# HadoopBase-MIP: Hadoop & HBase-based Toolkit for medical image processing

Shunxing Bao[1], Bennett A. Landman, Aniruddha Gokhale, Alan Tackett[1]
[1]Vanderbilt University, Nashville, TN 37203

## Introduction

- Medical imaging processing often involves large datasets. A traditional grid computing architecture like Sun Grid Engine (SGE) for processing these datasets requires moving data from a central storage location to be processed on computing nodes[1]. This can saturate a local network under certain conditions, creating a performance bottleneck.
- A solution is to distribute the data across computing nodes. The challenge is that substantial resources have been invested in creating existing algorithms, software tools, and pipelines, and there is a substantive cost associated with algorithm re-design specifically for big data.
- Medical image processing also involves multi-stage analysis. Due to the sequential nature of executing the analysis stages by traditional technologies, any errors in the pipeline are only detected at the later stages which cause highly compute-intensive re-execution in first stage.
- Our effort is a mix of research and experimental work to demonstrate applicability to medical imaging, which, to date, has not used a data-collocation computational model, and instead typically relies on monolithic data warehouses.

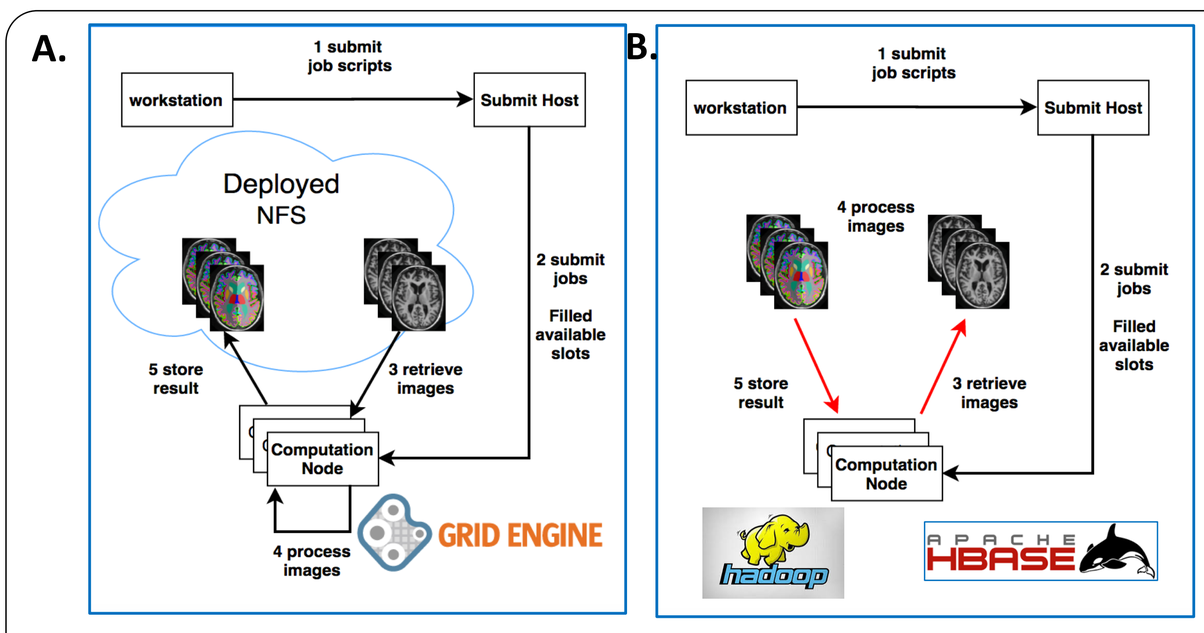


**Fig.1** Hadoop and SGE data retrieval, processing and storage working flow. (A)For SGE, each computation node retrieves the data within a shared NFS and stores the result back to the NFS. (B) The HBase Regionserver collocates with a Hadoop Datanode to fully utilize the data collocation and locality. The computational instructions can simply be sent to the data, and the data is then processed locally. Thus computing clusters using a distributed file system such as Apache Hadoop and HBase have potential to exploit.

