Rucio

Scientific Data Management

Martin Barisits & Mario Lassnig
on behalf of the Rucio team

RUCIO
SCIENTIFIC DATA MANAGEMENT

ATLAS
EXPERIMENT

About

Rucio in a nutshell

- Developed by the High-Energy Physics experiment ATLAS
- Rucio provides a complete and generic scientific data management service
  - Data can be scientific observations, measurements, objects, events, images saved in files
  - Facilities can be distributed at multiple locations belonging to different administrative domains
  - Designed with more than 10 years of operational experience in large-scale data management!
- Rucio manages multi-location data in a distributed environment
  - Creation, location, transfer, and deletion of replicas of data
  - Orchestration according to both low-level and high-level driven data management policies (usage policies, access control, and data lifetime)
- Rucio (arXiv) is open source and available under Apache 2.0 license
- Makes use of established open source tools

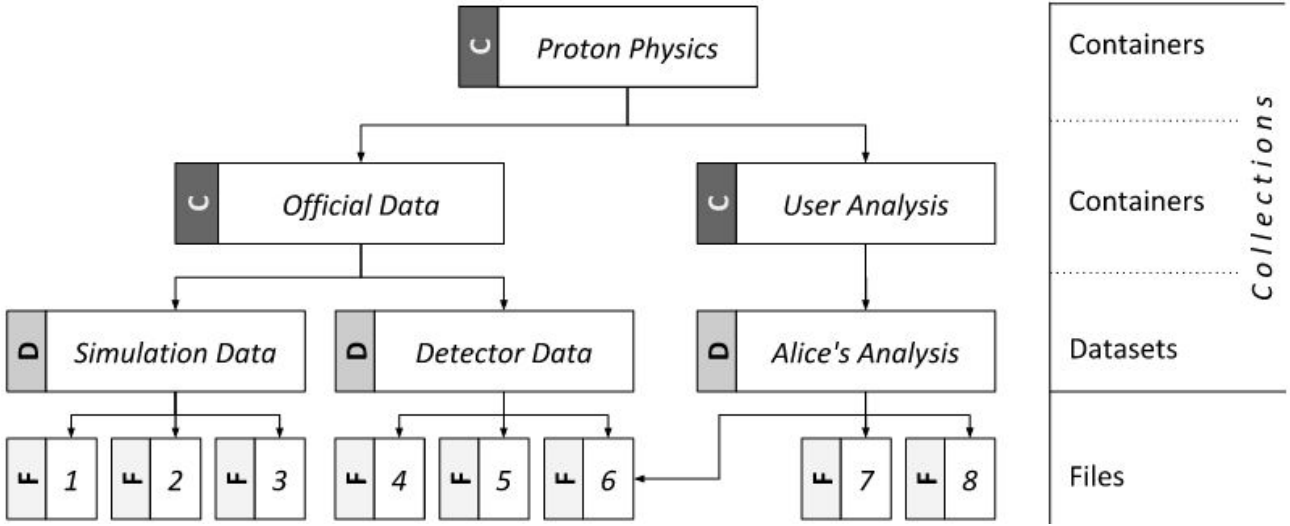
Rucio in a nutshell

- Features can be enabled selectively
  - Namespace management
  - Data transfers and tape archival
  - Web and CLI interfaces to discover, organise, upload, download and access data
  - Extensive monitoring
  - Powerful policy and dataflow engine
  - Corruption identification and data recovery
  - Popularity based data replication
  - …

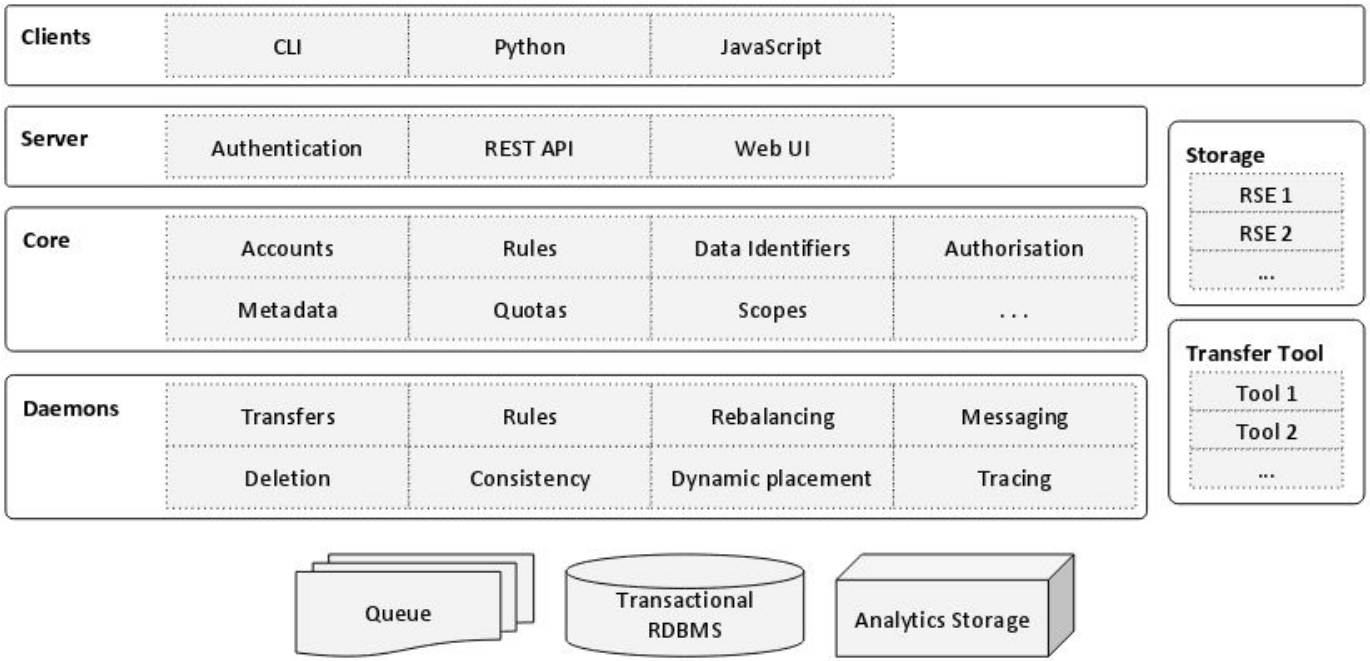
More advanced features

- Rucio can be interfaced and integrated with Workflow Management Systems
  - Already supporting the PanDA/ProdSys/Harvester ecosystem
  - DIRAC support currently under investigation

