Investigation Into the Use of Hardware Accelerators in Data Intensive Compute

CS310 Progress Report
David Richardson, 1314918

November 2015

This document details the progress made in the investigation into the use of hardware accelerators in data intensive compute. Section 1 reintroduces the project and its aims as well as gives a summary of the project's objectives. Section 2 identifies any existing research in this project's problem domain. Section 3 details the process for benchmark selection, as well as discusses any progress toward their successful execution. Section 4 reiterates the approach to project management as specified in the project specification, giving any necessary changes that have been brought to light by the work so far. Section 5 outlines any further work necessary for the project completion and any possible extensions to the project. Finally, Section 6 concludes this progress report by discussing the overall state of this project, as well as reiterating the key points that are to be completed to further its progression. The original specification documentation for this project is available in Appendix A.

1 Introduction

Hardware accelerators are specifically designed computer hardware that are designed to perform some forms of computation, such as floating point arithmetic, faster [1], [2] and more efficiently [3] than a general purpose CPU. These accelerators generally take the forms of General Purpose GPUs (GPGPUs) or Many Integrated Core (MIC) co-processors. It is becoming increasingly common to find hardware accelerators installed within the compute nodes of new compute clusters, and with this the possibility for their use is becoming more apparent. Despite this, the amount of research into their use within the paradigm of data intensive compute is underwhelming. The

integration of these hardware accelerators into data intensive compute programs, such as MapReduce jobs, could provide benefits such as a decrease in the total compute time and a reduction in the power consumed over the time a program is executing. Other possible benefits involve a reduced need to scale outwards to cope with the Tera- or Petabyte scale data sets that have come about from the data avalanche in areas such as bioinformatics [4]. These benefits are of interest to both academic and commercial applications, where it can reduce operational costs and reduce turn around time for compute workloads. Organisations that provide data-centric services such as Google or Facebook would also be able to enrich user experience with features that were previously not feasible due to slow compute times, increasing both value of service and profit.

1.1 Project Aims

The overarching aim for this project is to investigate and test the use of GPGPU and MIC co-processor accelerators in data intensive workloads, determining if their integration has significant improvements in compute and power consumption versus a CPU-only implementation. Their use has value for both academia and industry, where a reduction in compute time will generally lead to a reduction in operational costs incurred to run a data intensive program. It will also benefit infrastructure management companies such as Amazon, by increasing the performance per Watt of their compute nodes.

1.2 Summary of Objectives

The project has two main objectives that were outlined in the project specification documentation:

1. To understand if current benchmarking suites are suitable for hardware accelerated data analytics clusters.
2. To determine if accelerators can be used within data analytics with little modification to current software stacks or algorithm implementations.

Where the notion of a 'suitable' benchmark is a benchmark that tests a variety of work loads, makes use of as much of the hardware present in the system as possible like hardware accelerators; and can be scaled in input data set size.

2 Research Direction

With the project introduced, and its main aims and objectives discussed, the area of related research is now considered.

Research into the use of GPGPU and MIC co-processors within data intensive compute is limited at best, with very few technical reports or articles available.

2.1 Accelerating Breadth-First Search with Intel MIC Co-processors

Tao, Yutong, and Guang provide research into the application of the Intel MIC co-processor architecture to a Breadth-First Search (BFS) of a graph, a common workload in the realm of data intensive compute. Their research considers both native and offload optimisation techniques, giving the optimisation methodologies for both [5]. The native optimisation involves performing the BFS entirely on the co-processor and the optimisation techniques used involves the exploitation of thread- and data-level parallelism. The offload optimisation involves the partitioning of the tasks within the workload as well an optimisation of the data communications between CPU and co-processor. They found that their native optimisation could run up to 3.4x faster on two Intel Xeon Phi Knight's Corner than when run on two Intel Xeon E5-2670. Their offload optimisation results in a speed up of up to 1.67x, in comparison. The algorithm used for data communications optimisations in the offload optimisation also benefits greater from larger graph sizes.