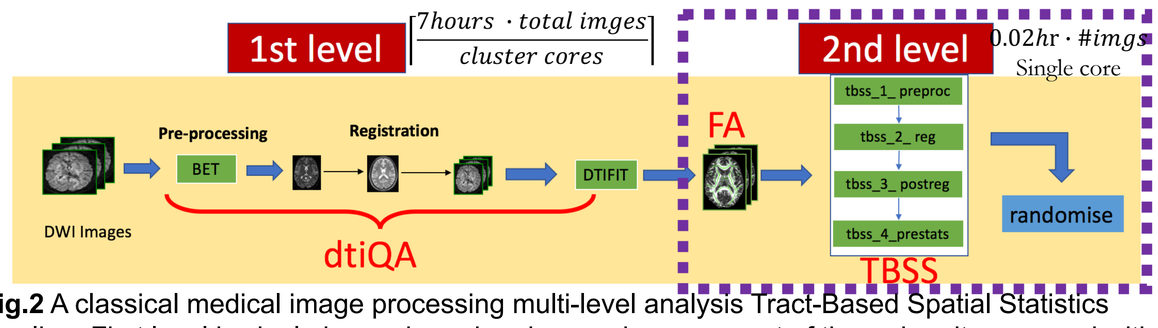


**Fig.2** A classical medical image processing multi-level analysis Tract-Based Spatial Statistics pipeline. First level is single image based and cause huge amount of time when it compared with second level group analysis.

## Methodology

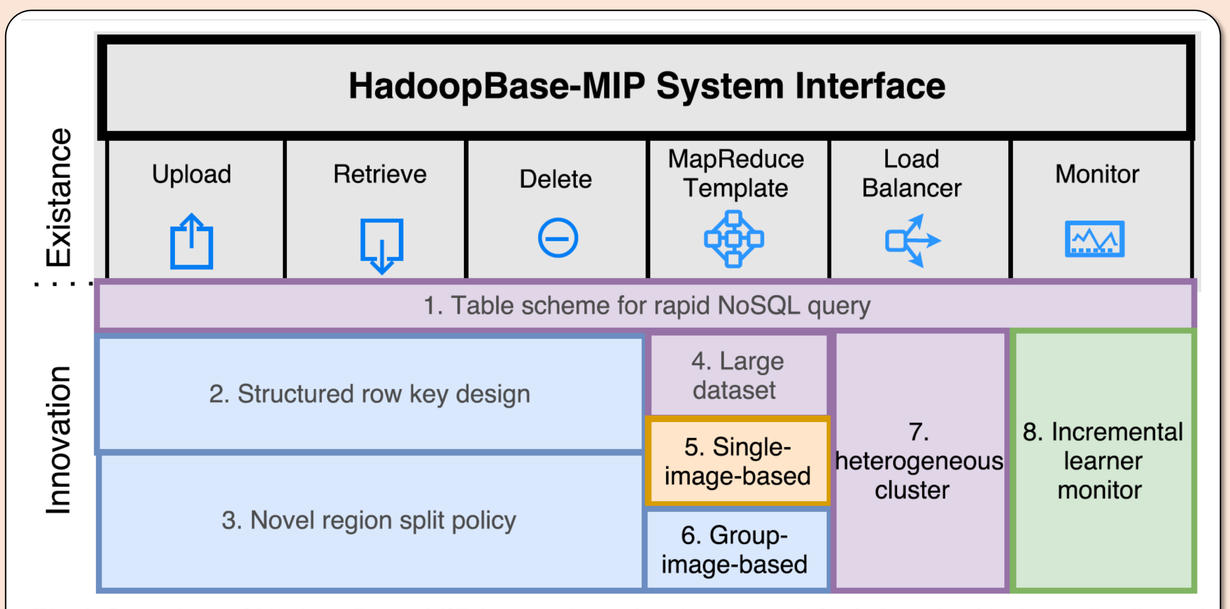


**Fig.3** Overview of HadoopBase-MIP innovations that improve default Apache Hadoop distributed file system, HBase MapReduce template and HBase data model for big data medical image processing.

**We summarized the innovations of HadoopBase-MIP system as follows:**

1. HBase Table design to enable fast data query and boot up the of MapReduce performance.
2. A hierarchical HBase key structure to accommodate nested layers of priority for data-collocation (logically and physically).
3. Customized HBase's region split policy of load distribution to optimally manage data collocation in the context of the hierarchical key structure.

4-6. MapReduce templates for different types of analysis: single image processing; group based analysis; large dataset summary statistics.

7. An off-line load balancer to better allocating data in a heterogeneous cluster.
8. A semi-automated, real-time quality assurance (QA) model monitor and checkpoint framework which aims to optimize the performance of medical image processing by finding anomalies in the first level processing in a timely manner thereby expediting the entire multi-level analysis.