Rucio in a nutshell — Namespace

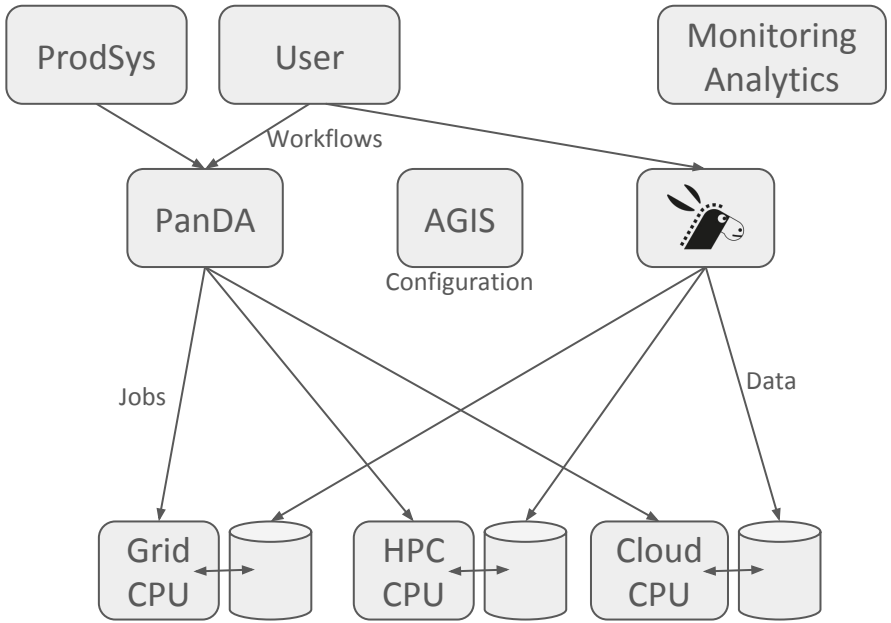


Rucio in a nutshell — Architecture



ATLAS Distributed Computing

- Data management system **Rucio**
  - Data cataloging and transfer management
  - Enforces data policies
- Workflow Management system **PanDA**
  - Job brokering and scheduling
  - Execution of jobs
- Additional systems
  - AGIS, ProdSys, Monitoring & Analytics
- Additional resources
  - WLCG sites, Tier0, HPCs, Cloud, Boinc, …



Community

Community


eneas
Advanced European Network of E-infrastructures
for Astronomy with the SKA


LCLS


Belle II


Science & Technology
Facilities Council

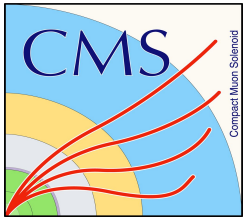

COMPASS


3D
EISCAT


ATLAS
EXPERIMENT


Alpha Magnetic Spectrometer
AMS-02
Europe • Asia
North America


Xe
XENON
Dark Matter Project

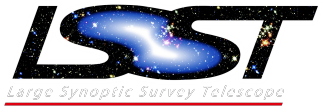

CMS
Compact Muon Solenoid


DUNE
DEEP UNDERGROUND
NEUTRINO EXPERIMENT


egi


ARC


cta
cherenkov telescope array


LSST
Large Synoptic Survey Telescope


IceCube
SOUTH POLE NEUTRINO OBSERVATORY


FTS


eXtreme DataCloud


Fermilab


BROOKHAVEN
NATIONAL LABORATORY

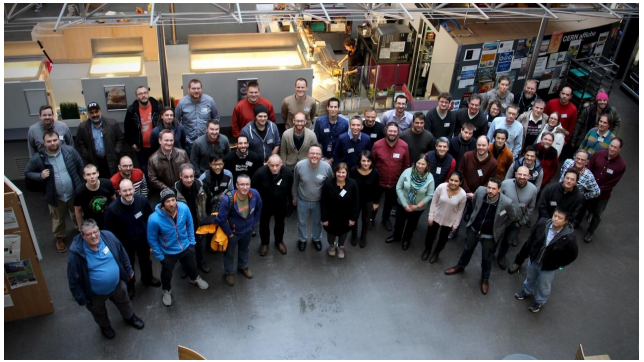

SKA
SQUARE KILOMETRE ARRAY


LIGO


Open Science Grid

Community

- 1st Rucio Community Workshop
  CERN, March 1-2, 2018
- 1st Rucio Coding Camp
  CERN, November 29-30, 2018
- 2nd Rucio Community Workshop
  Oslo, February 28 - March 1



Contributors

- 2017
  - **540** commits from **12** contributors
- 2018
  - **775** commits from **29** contributors
  - Top-10 contributors responsible for **85%** of commits
- Discussions in weekly development meetings
  - Indico
  - Planning meeting (3-4 month plan) for each feature release
- Communication via Slack

Rucio Coding Camp 2018

#2 Rucio Community Workshop

- Two Keynotes
  - Thursday Richard Hughes-Jones (GÉANT)
    Concepts and Architectures for the Next Generation Academic Networks
  - Friday Gudmung Høst (NeIC)
    The Nordic e-Infrastructure Collaboration
- 21 community presentations
  - Presentations 15’ + 5’
  - With a focus on (intended) Rucio usage, use cases, issues, and evolution
- Discussion sessions
- Technical discussions

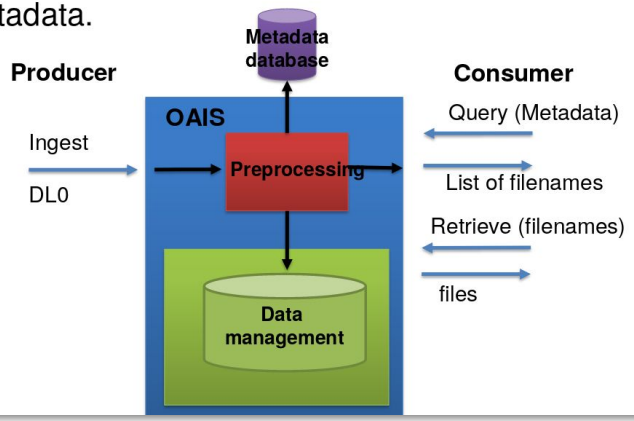
#2 Rucio Community Workshop

- Presentations from 21 communities, 66 attendees!
- Many different use-cases from HEP and astronomy experiments, software projects, and infrastructure providers
- Development plans and collaborations
  - Documentation improvements (from setup to monitoring and advanced operations)
  - Deployment (Kubernetes)
  - Archive support
  - Release model
  - Advanced metadata support
  - Additional authentication methods
  - Data hiding
  - Interfacing to additional WMS

CTA Feedback (F. Gillardo)

Data ingest / query / retrieve

- The CTA archive relies on the « Open Archival Information System » (OAIS) ISO standard. Event files are retrieved using metadata.