3 Benchmarking Progress

With the nature of this project being mostly based around investigation and research, it is quite hard to measure its progress. However, it is possible to measure progress with regards to the timetable outlined in the project specification, where it lists the key phases to the project. With this in mind, the progress of this project is now discussed.

3.1 Benchmark Selection

There are a number of benchmarking suites available to test, benchmark, and classify data intensive compute clusters. With the paradigm of data intensive compute lacking a 'standard' benchmarking suite like LINPACK is to high performance computing, many of these benchmarks also compete to become such a standard. For this project I will be selecting one benchmarking suite for use to compare the effect of the integration of hardware accelerators into them.

Through my investigation into these benchmarks, a few observations have been made:

1. Most benchmarking suites come with their own scalable data generators.
2. All benchmarking suites that have been considered are developed for MapReduce or similar compute workloads.
3. All benchmarking suites investigated have not been built with the consideration for the use hardware accelerators.

These observations can be used to conclude about objective 1 that was outlined in the project specification. This is that the current suite of benchmarks are not suitable for hardware accelerated data intensive compute clusters. This is due to the lack of consideration within the benchmarking suites, when developed, for the use of hardware accelerators like GPGPUs and MIC co-processors.

With this in mind, the benchmarking suites that were considered are now discussed and compared.

3.1.1 Graph500

The Graph500 is an initiative to establish a set of large-scale benchmarks for data intensive applications, being backed by both academia and industry experts [6]. At present, the Graph500 benchmark has only one workload that can be separated into two kernels: generating a graph from an edge list, and a breadth-first search of the generated graph [7]. Of these, the second kernel is used as a performance measure of the host system. The second kernel's performance is measured in Traversed Edges per Second (TEPS), providing a standard unit for comparison akin to LINPACK with Floating Point Operations per Second (FLOPS). The reference implementation provided by the

Graph500 is written in C and can come in sequential, multi-threaded and multi-process flavours that use standards like OpenMP, XMT and MPI [8].

3.1.2 BigDataBench

BigDataBench is a data analytics benchmarking suite created at the ICT, Chinese Academy of Sciences, with backing from industry partners such as Huawei. The benchmarks in this suite abandon sequential and multi-threaded workloads, that would typically use OpenMP or similar libraries, for scale-out [9] workloads that are designed to better represent the distributed nature of data intensive compute. The benchmarks themselves in this suite are derived from a common subset of 'dwarf' workloads, such as social network graph analysis or word multimedia analytics [10]. These benchmarks are implemented with MapReduce in mind, using associated frameworks such as Apache Hadoop or Spark. Other implementations are also available that use MySQL and MPI for inter-node communications [9].

3.1.3 Intel HiBench

Intel's HiBench is a suite of 10 Hadoop-based MapReduce benchmarks that are separated into two categories: synthetic micro-benchmarks and real-world applications [11]. The suite also takes into account the following 5 system characteristics when benchmarking and classifying a system:

- Job running time.
- Number of tasks per minute or job throughput.
- HDFS bandwidth.
- Utilisation of system resources like CPU, Memory and I/O.
- Data access patterns

The suite itself only provides implementations to use Apache's Hadoop framework. The real-world workloads cover web search, machine learning and analytical querying on large data sets. The micro-benchmarks cover basic jobs such as data sorting or extraction of information about a large data set. The suite also includes a benchmark to help determine the aggregated bandwidth that is delivered by HDFS [12].

3.1.4 Comparison

With all the benchmarks considered for this project outlined above, they are now compared and a suite will be selected.

Whilst the Graph500 benchmark is backed by both industry leaders and academics alike, its suite contains only one benchmark that is not wholly representative of data intensive workloads. Its implementation using C and MPI would make the use of hardware accelerators easier on more traditional compute oriented clusters. However, due to Chiron's architecture and chosen software stack, the use of MPI and/or OpenMP would not suit MapReduce and thus the suite will not be used for this project.