## Experiment

**HadoopBase-MIP was setup in a private research cloud comprising a typical Gigabit network with 224 CPU cores on heterogenous machines.**

- Case 1: To investigate the performance of our HBase data model, we converted standard DICOM (9,910,000 files ,530GB )to NiFTI (8120 scan files) format conversion with three test scenarios using HBase and Hadoop and one with Network Attached Storage (NAS).
- Case 2: 5,153 T1 images (77GB) retrieved from normal healthy subjects gathered from [2], and we divided all images into several groups based on age / sex and promoted large dataset averaging to generate a population based template image. We used two HBase table schema cases and one case with Network Attached Storage (NAS).
- Case 3: Validating the effectiveness of our monitoring and check-pointing capability in the concurrent multi-stage analysis pipeline by incrementally conducting second stage analysis (as Fig.2). Our example has 423 input diffusion weighted images with 3.5 GB, and they would generate around 0.74 GB in total of 423 FA images.

## Results

- **Case 1**: The row key architecture improves throughput by **60%** over the "Naïve HBase". The custom split policy enforces data collocation to further increase throughput by **21%** over default split policy.
- **Case 2**: When averaging large subsets like all female and all male's T1 images, SGE spends about **3-fold** wall time and **6-fold** resource time more than proposed Hadoop time. As the size of subsets decreases, SGE's resource time also decreases, and proposed HBase table scheme design also generates similar trend, naïve table design scheme leads to opposite trend.
- **Case 3**: We conduct empirical evaluation of our framework and show that it provides **76.75%** less wall time and **29.22%** less resource time compared to the traditional approach without the quality assurance mechanism.