6

Other requirements

- Monitoring :
  - File transfer error rate
  - Location of the files and replicas
  - Rate of read/write…
- Authentication/authorisation :
  - Identity Federation (EDUGAIN)
  - LDAP
  - OpenID
- Query file using condition on Metadata
- Dirac as a workload manager
- Staging opération
- Reprocess raw data

Findable
Accessible
Interoperable
Reusable

7

SKA Feedback (R. Joshi)

HURDLES

(so far)

ASTRONOMY DATA IS DIFFERENT THAN HEP DATA

Data models and compute models are simply not all known at the moment, leaving us to consider for all worst-case scenarios. Large variations in the size of data, number files, metadata, volume of secondary data generated, required proximity of the data. User community specific defaults might help.

CHALLENGES OF A FLAT NAMESPACE

A lot more thought and planning is required when working in a flat namespace (especially since DIDs must remain unique within a scope for all time). Users need to be convinced of the benefits as well.

X509 CERTIFICATES (NOT SPECIFIC TO RUCIO)

We are looking into a credential translation solution as an alternative to x509 certificates. Would be interesting to see how well it fits in with Rucio.

LOOKING AHEAD

ELASTIC SEARCH

We need plots!

LIGHTWEIGHT RUCIO CLI

Easy way to upload large amounts of data that doesn't sit on Grid storage/Grid compatible storage/isolated storage. SRC users may chose to download raw data, take it a suitable computing platform and upload the results.

PERMISSIONS

Astronomers like to keep their data protected. Traditionally, once data is available it is accessible only to the authorized people in the project. Once the proprietary period has passed, data is made public.

INTEGRATION WITH A WMS

DIRAC – Rucio collaboration in the form of a Rucio-mode for DIRAC sounds very promising. At the moment, we are unable to replicate a use case end to end

DTN – DTN TRANSFER

For long distance data transfers, there will be dedicated high-bandwidth network links that would be better suited. Routing the data via a DTN would provide a well-defined high speed data transfer environment.

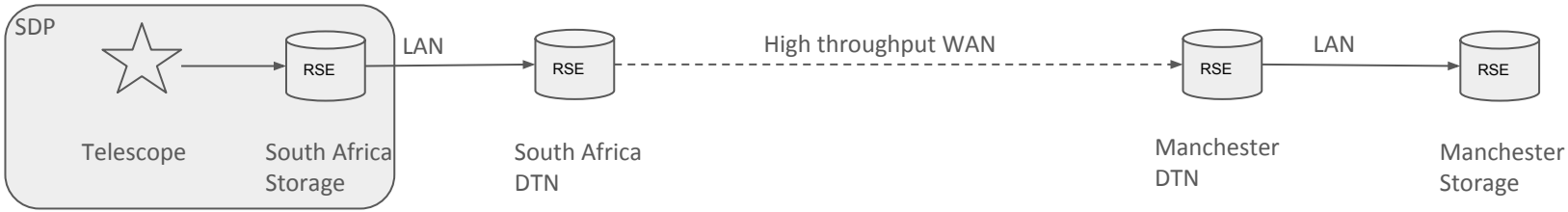
+ AENEAS All-Hands Meeting Feedback from yesterday: Generic & Searchable Metadata!

Ongoing evaluation by AENEAS

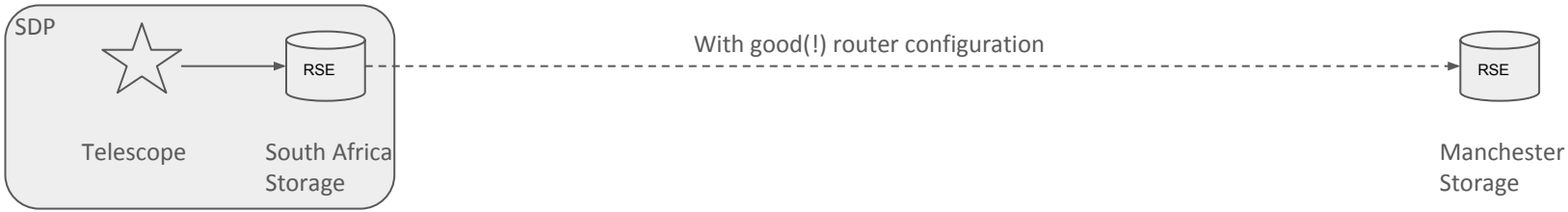
- Led by Manchester group
  - CERN providing best-effort support/guidance
- Full Rucio installation at Harwell Campus, incl. Clients/APIs/WebUI/etc...
  - Maintained by RAL computing group
  - ATLAS@CERN-like failsafe installation and setup
  - PostgreSQL as backend database (migrated from MySQL)
- Runs as *skatelescope.eu* VO against RAL FTS and storage
- Connects Manchester, RAL, Cambridge, QMUL, IDIA (.za) storage (soon Pawsey .au!)
  - Using various protocols (gsiftp, davs, s3, root)
- Demonstrated ingesting and transferring LOFAR data to/from South Africa
- Full mesh testing using hourly throwaway data
- Next main step is getting monitoring up and running (Elasticsearch)

Data Flow — Networks

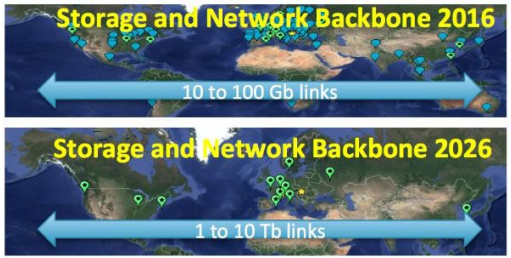
- Through various discussion with GEANT: exploit Data Transfer Nodes
  - Transfer from telescope to Jodrell using intermediary hops (3 rules)



  - Transfer from instrument to Jodrell using special router configuration (1 rule)



