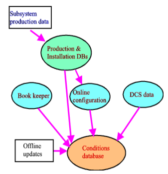
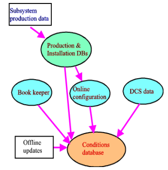
ADC technical interchange meeting
Tokyo, May 2013



Database aspects of ATLAS distributed computing

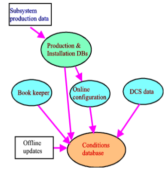
Gancho Dimitrov (CERN)

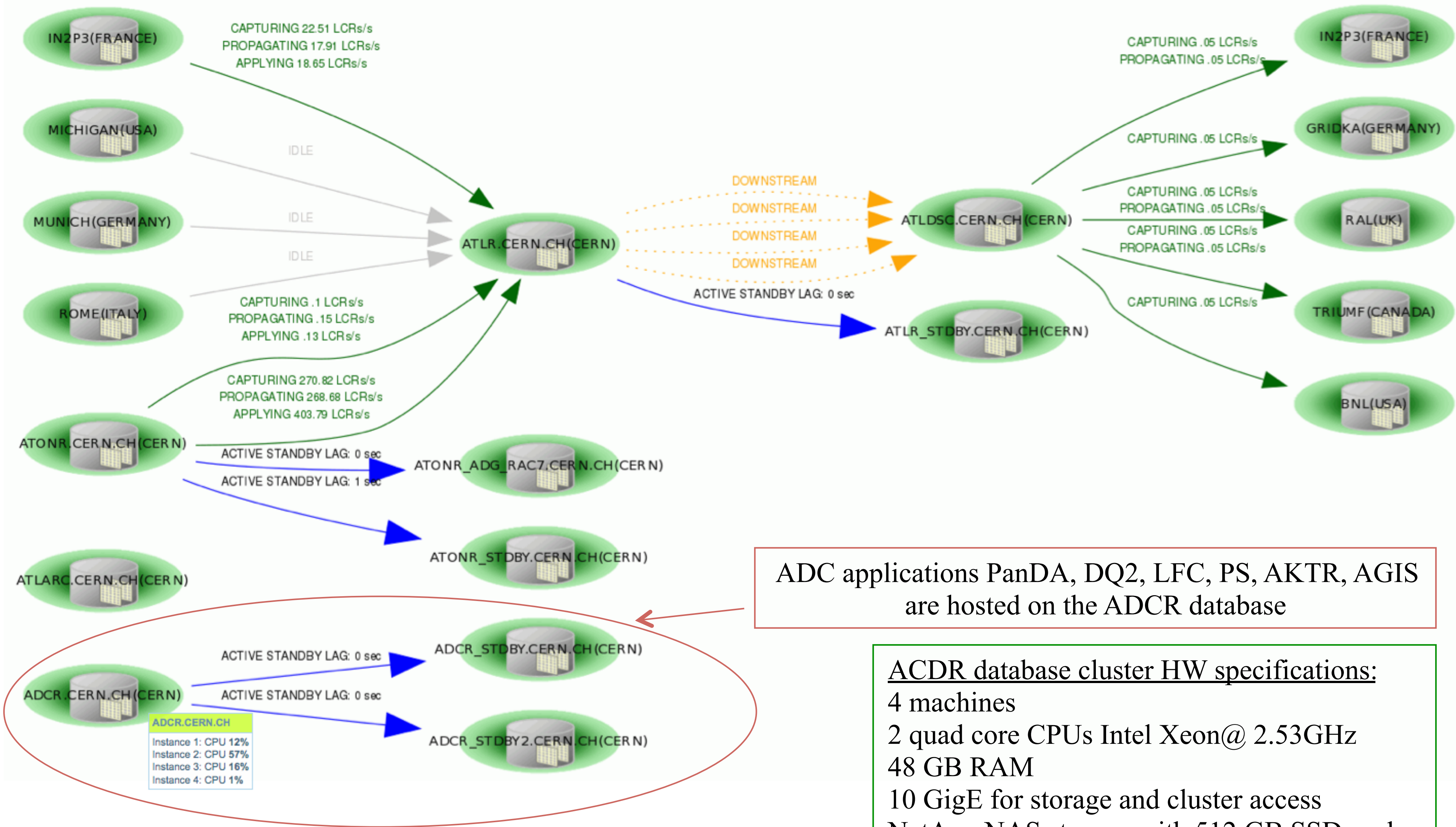


Outline

- Current ATLAS database topology, HW specifications, metrics
- PanDA accounts, roles and privileges organization
- PanDA data flow and management, volume and trends, improvements
- New developments
  - JEDI (Job Execution and Definition Interface)
  - Rucio data management system
- Newer hardware specs for the ADCR database
- Conclusions