BigDataBench and Intel HiBench both provide extensive suites of benchmarks that can be separated into micro-benchmarks and workloads that are representative of what you would find in use in the real world. BigDataBench has the overall larger number of benchmarks available when compared to HiBench, and the benchmarking suite will also test more than just Apache Hadoop. It doesn't, however, provide the in-depth characterisation of the system that HiBench provides. BigDataBench also has a larger amount of required software, which may make it infeasible to run some benchmarks on Chiron. With these considerations, Intel HiBench will be used for this project based upon its MapReduce workloads and in-depth benchmark reporting.

3.2 Benchmark Running

With the selection of the benchmark suite now complete, it is being used used to provide a baseline performance metric for comparison against the hardware accelerated implementations to come. There have been a few issues found with the lack of software libraries and with the configuration of Chiron when starting the benchmarking procedure. These issues were reported to the Centre of Scientific Computing and were resolved, with little effect to the timetabling of the project due to the provisioning of overrun buffers in the task durations. Only a small subset of benchmarks will be selected from the suite for execution and then adaptation to use hardware accelerators. This is to make it feasible to complete the project within the time given in the timetable.

Whilst benchmarking, it has also become apparent that the configuration of HDFS on Chiron will make a possible significant impact on the benchmark results. This is because the HDFS volume is partitioned into an SSD

partition and a HDD partition, with each differing in total storage size as well as read/write speeds. Due to the nature of data intensive compute being bounded by the speed of I/O, it is not beyond reason that the choice of storage medium would have an effect on results. This can be shown by difference in data access speeds for HDDs using SATA III and SSDs using the M.2 standard with 4x PCI-E 3.0 lanes: 31.56Gbps for M.2 [13] and 6.0Gbps for SATA III [14]. This results in SATA III being 5.26x slower than M.2 and its 4 PCI-E 3.0 lanes. On top of this, the node interconnects in Chiron are Mellanox InfiniBand with a maximum throughput of 56Gbps [15], showing that the network will not bottle neck I/O operations and the bottle neck in fact resides with the chosen storage media.

4 Project Management

Having discussed the progress made in the project, attention is now turned to the project's management. The original project management techniques will be reflected upon and any changes to the project timetable or risk assessment will be considered.

4.1 Timetable

In the original project timetable, it was assumed that Chiron would be near or in a production state. With Chiron remaining in a pre-production state, however, it is not without configuration errors and has a lack of user documentation. These issues will, as a result, add possible delays to the project as it moves forwards and the use of untested system components starts to occur. Fortunately the project timetable as given in the specification documentation has some contingencies built into it, in the form of task padding zones marked in red, as well as extra amount of time that allows the project to overrun without issue. Figure 1 shows where the project is currently, with any complete tasks in violet. The overrun allowances for benchmark testing and accelerated benchmark testing and analysis have been extended, as well.

4.2 Risk Assessment

In an ideal world, Chiron would have had user documentation generated and have been tested thoroughly. However, due to it being in a state of