Data Flow — Networks



- So far, the NRENs have been generous and working well
  - Two orders of magnitude increase expected within 10y
  - When not working well, very difficult to debug for us
- Recently, there were requests to smooth traffic
  - Potential options are alternate path routing and SDN+NFV
  - Offload overcommitted networks — need to know flow in advance and should last multiple hours
  - Interplay between Networks + FTS + Rucio
  - First tests very promising — doubling of capacity between CERN and Amsterdam!
- Data Transfer Nodes
  - Network capability and performance monitoring, alerting, dedicated high-throughput channels, …
  - DTNs integral to US HPC — also being deployed across Europe in response to SKA
- Need to foresee orchestration between Rucio instances on shared infrastructures

Future

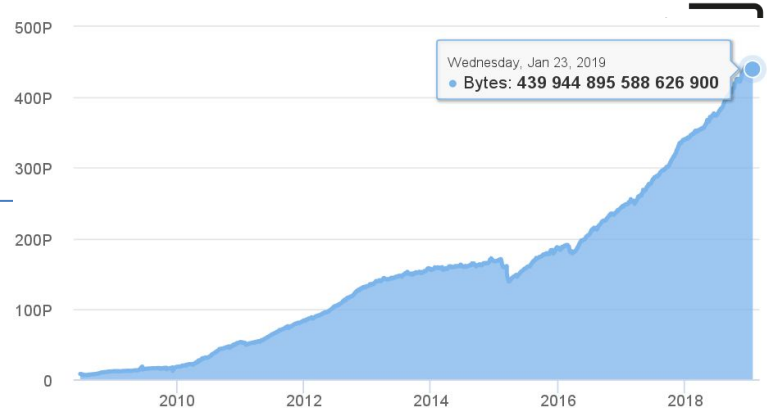
(a.k.a. Rucio in ESCAPE)

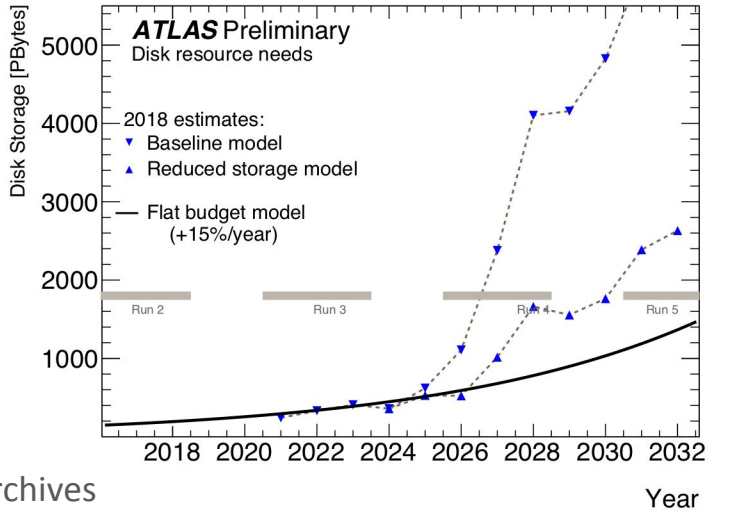
Data management in ATLAS is diverse

- Stable and reliable platform for the experiment
  - Rucio, dCache, XrootD, DPM, FTS, Dynafed, StoRM, ESnet, Geant, Davix, GFAL, AAI, …
  - Well integrated with infrastructure (OpenStack, Kubernetes, ActiveMQ, Hadoop, ElasticSearch, …)
  - Very low operational effort for the sheer scale of the provided platform
- We're positive that Run-3 will go smoothly — starting with R&D for Run 4 at HL-LHC
  - Teams are coming together in working groups, and producing and integrating prototypes such as ATLAS Data Carousel, WLCG AuthZ, DOMA QoS, DOMA TPC, DOMA Access, …
  - Address storage in terms of characteristics (latency, capacity, retrieval time, cost, …) → QoS
  - Will require optimized workflows and data flow planning with new IO patterns
  - Also looking into funded R&D programmes (European Commission & Co)
- Orchestrated multi-experiment data management is coming (HL-LHC, SKA, DUNE, …)
  - Sharing of infrastructure, software, expertise, and operations
  - Benefit from proven common solutions, enable large-scale science, build advanced functionality

Challenge: Data Volume

- Linear data increase
  - Steadily approaching half an Exabyte(!)
  - Storage budget crossing in 2021 if we don't do anything
  - HL-LHC volume jump is scary, even at reduced model
- What's being catalogued?
  - 1B files, 30M containers, 15M datasets
  - 5K accounts, 10K identities, 1K endpoints
- New workflows will bring significant increase on the content of the catalogue
  - E.g., massive flows with lots of very small files — support archives
  - Rucio has a scale-out architecture — challenging tasks to address but not prohibitive





