# GeStore – Incremental Computation for Metagenomic Pipelines

Edvard Pedersen[1] Nils Peder Willassen[2] Lars Ailo Bongo[1]
[1] Department of Computer Science, University of Tromsø
[2] Department of Chemistry, University of Tromsø
epe005@post.uit.no, nils-peder.willassen@uit.no, larsab@cs.uit.no

**Abstract**

Updating a metagenomics analysis compendium with respect to new meta-data from knowledge bases can provide biological insights not available during the initial analysis. However, the computational cost of such analysis makes it impractical to frequently update the compendium. This paper presents the GeStore system for incremental updates of compendium meta-data. GeStore does not require any changes to the analysis pipeline tools, and only minor changes to the analysis pipeline. GeStore provides a plugin framework such that a new meta-data source can easily be added. We demonstrate that GeStore can significantly reduce the computational cost for on-demand updates, thereby making it practical to frequently update metagenomics compendia with new meta-data.

## 1 Introduction

There is currently a great demand for DNA sequence analysis due to the great potential for novel biological insight such analysis can presents. Especially in metagenomics were it allows researchers to study microorganisms that are not cultivable [6]. These microorganisms may produce bioactives that can be developed into commercial products. New DNA sequencers may produce several hundreds gigabytes of data in one run, and may require a supercomputer for analysis, or many weeks of analysis on a smaller computer clusters. In addition the amount of output data from the DNA sequencers is increasing faster than both the increases in computation and storage capacity [3].

A metagenomics analysis typically requires integrating multiple datasets, including correlation of sequence data with meta-data from other data sources. The meta-data includes information such as biological function, the GPS coordinates (including depth, or height) and environmental features of the sample.

This information is essential to understand the biological content of the sequence data. The knowledge bases are frequently updated with new knowledge extracted from the published literature and experimental data. It is therefore often necessary to frequently

[1]This paper was presented at the NIK-2012 conference. For more information, see //www.nik.no/.

rerun the analysis to correlate the data with new meta-data, or to integrate new datasets into the analysis. In addition it may also be necessary to rerun the analysis if the analysis tools are updated or replaced. Frequent reanalysis of the data further increases the computational cost, often to the point where reanalysis is simply not done.

## 2 Incremental Meta-data Updates

To perform a metagenomics analysis different sets of tools are used depending on the type of information required by the analyst. These tools are generally arranged in a pipeline [12], where the output files of one tool acts as the input files for the next tool. In addition some tools use meta-data downloaded from one or more knowledge bases. When this meta-data changes, it often only introduces changes to a small portion of output data. But, without incremental updates it still requires the entire pipeline to be re-run. Incremental updates can therefore significantly reduce the computation time by only recalculating the relevant parts of the input-data.

We believe systems for incremental updates for metagenomic data have the following requirements; (i) efficient incremental update execution (ii) no modifications to the pipeline tools, since it is impractical to modify the many tools in use in metagenomic pipelines (iii) minimal changes to the job management and resource allocation system on the supercomputer or cluster, such that it is easy to deploy the system (iv) provide on-demand updates such that updates are not generated unless requested by the analyst (v) support most genomic analysis tools and run on most job management systems (vi) maintain a view of previous meta-data collections, to ensure experiment repeatability.

To our knowledge, no previous incremental update systems [8, 4, 10, 13] support all of the above requirements. Systems such as Nectar [8] rely on applications using a specific framework thereby not satisfying (v). InCoop [4] and Percolator [10] provide online data processing thereby not satisfying requirement (iv). In addition only few system, such as Nectar, maintain views of the old data, and most focus on detecting changes in the input data rather than meta-data.

## 3 GeStore Incremental Update System

The GeStore system provides incremental updates to metagenomics analysis pipelines. It reduces the storage size of the meta-data collections used by leveraging incremental update techniques combined with fine-grained control of how the tools access data, satisfying requirements (i) and (iv). In addition it reduces the storage requirements of the meta-data collections, while still maintaining a complete view of the meta-data collection, fulfilling requirement (vi). It also presents a minimal API, such that integrating the system with existing pipeline systems does not require large changes, in accordance with requirements (ii), (iii) and (v).

Incremental updates are done by generating meta-data collections for a certain timeframe, based on when the pipeline was last run, or re-using an existing incremental meta-data collection. These incremental meta-data collections are used by the tools to generate incremental results, which are combined with the existing results.