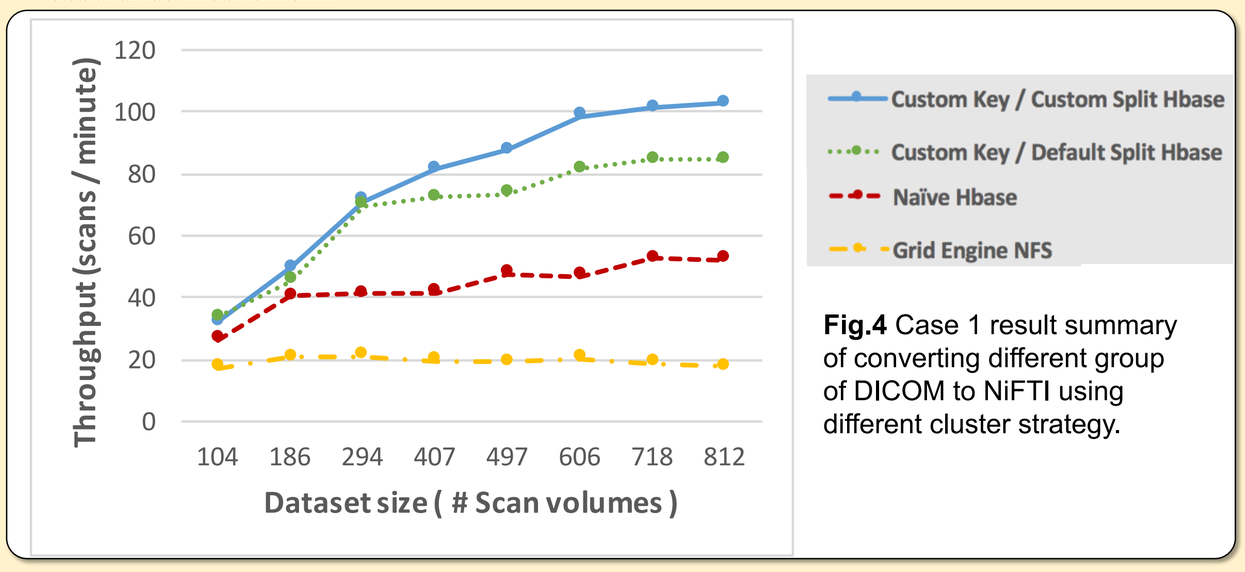


**Fig.4** Case 1 result summary of converting different group of DICOM to NiFTI using different cluster strategy.

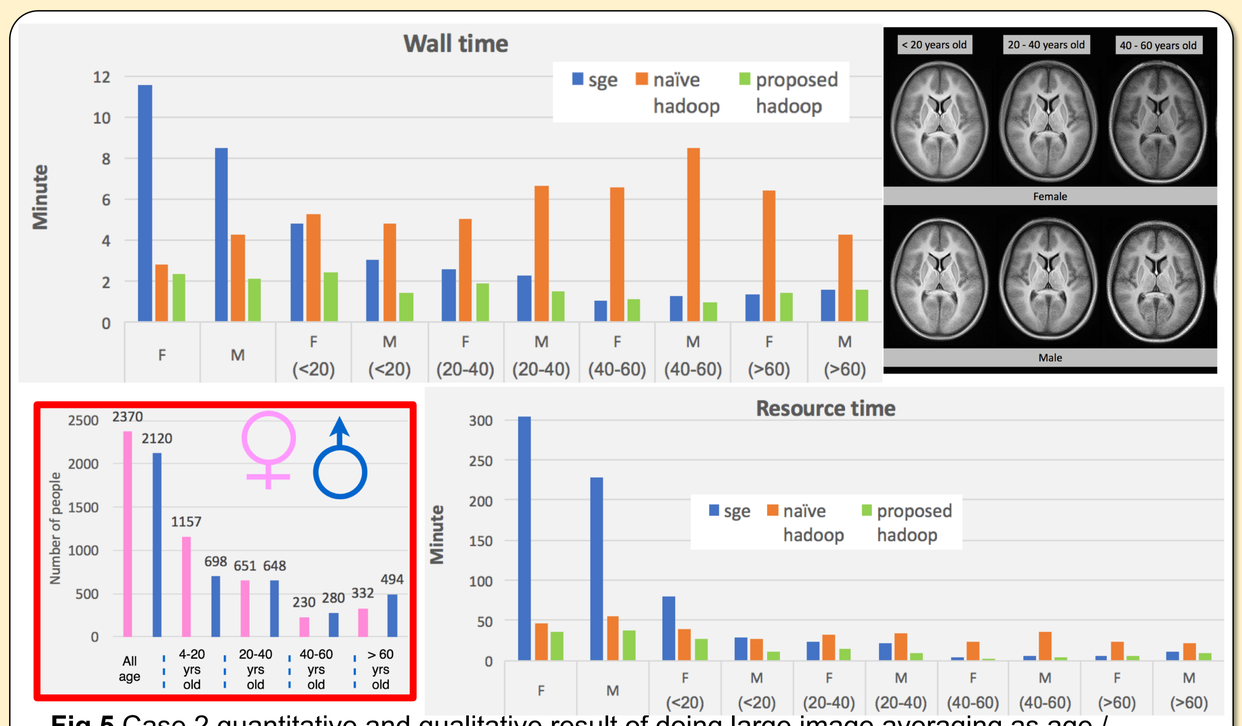


**Fig.5** Case 2 quantitative and qualitative result of doing large image averaging as age / sex based for population image template.