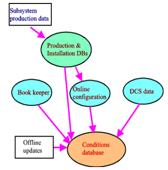
ATLAS databases topology





ADC applications PanDA, DQ2, LFC, PS, AKTR, AGIS are hosted on the ADCR database

<u>ACDR database cluster HW specifications:</u>
4 machines
2 quad core CPUs Intel Xeon@ 2.53GHz
48 GB RAM
10 GigE for storage and cluster access
NetApp NAS storage with 512 GB SSD cache



PanDa job management system: accounts, roles, privileges

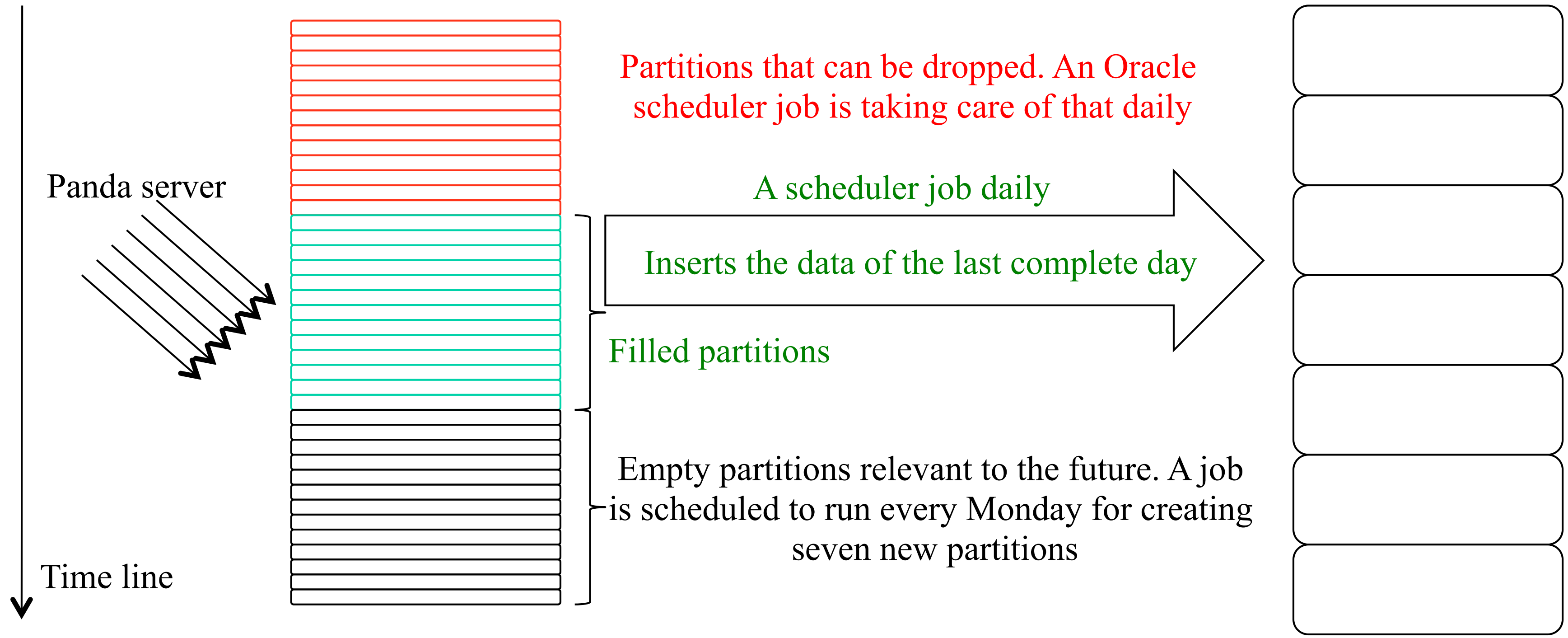
Owner account	Write privileges granted to	Read privileges granted to
ATLAS_PANDA	ATLAS_PANDA_WRITEROLE	ATLAS_PANDA_READROLE
ATLAS_PANDAMETA	ATLAS_PANDAMETA_WRITEROLE	ATLAS_PANDA_READROLE
ATLAS_PANDAARCH	ATLAS_PANDA	ATLAS_PANDA_READROLE
ATLAS_DEFT	ATLAS_DEFT_W	ATLAS_PANDA_READROLE

