A Probabilistic Analysis of Data Popularity in ATLAS Data Caching

M Titov\textsuperscript{1}, G Záruba\textsuperscript{1}, A Klimentov\textsuperscript{2}, K De\textsuperscript{1} for the ATLAS Collaboration

\textsuperscript{1} The University of Texas at Arlington, Arlington, TX 76019, USA
\textsuperscript{2} Brookhaven National Laboratory, Upton, NY 11973, USA

E-mail: mikhail.titov@cern.ch

Abstract. One of the most important aspects in any computing distribution system is efficient data replication over storage or computing centers, that guarantees high data availability and low cost for resource utilization. In this paper we propose a data distribution scheme for the production and distributed analysis system PanDA at the ATLAS experiment. Our proposed scheme is based on the investigation of data usage. Thus, the paper is focused on the main concepts of data popularity in the PanDA system and their utilization. Data popularity is represented as the set of parameters that are used to predict the future data state in terms of popularity levels.

1. Introduction
Efficient distribution of physics data over ATLAS grid sites is one of the most important tasks for user data processing [1]. The initial static data distribution model (Planned Data Placement) has been shown to create a bottleneck in data processing, where the availability of popular datasets had a profound effect on the utilization of the specific computational resources and resulted in uneven loads; this was caused by over-replication of unpopular data that have filled up disk storage space while under-utilizing of some processing resources due to low numbers of replicas of popular (desired) data.

Thus, a new data distribution mechanism was implemented within the production and distributed analysis system PanDA; PD2P (PanDA Dynamic Data Placement) dynamically reacts to user data needs [2], basing dataset distribution principally on user demand. Data deletion is also demand driven (by the Replica Reduction Agent), reducing the numbers of replicas for unpopular data to get space for more popular data [3]. This dynamic model has led to substantial improvements in efficient utilization of storage and processing resources [4].

Based on the above experience, in this work we seek to further improve the data placement policy by investigating in detail how data popularity can be obtained and its impact on prediction of data popularity in the future. The history data of PanDA jobs including both production and analysis is used for our analysis of the behavior of data usage.

2. Primary data analysis
In order to perform our experiments, we have restructured and stored data about PanDA jobs in a NoSQL MongoDB database (schema-free, document-oriented database). This provides an advantage when retrieving and further processing data statistics information. Two repositories are used for data collections (with only logical separation between these repositories):

- Raw data from PanDA database (jobs collection).
• Processed data that contains job inputs such as: datasets, containers (logical grouping of datasets), data patterns.

The final structure for data collections has not been defined yet, thus requests for information retrieval are sent to both data repositories.

2.1. Evaluation of PanDA data

One of the possible ways to predict dataset popularity is by using classical probability theory. This approach is defined in the first stage of the “Two-stage replica replacement algorithm” [5], which is a part of the research work for data processing for the AMS experiment. More precisely, the prediction of the replica value is “the prediction of the number of times that a replica corresponding to the identifier \( f \) would be accessed in the next \( n \) requests, based on the previous \( m \) requests (in a fixed future time window): \( V(f, m, n) = \sum_{i=1}^{m} P_i(f) \), where \( P_i(f) \) is the probability of receiving a request for file \( f \) at request \( i \).”

However, this approach is not applicable when looking at job statistics (all jobs including both production and analysis are considered) obtained from PanDA for the year 2011. Our analysis was applied to the data with the most popular data types such as: RAW (data retrieved from the detector), ESD and AOD (processed or reprocessed data from RAW), NTUP (derived data from AOD or ESD, to be used in the majority of end-user analysis). Results show that the prediction of data popularity is inaccurate due to the temporal decreasing nature of user interest of the corresponding data that is mostly caused by the production of new data (derived data of a new type or reprocessed data in new transformation conditions). The applied “Two-stage replica replacement algorithm” does not consider the popularity of the related data, and the popularity of data parameters, such as data type, but this information gives an important adjustment for the prediction of data access. Figure 1 shows the decreasing trend in the number of successful jobs (thereby the decreasing popularity of the input datasets) for two types of inputs, where inputs are:

• Datasets before reprocessing.
• Datasets after reprocessing: reprocessed datasets (ESD and AOD datasets) or datasets that are derived from reprocessed datasets (NTUP datasets).