The main components of GeStore(Figure 1) are the **addDb** and **move** modules, and the **plugin** system. These modules use the Hadoop software stack (HBase [1], MapReduce [7] and HDFS [11]) for scalable data storage and processing.

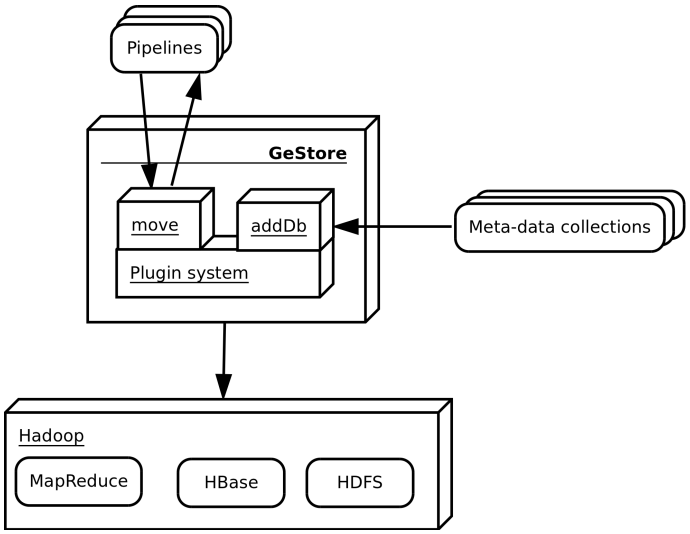
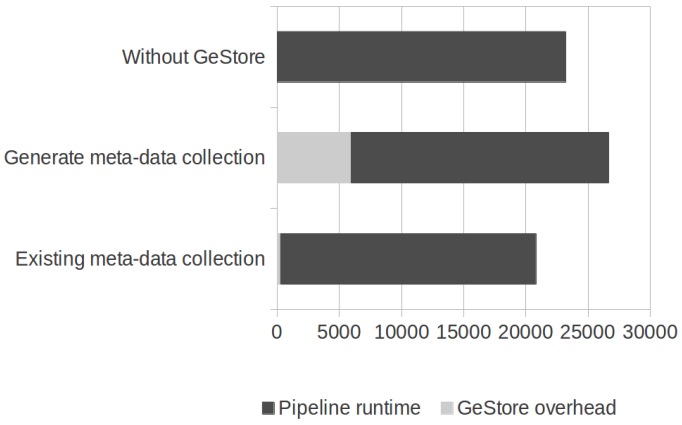


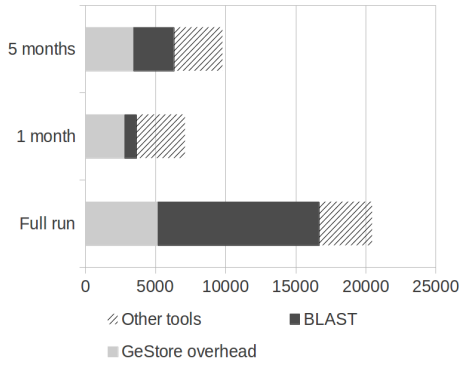

Figure 1: GeStore architecture

# 4 Iniatial Evaluation

In our initial evaluation we used GeStore to provide incremental meta-data updates in the (unpublished) GePan metagenomic analyis pipeline manager, which generates a pipeline based on selected tools and meta-data collections. The input data was a publicly available metagenomic dataset [5], and the meta-data was the widely used UniProt Knowledge Base [9]. The GePan pipeline includes the BLAST [2] tool, which is a computationally intensive tool used in many genomics data analysis pipelines. GePan uses the Sun Grid Engine for job management, hence we also used it for our experiments. In addition GeStore uses the Hadoop software stack as described above.




(2.1) Overhead of GeStore, shows the added runtime by using the GeStore system to generate incremental meta-data collections




(2.2) Runtime measurements for incremental updates, shows 65% runtime savings when doing single month of updates, and the benefit of on-demand updates

Figure 2: Performance measurements

Our experimental evaluation shows that GeStore can provide up to a 65% speedup for incremental updates in the GePan analysis pipeline (Figure 2.2), while introducing a low overhead (Figure 2.1). In addition we only had to change around 90 lines of code to add incremental updates for GePan.