- Panda server is allowed to modify the PANDA, PANDAMETA and PANDAARCH
- PanDA monitor is allowed to modify only the PANDAMETA data
- DEfT (Database Engine for Tasks) can modify only its own tables via direct privileges (PanDA gets a direct write privilege in a single DEFT table)
- All owner accounts grant select privilege on their objects to the ATLAS_PANDA_READROLE
- All application writer and reader accounts get read privilege on all objects by granting the ATLAS_PANDA_READROLE to them

PanDA data organization

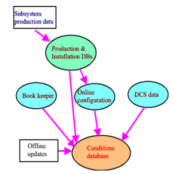
ATLAS_PANDA JOB, PARAMS, META and FILES tables: partitioned on a 'modificationtime' column.
Each partition covers a time range of a day

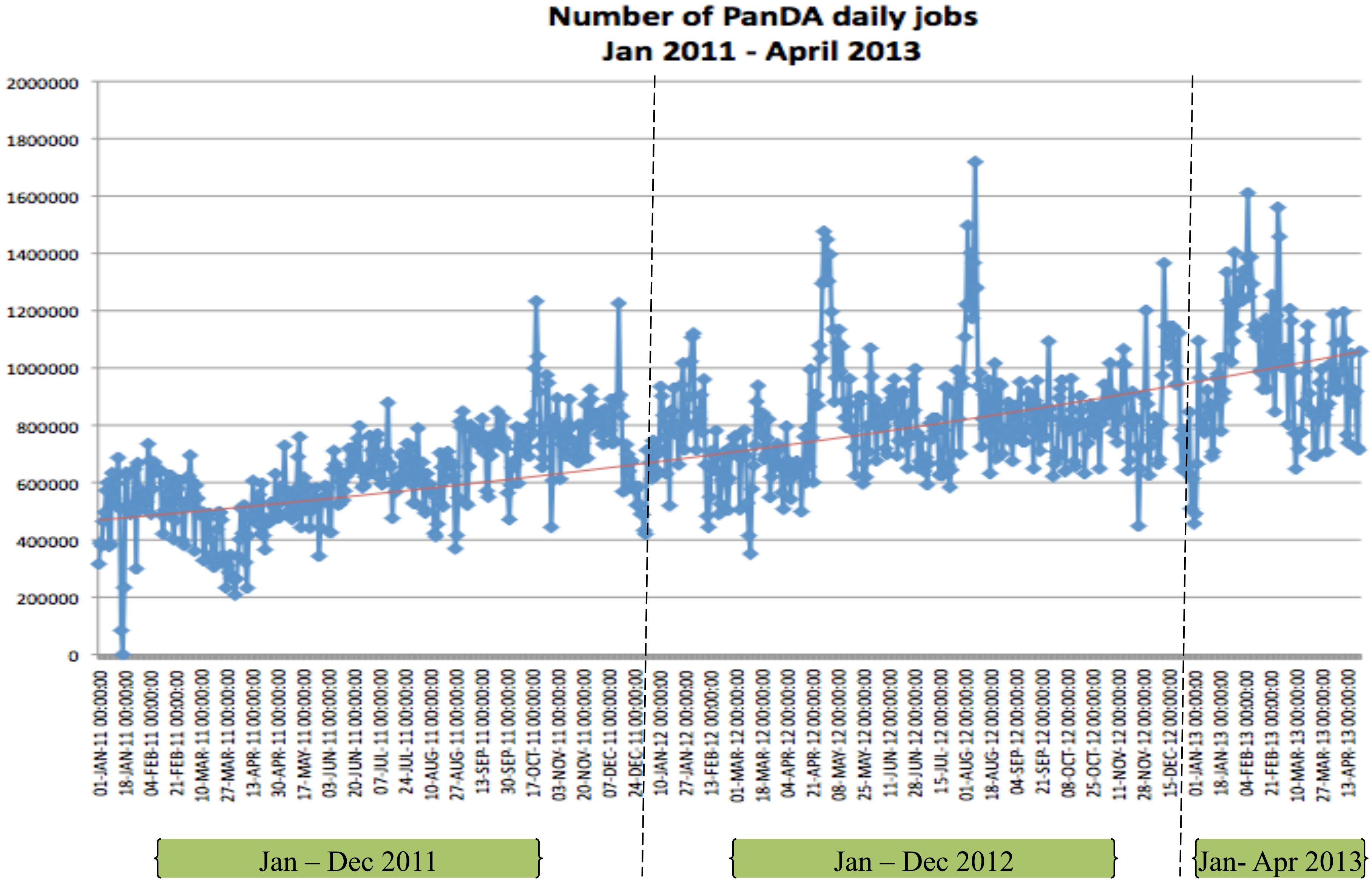
ATLAS_PANDAARCH archive tables partitioned on the 'modificationtime' column.
Some table have defined partitions each covering three days window, others a time range of a month



