1-2PB/day
    - Synch: 6-7PB/day
- Strategy:
  - Clean and clear interfaces between WFMS and DDM
  - Very much together, very close, but clear separation of responsibility and definition of interface
- Rucio movers
  - Since 2+ years we are working to have these Data Management parts managed by Rucio
  - replacing the old (solid, working since years) movers with Rucio based movers
- Moral - similarly to previous slide: things are complicated
  - Slightly different: we knew this was going to be very complicated work: but also very important
  - For "new Rucio embracers": don't go for the quick and dirty "I'll do this script now, we will do clean things later".

Topology

Disclaimer: I'm mixing hats here: I'm also working on the CRIC project - the AGIS evolution, experiment independent

- Working with Rucio team to make Rucio even more "user- friendly"
- ATLAS invested few years ago in a central topology information system, AGIS, which allow to define RSEs in pre-defined agreed ways
  - Workflow management, ddm, monitoring, ….: taking info needed for configs from one single place, AGIS. Different views of the same information or different level of info for the various parts, e.g. storage protocols vs Computing Elements gatekeepers and other attributes.
- Working with Rucio team to have Rucio topology configuration loadable from human readable JSONs
  - Which can come from frameworks like CRIC but also can be edited and created in whatever other ways to let other experiments the flexibility to do what they want (take the systems they want)
  - For instance: 7 sites: don't bother, write by hand a JSON and it's done. 10-15 sites? Mmm, not clear to me anymore…. 20-30+ sites? configs become a nightmare. You might think today is not an issue, you will suffer from year 2 for ever

More intelligence

Honestly, I didn't like the fact that Rucio is a donkey... Actually that ATLAS was moving to a DDM system which was a donkey... [but I changed my mind!]

- Too often we fail transfers because of timeouts
  - And the we retry, and we retry, and we retry....
- Multiple sources
  - We don't have so many multiple replicas, but the ones we have, can we exploit them all? Yes with "new" protocols
- Tighten interfaces with WFMS
  - If I have files in Site A but I can't run there, I can run in Site B, C or D, where do I have the minimal overhead?
- Usage of caches, federations:
  - We need to move to a more sustainable storage organization: today ATLAS uses 130 sites (with storage).
  - We are working on several R&Ds
  - We need our DDM system capable of evaluating performance and propose solutions

Distributed Analysis & Rucio

Distributed Analysis & Rucio

- I was tempted to let just the empty slide
- People tend to say only bad things
- DA folks are actually saying that docs are very good, users are happy (they mean they are not unhappy), and only ~10% of the problems/request is Rucio related
    - For me this is very good
    - Did not come for free, I do clearly remember the first cli documentation!
    - A lot of efforts from Rucio and Distributed Analysis team to make the bootstrap of newcomers as smooth as possible
    - There is still space for more intelligence, sites get broken and this still create troubles….

Bonus req: Full lifecycle of files (dataset)

Logs on data movements and access are not (yet) coherently organized

- Traces, pure JSON, for data access
- Files/datasets creation, transfer queued/submitted/done/failed, deletion are in AVRO format
- Not impossible to join the two, but why not try to reorganize them better?
- Critical (might be superuseful) for R&D studies

RucioS

Now that (if) Rucio has become (is becoming) the Data Management system of various experiments, what can we do together (to all gain)?

Crazy ideas: DDM Ops together!

- I know that reading the title people might have already got an heart attack
  - I am bold but not completely out of mind
  - Stay with me, I wrote this title to grab your attention

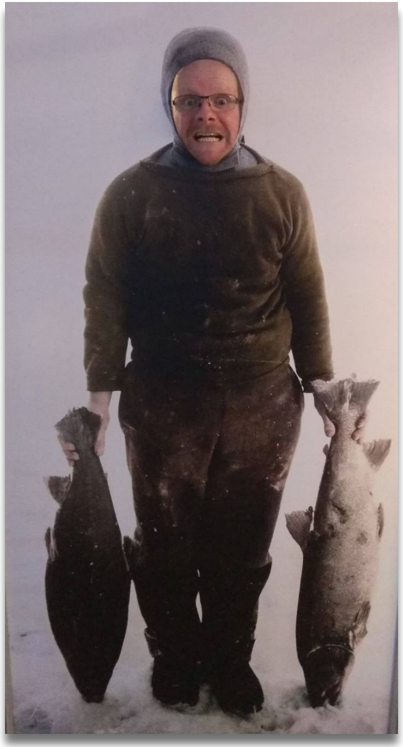
- Reality is that we share infrastructure: sites, network...
- We could do quite a lot together
  - Sharing information: problems of sites, usage of networks...
  - Common monitoring, accounting, procedures....
  - Operational Intelligence:
    - Automating computing operations, e.g. automatic ticketing of sites, ML algorithms to suggest shifters most probable cause of the problem, autodiscovery of network paths degraded...
  - ....

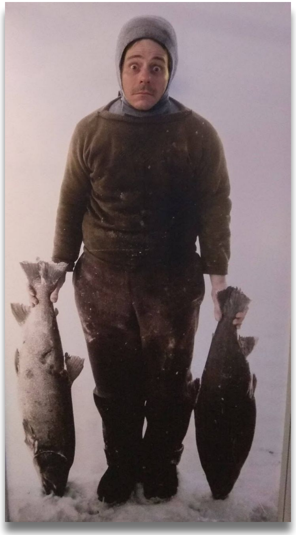
We all live on the planet earth

Some conclusions

- Data Management is complex business
  - Huge challenges in front of us.
- Need great architects, devs, devops, and consultants
  - And strong will of doing things together, reusing and improving what is around (if possible)
- We have and we will have substantials requests for Rucio
  - Work won't lack
- We can (and we *should*) do quite a lot together
  - to save brainpower
  - to improve even more the Rucio scientific data management ecosystem that we use.

Thanks! Oslo is inspiring





Sometimes we wonder: why are we doing this? … I went to the FRAM and Viking museum: sometimes you just need someone that is willing to do the impossible, to make it possible!