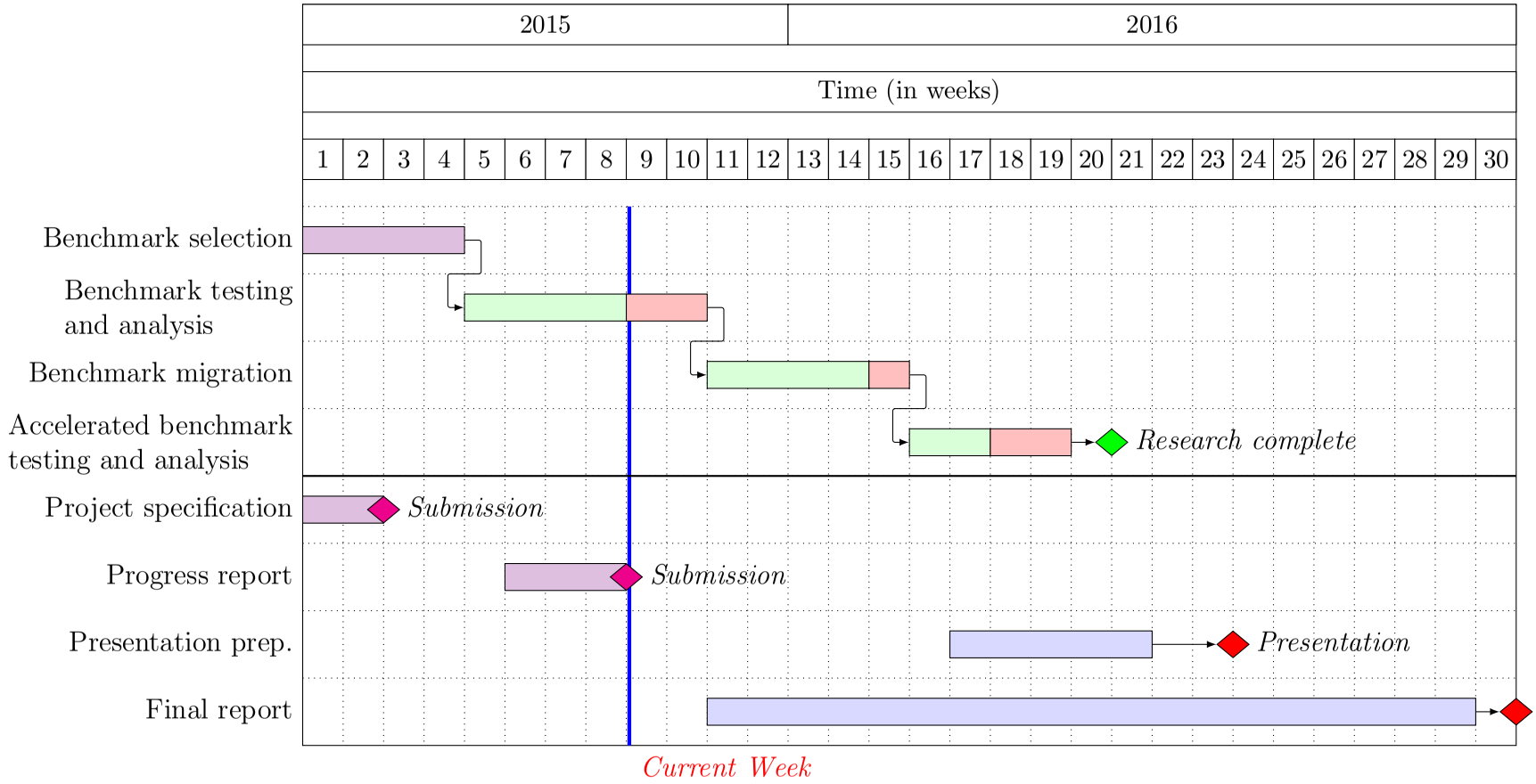

Figure 1: Revised project timetable from week 1 term 1 to week 1 term 3

pre-production, it is necessary to add the following risks to the risk matrix provided in the project specification documentation:

- A confuration error with Chiron's software stack.
- Software required to run a benchmark is not installed.

The amended risk matrix, with the risks mentioned above, is shown in figure 2.

Risk	Severity	Likelihood	Mitigating Action(s)
Chiron unavailable	Severe	0.01%	Locate suitable replacement for use in testing — replacement should have similar feature set to Chiron.
Benchmark code unavailable	Severe	5%	Check internet archives for possible location of older version. Find other suitable benchmarks.
Networking failure	Moderate-Severe	5%	Temporarily locate to different area to use a different network.
Project leader falling ill	Moderate	10%	Do work that can be done without further risk to health.
Configuration error with Chiron	Moderate	15%	Report the bug using the Centre of Scientific Computing's BugZilla bug tracker. Attempt other work that doesn't depend on that particular software stack.
Required software for benchmark not installed	Moderate-Severe	5%	Report the missing software to the Centre of Scientific Computing. In event of no resolution, another benchmark will be selected.