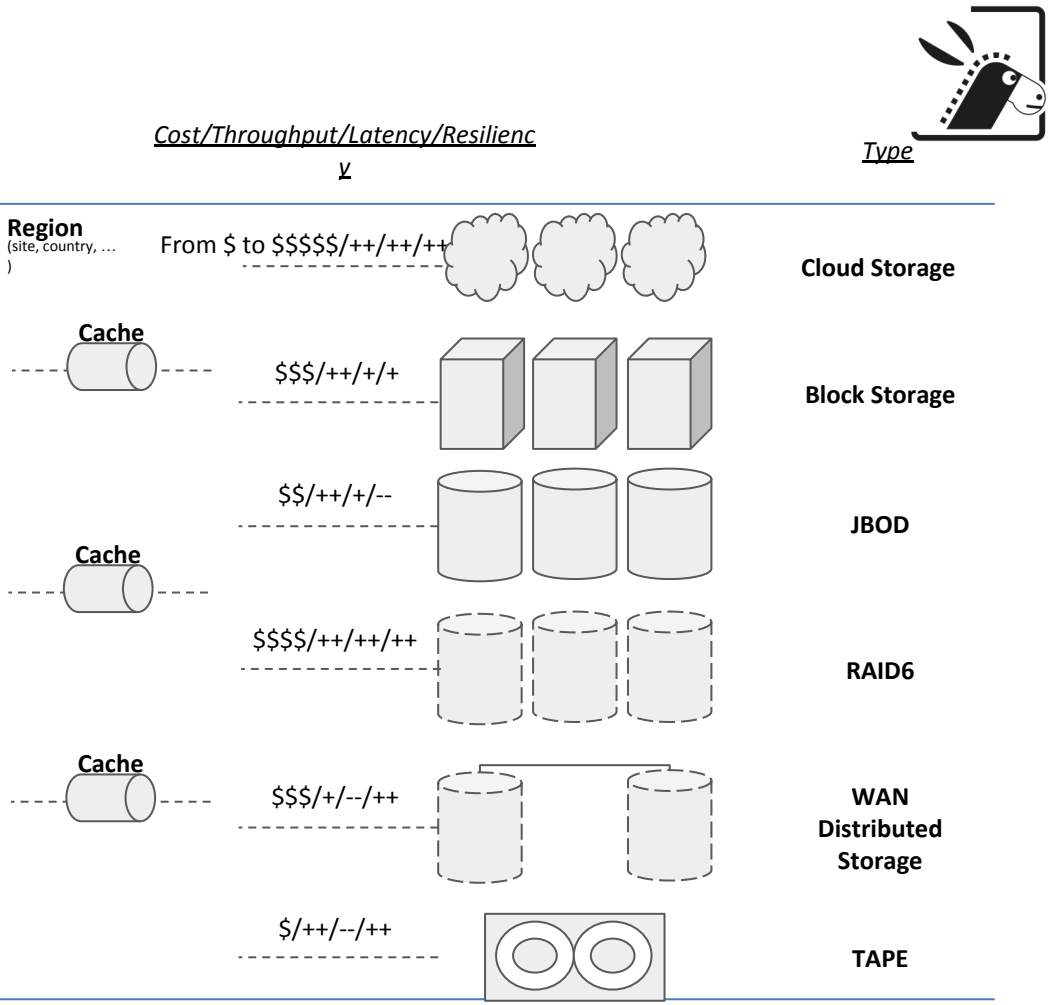
Expectation mgmt.

Dataset 1 — *Measurement*
- High cost to produce
- Used once

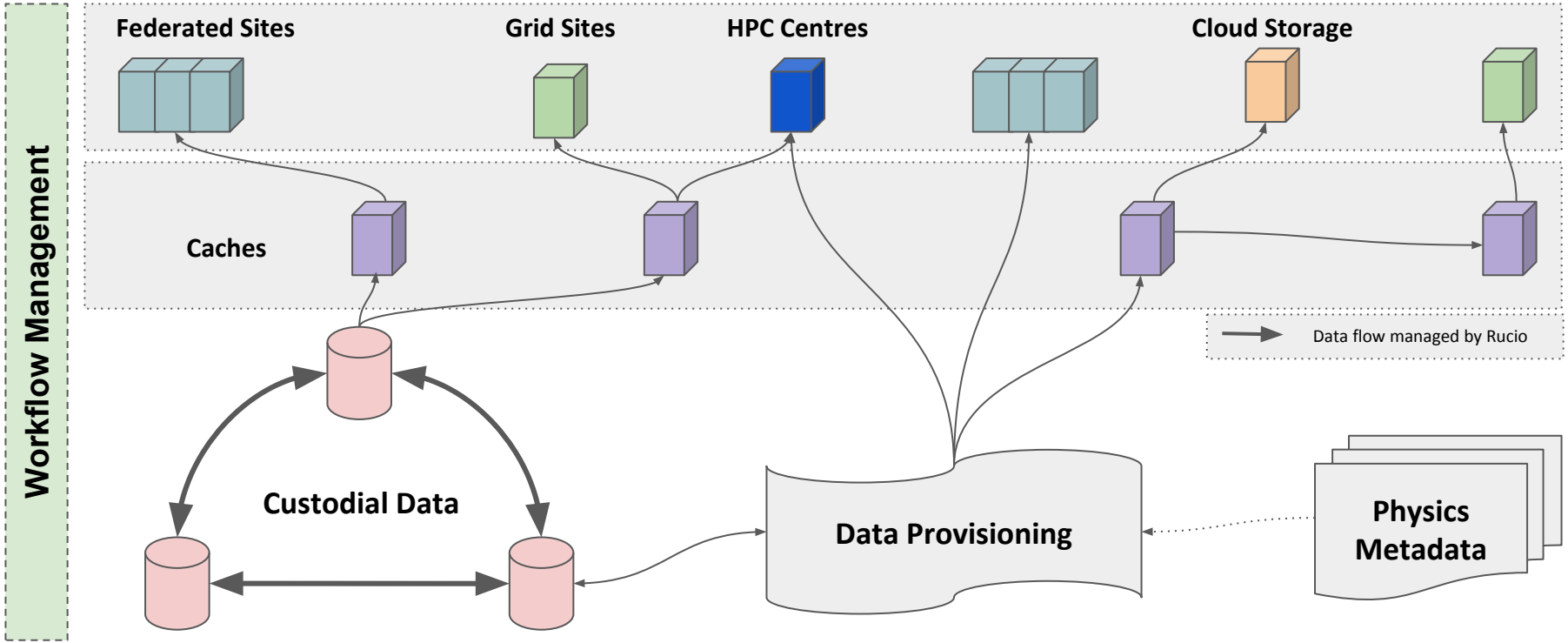
Dataset 2 — *Job Input*
- High cost to produce
- Used often for 2 months

Dataset 3 — *Job Output*
- Important for 2 weeks
- Might be needed in 6 months again