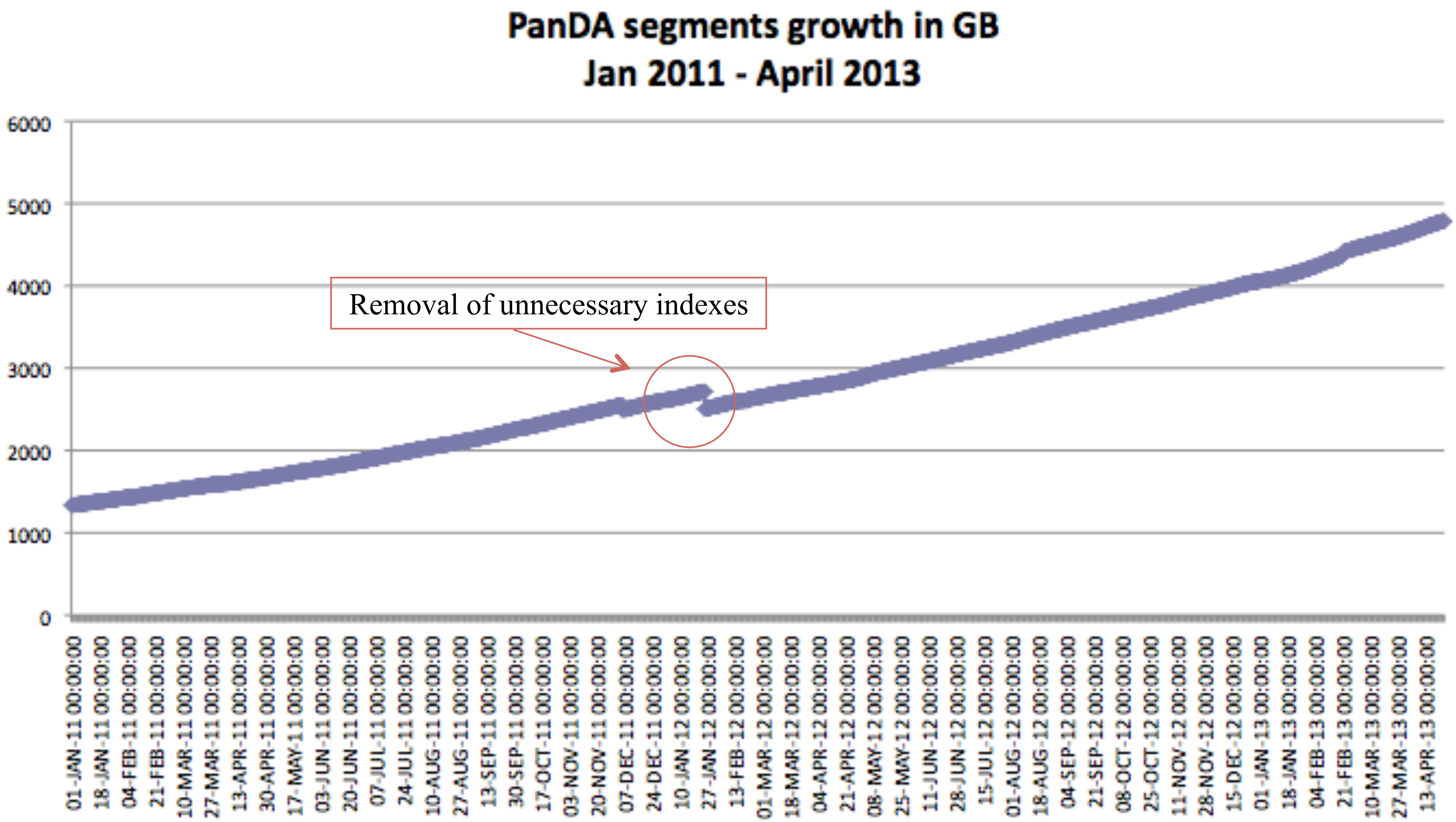
Certain PanDA tables have defined data sliding window of three days, others 30 days.
Data removal on partition level happens only after data being copied to the PANDAARCH schema.
This natural approach showed to be adequate and not resource demanding !