Figure 2: Risk matrix that associates possible risks with severity and mitigating actions

5 Further Work and Project Extensions

5.1 Further Work

Benchmark suite selection for this project has been finalised, and the execution of those benchmarks is under way. With this in mind, there are two remaining tasks to be completed: Benchmark migration and accelerated benchmark testing and analysis. This section will detail the remaining tasks that are to be fulfilled as part of the project's completion.

5.1.1 Benchmark Migration

The next step after the completion of benchmarking without hardware acceleration is to then port these codes to the hardware accelerator platforms outlined within the project's specification documentation. This will involve identifying areas that are most suited to the accelerator's architecture. After this identification process has finalised, the benchmark will be re-developed with accelerator-specific code and this code will then be tested. Once the code has been migrated to the accelerator, re-integration into the MapReduce model will take place.

5.1.2 Accelerated Benchmark Testing and Analysis

After benchmark migration has been completed, a similar approach to the benchmarking procedure for non-accelerated benchmarks will take place. This will involve the execution of the accelerated benchmarks using both solid state and hard disk storage options. The input size for these benchmarks will also be varied as to provide an idea of how the accelerated solution(s) scale with data set size. This scaling may also show any points where data communications may oversaturate the accelerator nodes and overall reduce performance.

5.2 Project Extensions

Whilst this project is in an area of research, it is possible that it can be put into commercial use. An example of this would be to highlight areas for migration to hardware accelerators for existing data intensive programs. The

extensions of this project listed below reflect some of the countless possibilities in which it could be used.

Additional Accelerator Types
: The research in this project is aimed only at GPGPU and MIC coprocessor hardware accelerators, but other types such as Fully Programmable Gate Array (FPGA) or Application-specific Integrated Circuit (ASIC) accelerators could be used as well. These have the benefit of being highly optimised for a specific task, although this also acts as a limitation as ASICs cannot be reprogrammed to be used in another type of workload, and FPGAs have a large development overhead associated with them in long program compilation times and difficulty with debugging.

Infrastructure Design
: This project's outcome could inform infrastructure providers such as Amazon or Google about the benefits to installing hardware accelerators into their cluster compute nodes. This could reduce operational costs through the increased power efficiency of hardware accelerators. Compute time could also be drastically reduced, thus the number of users of the service could increase. Finally, the total number of compute nodes required for the same performance could drop, resulting in a reduced initial investment when purchasing infrastructure.

Additional Workload Testing
: This project only covers a small subset of the available workloads and benchmarks for data intensive compute. Other workloads could be examined and tested on hardware accelerators, providing a broader insight to the applicability of these accelerators in the data intensive compute paradigm. This could further guide researchers and organisations alike with their choices to start utilising the hardware accelerators on their chosen compute clusters.

6 Conclusion

This project is running to schedule thanks to contingencies that were built into the original project timetable, as outlined in the project's specification documentation. It was also made possible to extend these padding zones due to the timetabling of the project being short. The benchmark selection process has also brought to light that the current set of available benchmarking

suites haven't been developed with the capability to utilise hardware accelerators when they are present in the system or cluster, addressing one of the objectives of this project.

With the only measure of progress in the project being the timetable, and with all tasks currently on time, the progress that has been achieved for this project has been successful. The key points of progression so far include:

- Benchmark suite selection.

The next stages for the progression of this project are:

- Finalise benchmark testing and analysis.
- Benchmark migration.
- Accelerated benchmark testing and analysis.

References

[1] F. Bensaali, A. Amira, and A. Bouridane. "Accelerating matrix product on reconfigurable hardware for image processing applications". In: *Circuits, Devices and Systems, IEE Proceedings* 152 (June 2005).