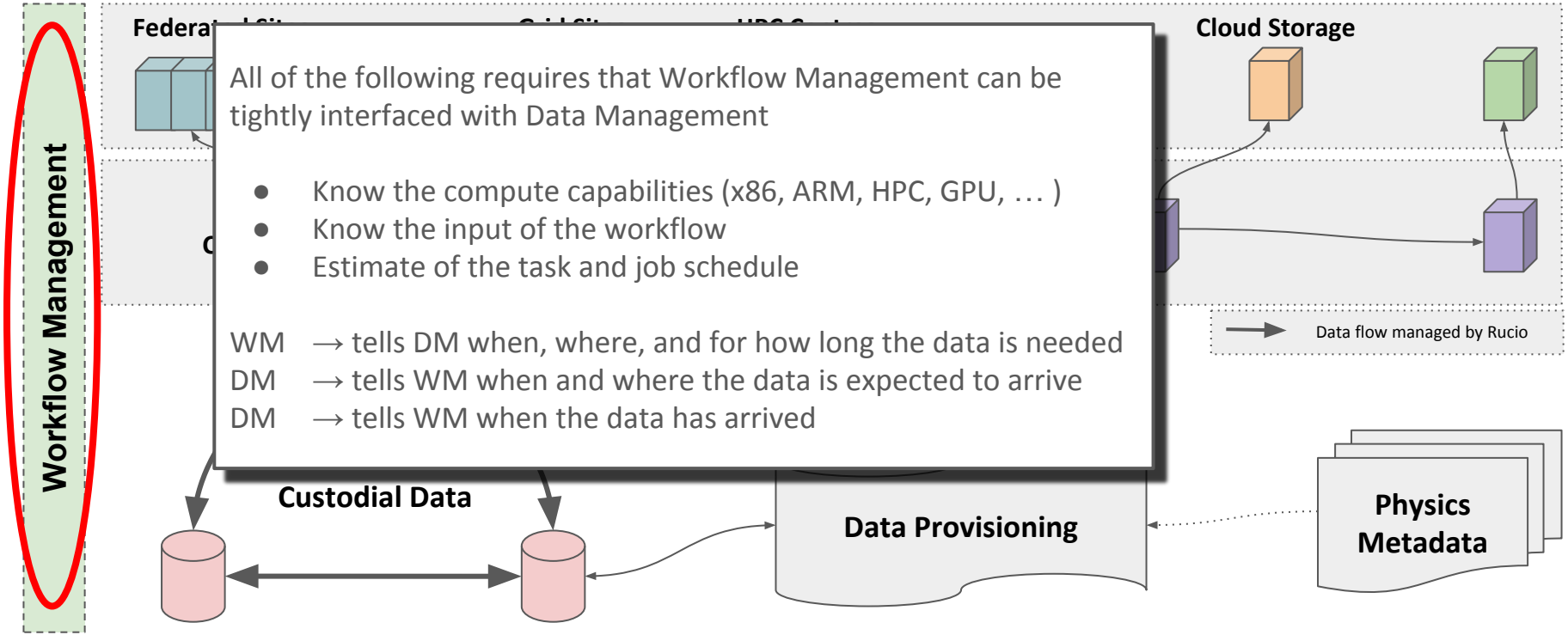
Dataset 4 — *Static auxiliary data*
- Cheap to produce
- Used very often