Trend in number of daily PanDA jobs

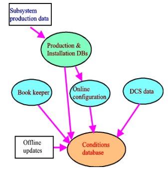




Trend in PanDA segments growth



Jan – Dec 2011: **1.3 TB** | Jan – Dec 2012 : **1.7TB** | Jan–Apr 2013:**0.7 TB**



Improvements on the current PanDA system

- Despite the increased activity on the grid, using several tuning techniques the server resource usage stayed in the low range: e.g. CPU usage in the range 20-30%

- Thorough studies on the WHERE clauses of the PanDA server queries resulted in:

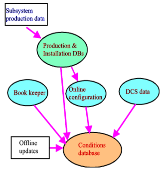
  - revision of the indexes: removal or replacement with more approriate multi-column ones

  - weekly index maintenance (rebuids triggered by a scheduler job) on the tables with high transaction activity

OWNER	JOB_NAME	JOB_CLASS	RUN_DURATION	LAST_START_DATE	LAST_RUN_DURATION	LAST_STATUS	STATE	NEXT_RUN_DATE	REPEAT_INTERVAL
ATLAS_PANDA	PANDA_TAB_INDICES_REBUILD	PANDA_JOB_CLASS	-	04-MAR-2013 11:00	0 00:00:48.0	SUCCEEDED	SCHEDULED	11-MAR-2013 11:00	FREQ=WEEKLY; BYDAY=MON

  - queries tuning
  - an auxiliary table for the mapping Panda ID <=> Modification time

- In the light of the Rucio and the DQ2, PanDA now works with files and datasets with defined scope



PanDA complete archive

- PanDA full archive now hosts information of 800 million jobs – all jobs since the job system start in 2006
- Studies on the WHERE clauses of the PanDA monitor queries resulted in:

  - revision of the indexes: replacement with more approriate multi-column ones so that less tables blocks get accessed

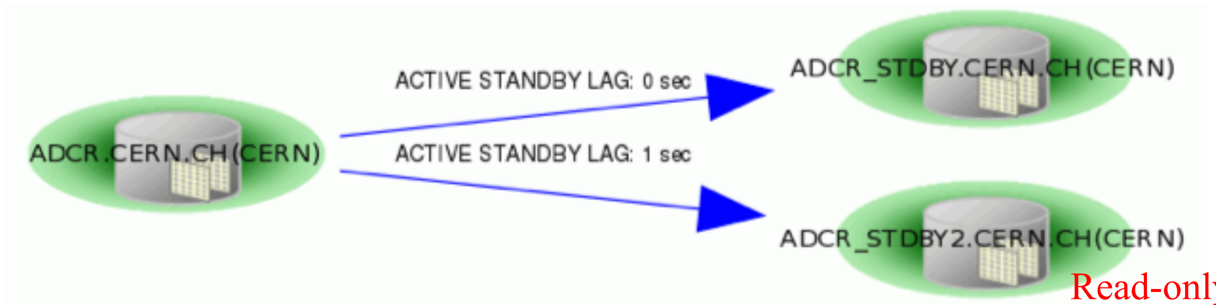
  - dynamic re-definition of different time range views in order to protect the large PanDA archive tables from time range unconstrained queries and to avoid time comparison on row level.

  The views comprise set of partitions based on ranges of 7, 15, 30, 60, 90, 180 and 365 days. These views are shifting windows with defined ranges and the underlying partition list is updated daily automatically via a scheduler job.

VIEW_NAME	STATUS	LAST_DDL_TIME
JOBSARCHVIEW_7DAYS	VALID	26-APR-13 12:00:09
JOBSARCHVIEW_30DAYS	VALID	26-APR-13 12:00:11
JOBSARCHVIEW_15DAYS	VALID	26-APR-13 12:00:11
JOBSARCHVIEW_60DAYS	VALID	26-APR-13 12:00:11
JOBSARCHVIEW_90DAYS	VALID	26-APR-13 12:00:11
JOBSARCHVIEW_180DAYS	VALID	26-APR-13 12:00:11
JOBSARCHVIEW_365DAYS	VALID	26-APR-13 12:00:10

Potential use of ADG from PanDA monitor

- ADCR database has two standby databases:
  - Data Guard for disaster recovery and backup offloading
  - Active Data Guard (ADCR_ADG) for read-only replica



Read-only replica

- PanDA monitor can benefit from the Active Data Guard (ADG) resources

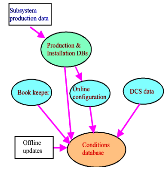
  It is planned that PanDA monitor sustains two connection pools:

  - to the primary database ADCR

  - to ADCR's ADG

  The idea is queries that span on time ranges larger than certain threshold to be resolved from the ADG where we can afford several paralell slave processes per user query.