[2] Jr. Gerald A. Hanwerck. *Accelerating Quantitative Financial Computing with CUDA and GPUs*. Mar. 2013. URL: `http://on-demand.gputechconf.com/gtc/2013/presentations/S3373-Accelerating-Quantitative-Financial-Computing.pdf`.

[3] S. Huang, S. Xiao, and W. Feng. "On the energy efficiency of graphics processing units for scientific computing". In: *IEEE International Symposium on Parallel and Distributed Processing* (May 2009).

[4] Doug Howe et al. "Big data: The future of biocuration". In: *Nature* 455 (Sept. 2008).

[5] Gao Tao, Lu Yutong, and Suo Guang. "Using MIC to accelerate a typical data-intensive application: the Breadth-first Search". In: *IEEE International Symposium on Parallel & Distributed Processing Workshops* 27th (2013).

[6] The Graph500 Project. *The Graph500: Brief Introduction*. 2010. URL: `http://www.graph500.org/`.

[7] The Graph500 Project. *The Graph500 Specification*. Sept. 2011. URL: `http://www.graph500.org/specifications`.

[8] The Graph500 Project. *The Graph500 Reference Implementations*. 2012. URL: `http://www.graph500.org/referencecode`.

[9] Chinese Academy of Sciences ICT. *BigDataBench Homepage*. Oct. 2015. URL: http://prof.ict.ac.cn/BigDataBench/.

[10] Wanling Gao, Chunjie Luo, Jianfeng Zhan, et al. "Identifying Dwarfs Workloads in Big Data Analytics". In: *CoRR* abs/1505.06872 (2015). URL: http://arxiv.org/abs/1505.06872.

[11] Shengsheng Huang, Jie Huang, Jinquan Dai, et al. *The HiBench Benchmark Suite: Characterization of the MapReduce-Based Data Analysis*. Tech. rep. Intel China Software Center, Shanghai, 2010.

[12] Shengsheng Huang, Jie Huang, Yan Liu, et al. *HiBench: A Representative and Comprehensive Hadoop Benchmark Suite*. Tech. rep. Intel Asia-Pacific Research and Development, 2012. URL: https://software.intel.com/sites/default/files/blog/329037/hibench-wbdb2012-updated.pdf.

[13] Jason Lawley. *Understanding Performance of PCI Express Systems*. Tech. rep. XILINX, Oct. 2014.

[14] The Serial ATA International Organisation. *Serial ATA Revision 3.0*. June 2009.

[15] Mellanox Technologies. *SwitchX 36-Port QSFP FDR InfiniBand System Hardware User Manual*. 2013. URL: http://www.mellanox.com/related-docs/user_manuals/SX60XX_User_Manual.pdf.

Appendices

A Investigation Into the Use of Hardware Accelerators in Data Intensive Compute Specification

The following 12 pages consist of the original specification document as submitted to Tabula in Week 2 of Term 1, 2015

Investigation Into the Use of Hardware Accelerators in Data Intensive Compute

CS310 Project Specification
David Richardson, 1314918

October 2015

This document presents a project on the research into the use of hardware accelerators in the data intensive compute paradigm. Whilst hardware accelerators are pervasive within the compute intensive paradigm of parallel computing, the use of them in data intensive compute remains an area of little research. Section 1 of this document provides the motivation for this project. Section 2 gives background information on the two key areas; data intensive compute and hardware accelerators. Section 3 outlines the objectives for this project. Section 4 explains the methodologies to be used throughout the project. Section 5 will discuss how the project will be managed. Section 6 will outline any legal, ethical, social or professional issues that this project will cover. Finally, Section 7 will conclude this document.