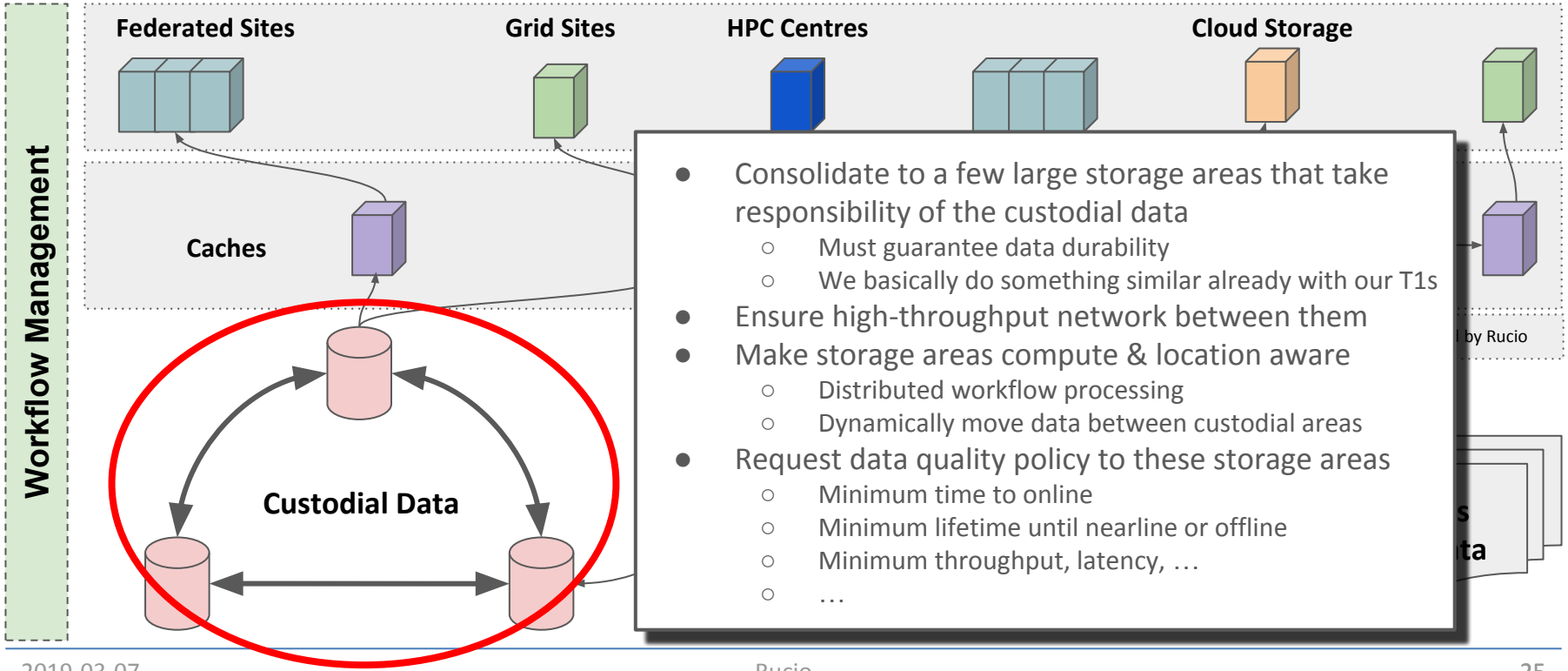
Challenge: Data Flow



Challenge: Data Flow

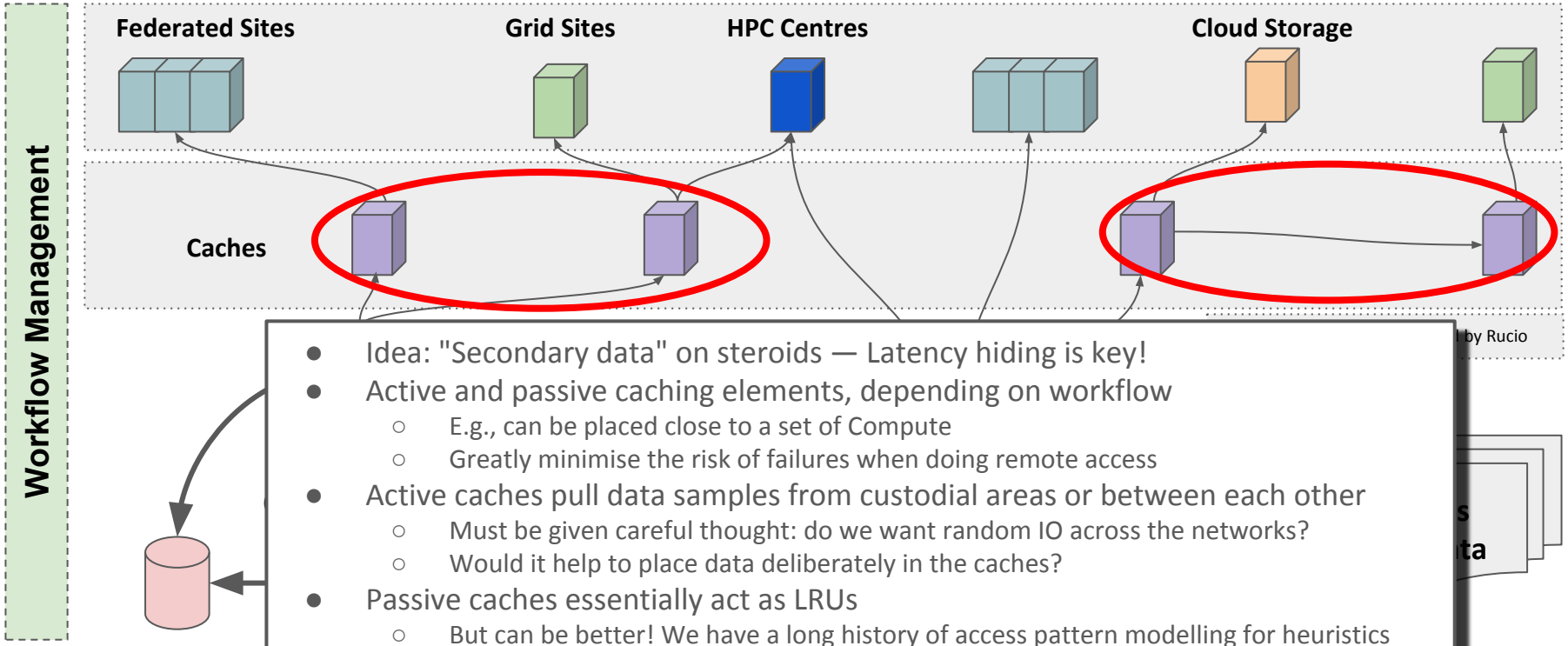


Challenge: Data Flow



- Consolidate to a few large storage areas that take responsibility of the custodial data
  - Must guarantee data durability
  - We basically do something similar already with our T1s
- Ensure high-throughput network between them
- Make storage areas compute & location aware
  - Distributed workflow processing
  - Dynamically move data between custodial areas
- Request data quality policy to these storage areas
  - Minimum time to online
  - Minimum lifetime until nearline or offline
  - Minimum throughput, latency, …
  - …

Challenge: Data Flow

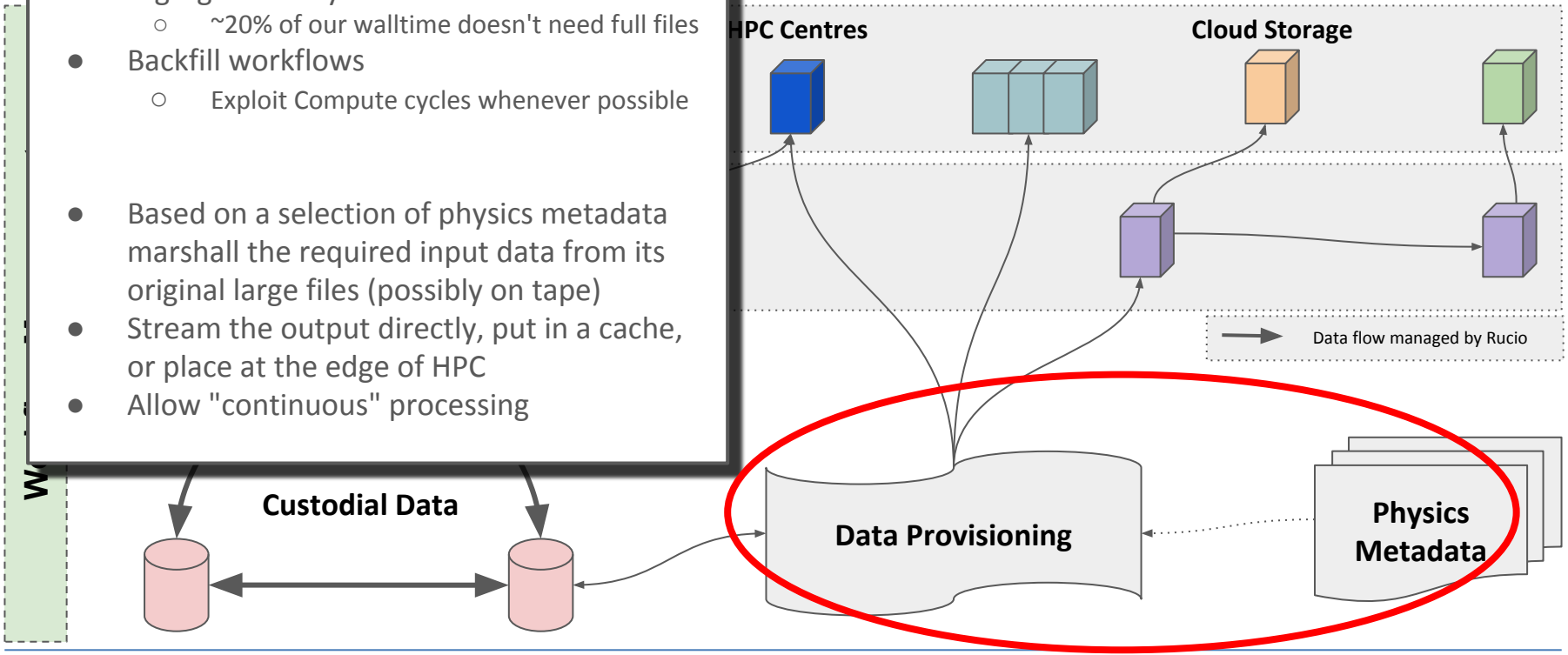


- Idea: "Secondary data" on steroids — Latency hiding is key!
- Active and passive caching elements, depending on workflow
  - E.g., can be placed close to a set of Compute
  - Greatly minimise the risk of failures when doing remote access
- Active caches pull data samples from custodial areas or between each other
  - Must be given careful thought: do we want random IO across the networks?
  - Would it help to place data deliberately in the caches?
- Passive caches essentially act as LRUs
  - But can be better! We have a long history of access pattern modelling for heuristics