New development: JEDI (a component of PanDA)



- JEDI is a new component of the PanDA server which dynamically defines jobs from a task definition. The main goal is to make PanDA task-oriented.
- Tables of initial relational model of the new JEDI schema (documented by T. Maeno) complement the existing PanDA tables on the INTR database

- Ongoing activities :

  - understand the new data flow, requirements, access patterns

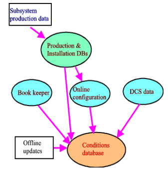
  - studies for the best possible data organization (partitioning) from manageability and performance point of view.

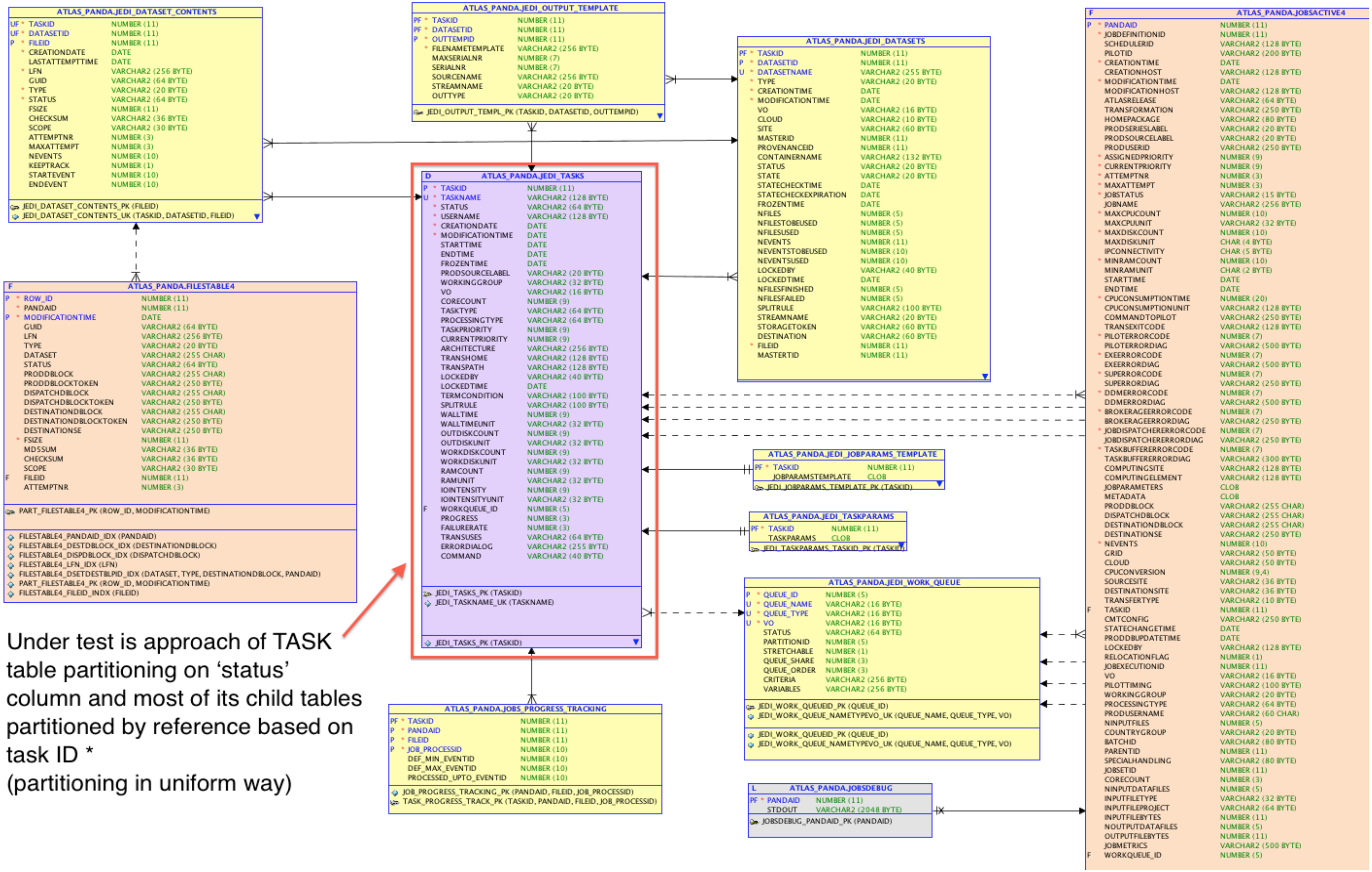
  - address the requirement of storing information on event level for keeping track of the active jobs' progress. That data will be transient, but the writing and reading of it must be highly optimised.

