A Study on Dynamic Data Distribution for the ATLAS Distributed Data Management

CHEP 2015, 13.04.2015, Thomas Beermann
CERN / University of Wuppertal

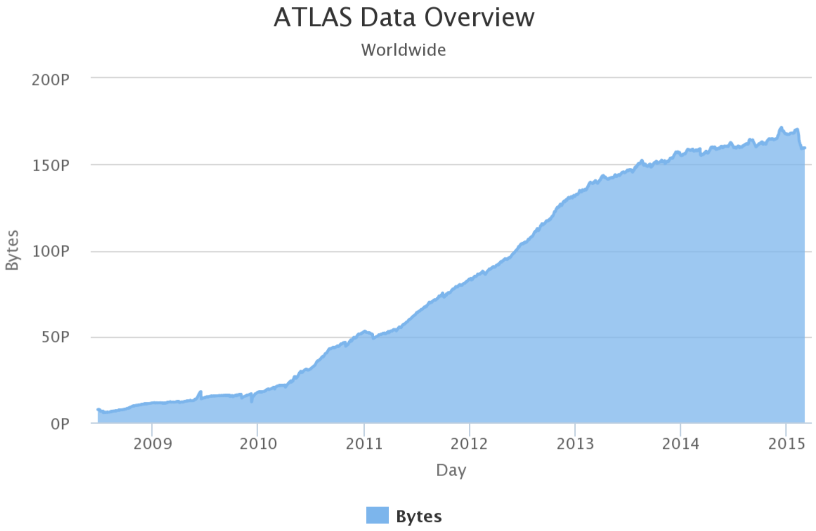
CERN

BERGISCHE
UNIVERSITÄT
WUPPERTAL

ATLAS EXPERIMENT

What is ATLAS DDM?

- DDM = Distributed data management
- Manages all ATLAS data on the WLCG
- Currently around 160PB of experiment data (detector data, Monte Carlo, user output, …)
- Spread over more than 150 sites worldwide
- Users interact with DDM through the workload management system (WMS), directly (local download and data transfer requests)



Popularity System

- Based on a tracer system, which tracks all data accesses on the grid. Example trace:

time	dataset	file	user	site	eventtype	...
2014-03-26 14:02:31	dataset1	file1	jdoe	CERN	analysis	...

- Creates daily reports that are aggregated per:
  - Dataset
  - User
  - Event Type (local download, analysis, production)
  - Involved Sites
- Contained information:
  - Number of operations
  - Number of file accesses

About this study

Study dynamic data distribution to minimise waiting time for analysis jobs

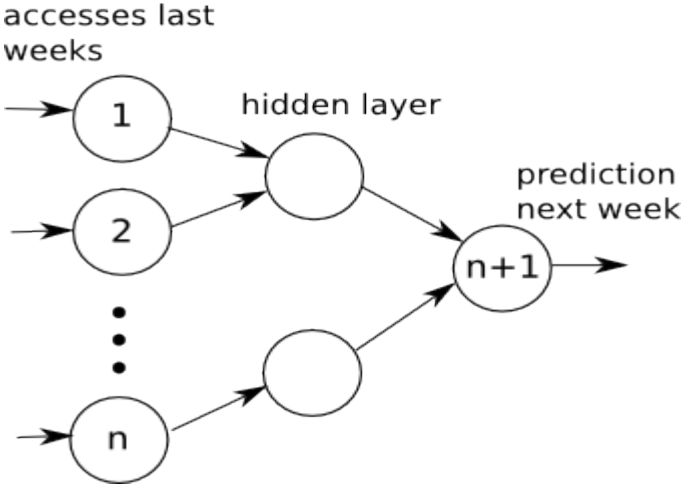
→ Delete rarely used replicas and use space for extra replicas of more popular data

→ Information about dataset popularity is available but not used in an automated way

⇒ Automatically optimise the data distribution on the grid, based on dataset popularity, to better exploit resources and reduce user waiting times

Dataset Access Prediction

- Prediction based on past accesses
- Different methods have been evaluated
- Static Prediction:
  - accesses stay constant from one week to another
- Neural Network prediction:
  - train on common dataset with similar access patterns
  - based on last **n** weeks
  - predict accesses on week **n+1**



- Hybrid (used later in evaluation):
  - NN prediction for popular data, filtering based on threshold of past week accesses
  - static for everything else

Redistribution

- Based on past and future popularity
- First part: Replica deletion
  - If no accesses for a certain number of weeks, reduce number of replicas for a dataset
  - but keep minimum custodial copies per dataset
- Second part: Replica creation
  - For datasets predicted as popular, add new replicas
- Both parts have to work together:
  - deletion only removes replicas at a site if a new replica is added
  - maximum amount of bytes to delete/add (i.e., turnover limit) can be set