These datasets are produced from the RAW dataset that was the most popular among RAW datasets based on information from DQ2 Popularity service on September 2011 [3]: “data11_7TeV.00184130.physics_JetTauEtmiss.merge.RAW”. The obtained data will help to track the behavior of derived datasets usage. The summary for the number of successful jobs in this example is shown in table 1.

**Table 1.** Number of successful jobs with non-reprocessed (initial) datasets and reprocessed datasets for “data11_7TeV.00184130.physics_JetTauEtmiss”.

| Data type | # jobs, with non-reprocessed datasets | # jobs, with reprocessed datasets | Ratio* (%) |
|-----------|--------------------------------------|-----------------------------------|------------|
| ESD       | 22 731                               | 18 756                            | 82.5       |
| AOD       | 21 726                               | 15 492                            | 71.3       |
| NTUP      | 18 376                               | 8 360                             | 45.5       |

* Ratio = \( \frac{I_{ReprDS}}{I_{NonReprDS}} \)

We would expect that after all campaigns of data reprocessing, there will be an increased “interest” in datasets that used to be popular in the past. However, our analysis shows that just a relatively small number of jobs are reproduced with the new version of the past-popular dataset after reprocessing.
Figure 1. Number of successful jobs (jobStatus="finished") with dataset inputs before and after reprocessing (dataset name is “data11_7TeV.00184130.physics_JetTauEtmiss”).
(a) ESD datasets; (b) AOD datasets; (c) NTUP datasets.

For the analysis of job statistics against the datasets the following parameters are used: job similarity coefficient, and similarity metric. Job similarity coefficient is a minimum threshold to filter
new jobs that are similar to original jobs with the certain similarity value which is an average of similarities between the same attributes of different jobs (examples of attributes: prodUserName, prodSourceLabel, e.g. “user”, processingType, e.g. “athena”, inputFileProject, e.g. “data11_7TeV”, nEvents, assignedPriority, etc.; examples of similarities: 1.0 – attributes are equal, 0.6 – attribute from the new job is on 60% the same as the attribute from the original job, etc.). This similarity value represents how similar the new job to the one of the original jobs. It helps to distinguish the reproduced jobs (jobs with datasets before reprocessing, which are reproduced with datasets after reprocessing) and the new jobs that most probably were not processed before. Similarity metric is calculated as the percentage of the reproduced jobs to the original jobs by using the jobs with the job similarity coefficient equal or larger than a certain value. For the above example, the similarity metric of ESD datasets is 23.4% with the job similarity coefficient 0.8 (80%). We conclude that it is more likely for reprocessed datasets to have higher popularity compared with datasets before reprocessing, but even this popularity decreases with time without being constant during any period of time. Reprocessed datasets are less distributed over computing centers, so the mechanism for the definition of destination sites for new replicas should utilize statistics of the popularity of the related datasets including datasets before reprocessing (local popularity, see section 3.1.1).

Information about the ratio between the number of jobs with datasets as inputs after reprocessing, including all campaigns of data reprocessing during 2011 (reprocessed datasets), and the number of jobs with datasets as inputs before reprocessing (non-reprocessed datasets) is shown in table 2.

| Data type | Job final state  | Ratio* |
|-----------|----------------|--------|
| ESD       | finished       | 0.23   |
|           | failed / cancelled | 0.27   |
| AOD       | finished       | 0.69   |
|           | failed / cancelled | 0.43   |
| NTUP      | finished       | 0.38   |
|           | failed / cancelled | 0.35   |

* Ratio = I_{ReprDS}/I_{NonReprDS}

3. Data models
There are three main sets of entities that have to be considered for data replication (all other parameters are defined as attributes for objects from these sets):

- Set of datasets - \{D_i\} (the term “dataset” is used for the job input or output object; in terms of PanDA jobs - an input can be either dataset that is a unit for data replication in ATLAS Data Distribution Management system, a container that consists of datasets, a dataset pattern, a container pattern, or a group of datasets / containers / patterns);
- Set of users - \{U_j\} (grid identifiers of PanDA job owners);
- Set of sites - \{S_k\} (the term “site” is used for any storage or computing element).