  - get to the most appropriate physical implelentaion of the agreed relational model

  - tests with representative data volume

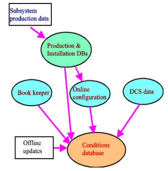
JEDI database relational schema





Under test is approach of TASK table partitioning on 'status' column and most of its child tables partitioned by reference based on task ID *
(partitioning in uniform way)

Transition from current PanDA schema to a new one



- The idea is the transition from the current PanDA server to the new one with the DB backend objects to be transparent to the users.
- JEDI tables are complementary to the existing PanDA tables. The current schema and PANDA => PANDAARCH data copying should not change much.
- However the relations between the existing and the new set of tables have to exist. In particular:

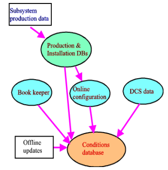
  - Relation between JEDI's ***Task*** and PanDa's ***Job*** by having a foreign key in all JOBS* tables to the JEDI_TASKS table

  - Relation between JEDI's ***Work queue*** (for different shares of workload) and PanDa's ***Job*** by having a foreign key in all JOBS* tables to the JEDI_WORK_QUEUE table

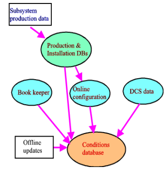
  - Relation between JEDI's ***Task, Dataset and File*** (new seq. ID) and PanDA's ***Job*** processing the file (or fraction of it) by having a foreign key in the FILESTABLE4 table to the JEDI_DATASET_CONTENTS table

  (when a task tries a file multiple times, there are multiple rows in PanDA's FILESTABLE4 while there is only one parent row in the JEDI's DATASET_CONTENTS table)

Steps in JEDI database tests



- Step 1: The ATLAS_PANDA schema and data exported from production and imported on the INTR testbed database – done
- Step 2: New set of JEDI tables created into the imported schema. Alter tables where necessary. Set relations between the existing tables and the new ones, enforce integrity when possible – done
- Step 3: DEfT (Database Engine for Tasks) schema validation. Test JEDI and DEfT interaction and access rights.
- Step 4: Validate access patterns to the new objects. Studies on the efficiency of the chosen partitioning and index strategy. Agreement on data archiving policy to be achieved.
- Step 5: If needed introduce changes to the PANDA => PANDAARCH data flow. Activate the scheduler jobs responsible for the data copying and maintenance of the PANDA sliding window.
- Step 6: Test with representative workload and repeat 2,3,4,5 until getting to an acceptable state.
- *Step 7:* *Import the PANDA, PANDAARCH, PANDAMETA data from the production. Repeat step 2 to validate again the SQL script. Perform pre-production test with all PanDA, JEDI, DEfT components.*



New development: Rucio system

Rucio is the new ATLAS file management system meant as successor of DQ2. The performed validation on the first Rucio relational schema initially created by the DDM team resulted in :

- DB objects (tables, constraints, indices, triggers, sequences) are named based on an agreed naming convention. The purpose is to have easy way to identify object role and relation with the others.

- studies of different partitioning techniques. A promising one is the list partitioning based on a virtual column “scope” concatenated with “data type” (file, dataset, container, del_file, del_dataset, del_container)

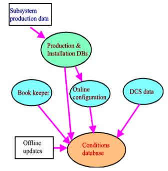
A home made logic of automatic partition creation tested and showed to be reliable.