# 5 Conclusions

Our contributions are threefold: (i) we have designed, implemented and evaluated the GeStore system for doing intermediate updates of metagenomic data, (ii) we developed a novel file based incremental update approach that does not require any modifications to pipeline tools and minimal modifications to the job management system, and (iii) we demonstrated the viability of incremental updates for metagenomic work, showing that computational resource requirements can be lowered when using incremental update techniques.

We believe that GeStore system can be used to update metagenomics analysis data on demand without requiring a supercomputer or days of computational time on a small cluster. It can therefore provide biological insights not available at the time of the initial analysis.

# 6 Acknowledgements

Thanks to Espen Robertsen and Tim Kahlke for help with the GePan pipeline, Jon Ivar Kristiansen for maintaining our cluster and Lars Tiede for his comments and insights.

# References

[1] Apache HBase. `http://hbase.apache.org/`. Retreived May 2nd 2012.

[2] ALTSCHUL, S. F., GISH, W., MILLER, W., MYERS, E. W., AND LIPMAN, D. J. 1990. Basic local alignment search tool. *Journal of Molecular Biology 215,* 3, 403–410.

[3] BAKER, M. 2010. Next-generation sequencing: adjusting to data overload. *Nature Methods 7,* 7, 495–499.

[4] BHATOTIA, P., WIEDER, A., RODRIGUES, R., ACAR, U. A., AND PASQUINI, R. 2011. Incoop : MapReduce for Incremental Computations. In *Proceedings of the 2nd ACM Symposium on Cloud Computing*. ACM Press, 7.

[5] BHAYA, D., GROSSMAN, A. R., STEUNOU, A.-S., KHURI, N., COHAN, F. M., HAMAMURA, N., MELENDREZ, M. C., BATESON, M. M., WARD, D. M., AND HEIDELBERG, J. F. 2007. Population level functional diversity in a microbial community revealed by comparative genomic and metagenomic analyses. *The ISME journal 1,* 8, 703–713.

[6] BOHANNON, J. 2007. Ocean study yields a tidal wave of microbial DNA. *Science (New York, N.Y.) 315,* 5818, 1486–7.

[7] DEAN, J. AND GHEMAWAT, S. 2008. MapReduce: simplified data processing on large clusters. *Communications of the ACM 51,* 1, 107–113.

[8] GUNDA, P. K., RAVINDRANATH, L., THEKKATH, C. A., YU, Y., AND ZHUANG, L. 2010. Nectar: automatic management of data and computation in datacenters. In *Proceedings of the 9th USENIX conference on Operating systems design and implementation*. OSDI'10. USENIX Association, Berkeley, 1–8.

[9] MAGRANE, M. AND THE UNIPROT CONSORTIUM. 2011. UniProt Knowledgebase: a hub of integrated protein data. *Database the journal of biological databases and curation 2011,* 0, bar009.

[10] PENG, D. AND DABEK, F. 2010. Large-scale Incremental Processing Using Distributed Transactions and Notifications. In *Proceedings of the 9th USENIX conference on Operating systems design and implementation*. OSDI'10 Series, vol. 2006. Google, Inc., USENIX Association, 1–15.

[11] SHVACHKO, K., KUANG, H., RADIA, S., AND CHANSLER, R. 2010. The Hadoop Distributed File System. *2010 IEEE 26th Symposium on Mass Storage Systems and Technologies MSST 0,* 5, 1–10.

[12] TANENBAUM, D. M., GOLL, J., MURPHY, S., KUMAR, P., ZAFAR, N., THIAGARAJAN, M., MADUPU, R., DAVIDSEN, T., KAGAN, L., KRAVITZ, S., RUSCH, D. B., AND YOOSEPH, S. 2010. The JCVI standard operating procedure for annotating prokaryotic metagenomic shotgun sequencing data. *Standards in genomic sciences 2,* 2, 229–237.

[13] TURCU, G., NESTOROV, S., AND FOSTER, I. 2008. Efficient Incremental Maintenance of Derived Relations and BLAST Computations in Bioinformatics Data Warehouses. In *Data Warehousing and Knowledge Discovery*, I.-Y. Song, J. Eder, and T. Nguyen, Eds. Lecture Notes in Computer Science Series, vol. 5182. Springer, 135–145.