The main idea is to precisely identify all interactions between dataset objects only and between all the above three sets of objects (see figure 2). More precisely, dataset objects can interact with dataset objects; user objects can be both on the receiving and initiating end of interactions with dataset objects that reveals dependencies between popular datasets for users and popular users for datasets; similarly, users interact both ways with sites; however, site objects should not directly interact with dataset objects. This is what we call the user-oriented approach. This is further outlined below.
3.1. **Local and Global data popularity**

In our case, the definition of popularity is based on the popularity of certain datasets among users but not among sites; we propose a user-oriented approach for determining popularity. Future location of data requested by a user must be user- but not site-dependent (see figure 2). The site is chosen based on its cost and its popularity for a particular user. The site cost and popularity are defined as two of the main parameters for the site object definition for the data caching process. These parameters give some estimation about site reliability based on information from PanDA jobs.

3.1.1. **Data Local Popularity.** The local popularity of the dataset $D$ for the user $U$ is defined as:

$$D_{LP} = \frac{J_{DU}}{J_D}$$

where $J_{DU}$ is the number of jobs with the input dataset $D$ that are submitted by the user $U$, and $J_D$ is the total number of jobs with the input dataset $D$.

3.1.2. **Data Global Popularity.** The global popularity of the dataset $D$ is defined as:

$$D_{GP} = \frac{U_D}{U} \times \frac{J_D}{J}$$

where $U_D$ is the number of users who submitted jobs with the dataset $D$, $U$ is the total number of users, $J_D$ is the total number of jobs with the input dataset $D$, and $J$ is the total number of jobs.

3.1.3. **Site parameters.** For each user, weights are assigned to all sites, which store user data. The assignment of weights is based on two components: cost and popularity. These components are defined as:

$$S_{Cost} = \frac{F_{JUS}}{S_{JUS} + F_{JUS}} = \frac{F_{JUS}}{J_{US}}$$

$$S_{Popularity} = \frac{J_{US}}{J_S}$$

where $S_{JUS}$ is the number of successful jobs that are submitted by the user $U$ at the site $S$, $F_{JUS}$ is the number of failed jobs that are submitted by the user $U$ at the site $S$, $J_{US}$ is the total number of jobs that are submitted by the user $U$ at the site $S$, and $J_S$ is the total number of jobs at the site $S$.
3.2. **The Model of Datasets Interactions**

Popularity measure is one of the basic parameters that is used for the definition of the *Dataset Object* in the *Model of Datasets Interactions (MDI)*.

The MDI is a directed acyclic graph (see figure 3) that shows interactions between dataset objects: how a particular dataset object relates to other dataset objects. It represents how dataset objects are transformed through time (from RAW to NTUP dataset objects).

![Figure 3. The Model of Datasets Interactions.](image)

3.2.1. **The Dataset Object.** The Dataset Object (used as a job input) is a vertex in the MDI with the following parameters:

- The reference to the parent dataset (dataset that was as an input for jobs which have produced the current dataset).
- A list of ancestors (parent objects until the initial RAW dataset object).
- The global popularity and a list of users with corresponding local popularity values.
- A list of values containing a similarity metric between the new dataset object and each of previous dataset objects (with the same parent dataset object) that is a similarity metric between jobs with new dataset object and jobs with previous dataset objects (see section 2.1). It is represented as a percentage of reproduced jobs with new dataset as input compared to jobs with previous datasets as inputs (quantitative assessment of how close to each other datasets are in terms of jobs).

3.2.2. **The Object Link.** The Object Link is a directed edge in the MDI with the following parameters:

- References to the ancestor and the descendant dataset objects.
- A list of transformation attributes that leads from parents to children (represents the difference between parent and child dataset objects).
• The similarity metric between parent and child dataset objects (see section 2.1), i.e., the weight of the link. This value is significantly less than the value of any similarity metrics between children dataset objects, because the difference between the parent and child dataset objects is larger than the difference between children dataset objects.

3.3. Bayesian network modeling

A Bayesian network is a graphical model that encodes probabilistic relationships among variables of interest [6]. Condition parameters (set of variables $X=\{x_1...x_n\}$) are represented as a directed acyclic graph by encoding assertions of conditional independence. Based on the chain rule for Bayesian networks, the probability for $X$ is defined as:

$$p(X) = \prod_{i=1}^{n} p(x_i|pa_i)$$

where $x_i$ denotes both the variable and its corresponding node, $pa_i$ denotes the variables representing the parents of node $x_i$.