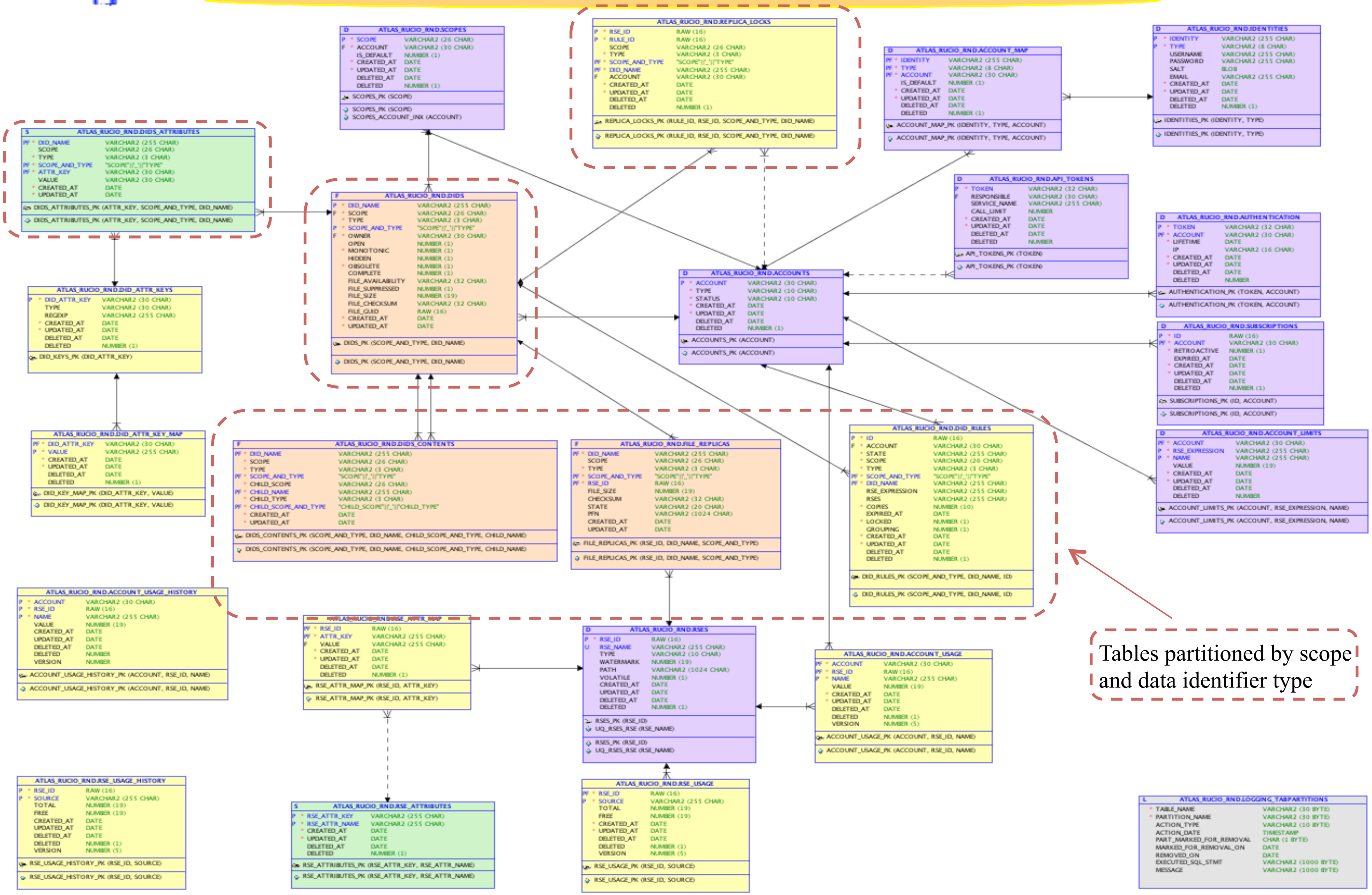
- special check constraints logic for enforcing important policies

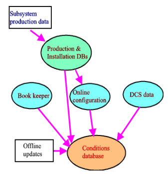
- explored techniques for guaranteeing any object name uniqueness within each scope for the full lifetime of the system

- test with representative data volume is needed

New development : Rucio DB relational schema







New HW for the ADCR cluster

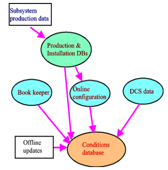
In Q4 of 2013 the ADCR hardware will be replaced by a newer one:

=> CPU 2x6 cores of newer type
(about 2x more CPU power compared to now)

=> RAM 128 GB
(3x more than now, great for caching more data blocks)

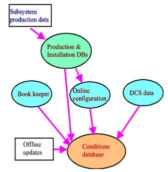
=> 10 GigE for storage and cluster access
(same as now)

=> Storage: newer generation and higher spec NetApp NAS with more SSD cache and more RAM
(about 2x in performance for random reads and writes)



What's next?

- The new HW will give a room for increased performance and throughput.
- However a big challenge and responsibility on many of us is to design and develop DB applications with scalability in mind to serve ever increasing workload for many years ahead.
- Much tight collaboration between SW developers, DB application admins and DB infrastructure DBAs is needed.



Conclusions

- The ADCR database resource usage stayed in the green zone for the whole 2012 and Q1 of 2013.

  All that is due to the hard work from many experts on keeping the DB HW and SW in good state, usage of new Oracle 11g features in production, ADCR application improvements and maturity.

- On focus now are developments of few new systems. Collaborative work is much appreciated.

THANK YOU!