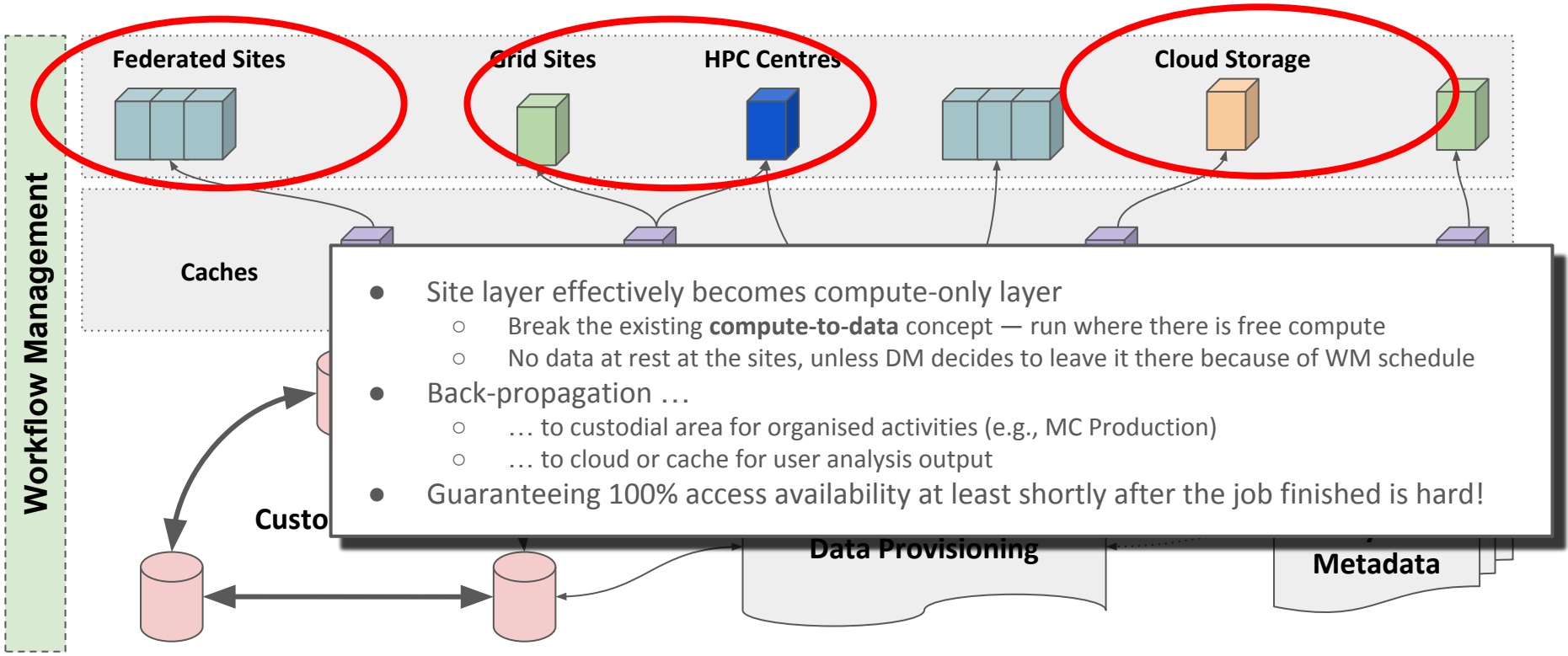
Challenge: Data Flow

- High-granularity workflows
  - ~20% of our walltime doesn't need full files
- Backfill workflows
  - Exploit Compute cycles whenever possible

- Based on a selection of physics metadata marshall the required input data from its original large files (possibly on tape)
- Stream the output directly, put in a cache, or place at the edge of HPC
- Allow "continuous" processing



Challenge: Data Flow



- Site layer effectively becomes compute-only layer
  - Break the existing **compute-to-data** concept — run where there is free compute
  - No data at rest at the sites, unless DM decides to leave it there because of WM schedule
- Back-propagation …
  - … to custodial area for organised activities (e.g., MC Production)
  - … to cloud or cache for user analysis output
- Guaranteeing 100% access availability at least shortly after the job finished is hard!

Miscellaneous data topics

- More cloud storage
  - Support scientific clouds via DynaFed
  - Google Cloud Storage — Native support in Rucio
- Better object store support
  - Opinion: We are limiting ourselves with S3-style access to Object Stores
  - Is there enough traction to do R&D for native objectstore interaction (e.g., go straight to RADOS?)
  - DUNE and RAL contributing effort to improve Rucio object store support
- Turnkey deployment with Docker & Kubernetes now available
  - Lots of operations time spent debugging Openstack+Puppet+Distrosync+PIP+...
  - Allow spot instances of Rucio for dedicated use-cases
- Further extend metadata support
  - Primarily needed by non-ATLAS experiments using Rucio
  - Extend dataflow engine to make use of custom client metadata

Miscellaneous data topics

- Integration with research databases
  - e.g., Zenodo pointing to Rucio datasets
- Integration with Sync'n'Share systems (CERNbox, Nextcloud)
- Data hiding with lifecycle/lifetime policies
- Next to ESCAPE, participating in EC submissions
  - ATTRACT: Extend Rucio for multi-experiment use cases, very helpful for AENEAS, led by Manchester
  - ATTRACT: Diverse cloud computing support (with Exoscale, Google and others)
  - EOSC-02-2019: INDIGO-Next
- CERN EP R&D submission for Rucio as multi-experiment common solution
  - Separate Rucio instances per experiment
  - But they can orchestrate their high-level dataflow policies with the shared infrastructure underneath!
  - Short-term lease of network or storage across experiments where/when appropriate and possible

More information

Website	http://rucio.cern.ch
Documentation	https://rucio.readthedocs.io
Repository	https://github.com/rucio/
Continuous Integration	https://travis-ci.org/rucio/
Images	https://hub.docker.com/r/rucio/
Online support	https://rucio.slack.com/messages/#support/
Developer contact	rucio-dev@cern.ch
Journal article	https://arxiv.org/abs/1902.09857