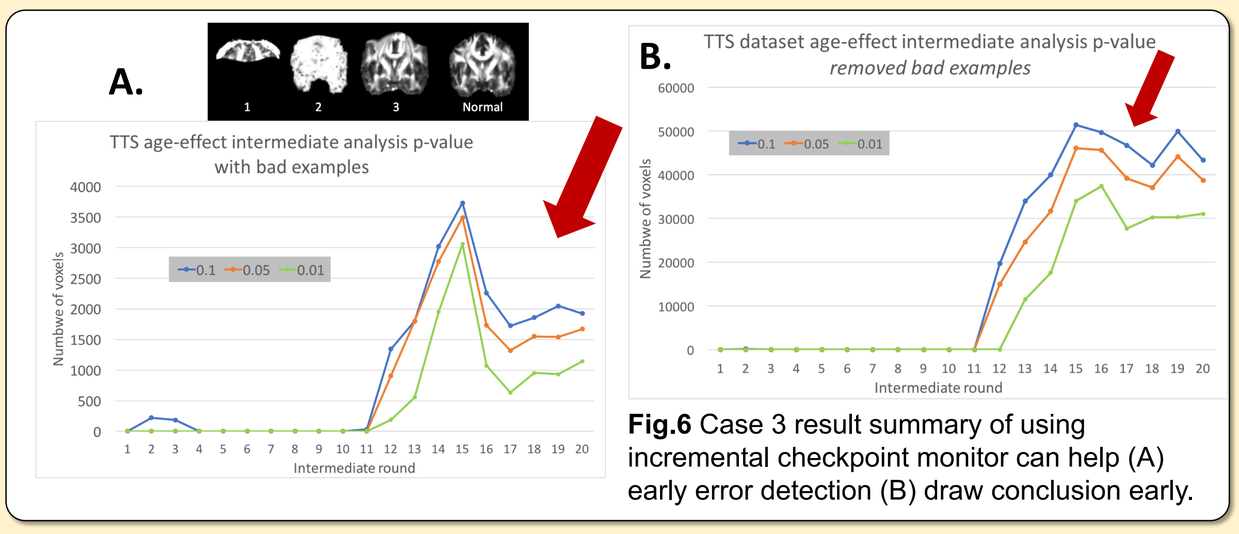


**Fig.6** Case 3 result summary of using incremental checkpoint monitor can help (A) early error detection (B) draw conclusion early.

## Conclusions

HadoopBase-MIP was implemented on a small, private data center, which includes the SGE. As the number of machines increases, NFS becomes nonviable with a single host, and distributed storage (e.g., GPFS) is commonly used on large clusters with 10+ Gbps networks. The proposed data and computation co-location solution is an alternative and could scale to well-more CPU-cores than beyond a GPFS solution on the same underlying network. Finally, we introduce a theoretical model to determine when performance improves by using proposed HadoopBase-MIP) and we empirically verify the accuracy of the model (Fig.7).

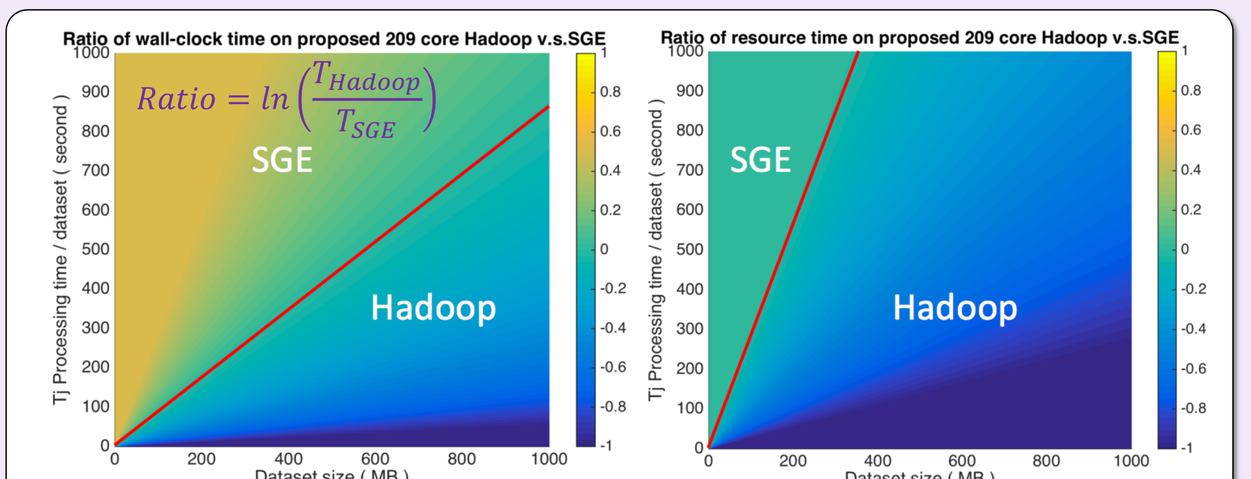


**Fig.7** Theoretical time ratio of Hadoop v.s. SGE based on core / data unbalanced cluster. The red lines in indicate the parameters for which Hadoop and SGE result in equivalent performance for the specified setup.

**We present design principles and empirical validation for using Hadoop & HBase that provides practical access to distributed imaging archives, integrates with existing workflows, and effectively functions with commodity hardware. We also how to utilize the system to do optimizing work for Medical image multi-level analysis.**

**SUPPORTED BY** ACCRE fellowship, NSF CAREER IIS 1452485 , ViSE/VICTR VR3029, Grant UL1 RR024975-01, Grant 2 UL1 TR000445-06

**REFERENCES** [1 ]W. Gentzsch, "Sun grid engine: Towards creating a compute power grid." 35-36.

[2] Y. Huo, K. Aboud, H. Kang, L. E. Cutting, and B. A. Landman, "Mapping lifetime brain volumetry with covariate-adjusted restricted cubic spline regression from cross-sectional multi-site MRI," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2016, pp. 81-88