All previously defined models are used to represent interactions between objects and to identify the influence of one object to another. Parameters of these models, such as description of vertices and edges, are employed as corresponding conditions for more sophisticated model – a Bayesian network based model [6]. This model will help in calculating corresponding probabilities for data popularity: global popularity of dataset, local popularity of dataset per user, site popularity per user, dataset attributes such as configuration tags, and corresponding jobs per dataset. Condition parameters define a belief-network structure based on dependencies between them: $D_{LP}$ represents a matrix with elements as Data Local Popularity of the certain dataset for the certain user; $D_{GP}$ represents a vector of elements as Data Global Popularity of the certain datasets among all users; $S_{Cost}$ and $S_{Popularity}$ are used as matrices with elements as site cost/popularity for the certain user. As a result of this modeling we obtain the probability for a certain dataset to be in demand for the user analysis at a certain destination site.

3.4. Data classification

ATLAS data can be classified into one of the following categories based on their importance and popularity (i.e., the Data Temperature Scale): {hot, warm, cold, frozen, obsolete} (new popular, popular, unpopular, not used, and deleted datasets respectively). Each state has its own set of conditions; this categorization is utilized in our data popularity model. Data Temperature Scale corresponds to intervals of Data Global Popularity that can be tuned based on its distribution, so the value of $D_{GP}$ for a defined dataset will determine its state.

Thus probabilities for each possible data state, in common with information about sites, can be estimated based on Bayesian network model (using prior knowledge, see section 3.3) and used in ATLAS Data Caching as a control parameter for the replication of new datasets. The condition parameter such as data state will enhance the quality of obtained prediction, so that the probability of a certain dataset to be popular at a certain site can be supplemented with a set of probabilities of dataset to be in a certain data state. We are in the process of defining variables, which requires a more precise investigation of the system object interactions.

4. Conclusion

In this study we have investigated the temporal behavior of user access to datasets, we have defined dependencies between three main sets of objects: datasets, users, and sites. Experiments for analyzing the behavior of the data popularity have shown that classical probabilistic models are unfit to handle the interactions among our variables, that it is not able to describe conditional dependencies among variables.

We described how graphical models can encode deterministic relationships between crucial factors (dataset parameters such as data type, job parameters, site parameters, Data Local and Global Popularities, etc.) and final outcomes (cause-effect relationships), and we explained why they are
expected to give a better fit to data. We believe that the methodology of Bayesian networks is a key for the definition of the probability of data popularity, and further research work will be focused on modeling data placement for PanDA jobs using Bayesian networks.

References
[1] Klimentov A, Anisyonkov A, Krivashin D, Titov M 2009 ATLAS Distributed Computing Proc. 22nd Int. Symp. on Nuclear Electronics and Computing (Varna, Bulgaria, 7-14 September 2009) pp 45-48
[2] The ATLAS Collaboration, Maeno T, De K, Panitkin S 2012 PD2P: PanDA Dynamic Data Placement for ATLAS J. Phys.: Conf. Ser. This volume
[3] Molfetas A et al 2011 Popularity framework to process dataset traces and its application on dynamic replica reduction in the ATLAS experiment J. Phys.: Conf. Ser. 331 062018
[4] Klimentov A 2011 ATLAS Distributed Computing Challenges and Plans for the Future CERN Document Server ATL-SOFT-SLIDE-2011-552, https://cdsweb.cern.ch/record/1383026
[5] Tian T, Luo J 2007 A Prediction-based Two-Stage Replica Replacement Algorithm Proc. 11th Int. Conf. on Computer Supported Cooperative Work in Design (Melbourne, Australia, 26-28 April 2007) pp 594-598
[6] Heckerman D 1997 Bayesian Networks for Data Mining J. Data Mining and Knowledge Discovery 1 79-119

Acknowledgments
We would like to thank the PanDA team for providing the data for our analysis and for their continued support. We would like to express our appreciations to both CERN and UTA for the facilities provided to conduct this research work. Finally, we would like to thank I. Ueda for all the time he has spent to make this document better.