Evaluation

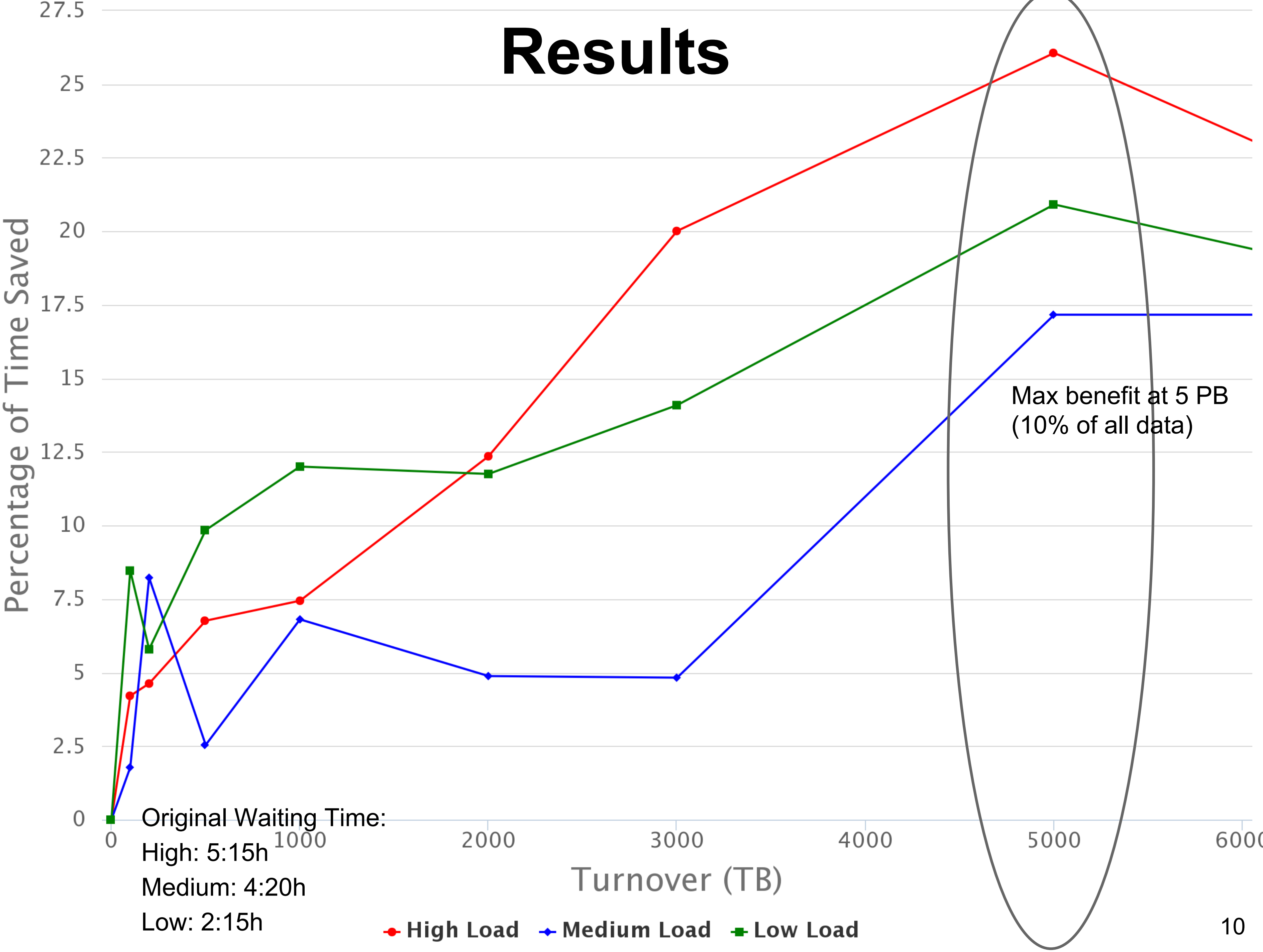
Simulator

- Simple event-based simulator
- Allows to run same workload on different data distributions
- Inputs:
  - Dataset replica catalogue (extracted from DDM)
  - Site configuration (number of job slots per site)
  - Jobs (from DDM traces and WMS archive) defined as (dataset, submission time, execution time)
- Jobs are scheduled at sites and block slots for their real measured runtime
- If all slots are taken, jobs have to wait

Simulation Setup

- Total of 13 weeks simulation period
- Evaluation of 3 specific weeks within 13 weeks with high, medium and low load
- Simulating user analysis jobs
- Only moving simulation and detector data used for user analysis
- Redistributing only on centrally managed space with different turnovers
- Metric: Waiting time per jobs => Average waiting time

Results



Max benefit at 5 PB (10% of all data)

Original Waiting Time:
High: 5:15h
Medium: 4:20h
Low: 2:15h

Conclusion & Outlook

- It is possible to make forecasts of future data accesses that are usable to redistribute data in order to decrease job waiting time.
- The more data is moved the bigger the benefit until a turning point where benefit stays constant.
- The maximum benefit for all weeks is between 18% and 26%, which depending on the week can correspond to more than 70 minutes of waiting time saved.
- Used data from Run 1 for simulation, now has to be adapted for the new data types and systems of Run 2

Questions?