1 Introduction

Data analytics and the paradigm of data intensive compute are areas of parallel computing that are growing at enormous rates. The data avalanche in recent years has enabled ever-deepening study and research into areas such as biology, as well as commercial use such as Google's Knowledge Graph. The increase in compute power required to cope with this data avalanche is usually met with a scale-outwards approach, resulting in data centres with thousands of compute nodes. The research into the use of hardware accelerators here is underwhelming, and most software stacks do not have suitable provisions or APIs for their use. The applications of hardware accelerators could reduce program execution times, as well as increase power efficiency, being of interest to both academic and commercial applications. End users of services such as

Google or Facebook could benefit from this due to new metadata obtained from previously slow computations.

1.1 Project Aims

The underlying aim for this project is to investigate the use of hardware accelerators within data intensive compute. Their use within data intensive compute has value for both scientific and commercial areas. For example, warehouse scale compute facilities or data centres could drastically reduce the power requirements for the same computational capabilities. Results of compute jobs could be determined faster due to a reduction in compute time, to the benefit of scientific researchers.

2 Background

Having discussed the aims of my project, I will now go on to address the background knowledge required for this project. This will include explanations of data intensive computing, hardware accelerators, the Chiron data analytics cluster and BigDataBench.

2.1 Data Intensive Compute

Data intensive computing is an emerging computing paradigm [1] designed to deal with the processing of petabyte-scale datasets. It combines high performance computation, massive data storage, high bandwidth access, and high-speed local and wide area networking [2] to address problems that were previously thought to be infeasible or impossible [3] because the compute time would have previously been too long. The paradigm is dominated by the use of distributed computing clusters, and cloud computing has facilitated its use on a large scale [3]. Data intensive compute typically uses Apache's Hadoop, an implementation of Google's MapReduce distributed batch computing white paper [4]. Other batch computing implementations are also available, such as LexisNexis' High Performance Computing Cluster (HPCC). Other compute approaches to the paradigm are also available such as Apache Storm, which allows for realtime data stream computation [5], and Apache Spark allows for in-memory data compute [6]. These different

approaches are usually combined into a software stack to allow for a comprehensive suite of tools for data analysis.

The use of data intensive compute is diverse, ranging from the sequencing of genomes through genomics [7] to gain understanding of how genetic traits can alter suceptibility to diseases, to the analysis of social networks to produce further information about people and their connections to others to create a more personalised experience.

2.2 Hardware Accelerators

Hardware accelerators are specialised computer system components designed to complement the general purpose CPU within the system. These accelerators provide a wide range of benefits over a general purpose CPU architecture, such as x86. These benefits include increased power efficiency [8], measured as FLOPS/Watt, as well as significantly increased performance with specific workload types. An example of this is matrix product calculations [9]. These characteristics has resulted in their uptake within the compute-intensive workloads, such as those in computational physics [10] and quantitative finance [11]. The most common types of hardware accelerators in use today are general-purpose GPU (GPGPU) and many integrated core (MIC) co-processors. Some examples of these are the Nvidia Tesla GPGPUs and the Intel MIC Xeon Phi co-processors. Other accelerator types are also available, such as Fully Programmable Gate Array (FPGA) cards from Altera.

Hardware accelerators have many benefits, but they can be underutilised if present in a system. This is usually a result of the accelerators being an afterthought in development or they are being used in workloads that they are not suited for. This can waste power, reducing their efficiency and overal effectiveness as part of a compute system.

2.3 Chiron

Chiron is a compute cluster at the University of Warwick designed for data intensive compute [12]. It provides the capability to perform traditional MapReduce computation, as well as data streaming and in-memory analytics. It also has 2x Nvidia Tesla K40 GPGPU nodes, 2x Intel Xeon Phis nodes, and 2x Nallatech 395 FPGA nodes. The cluster is configured using

Apache YARN as a base, with the Hadoop File System (HDFS) for a distributed filesystem. MapR Hadoop is then used along with Apache Storm for bulk data analytics and data streaming, respectively.

2.4 BigDataBench

BigDataBench is a data analytics benchmarking suite created at the ICT, Chinese Academy of Sciences, along with industry partners such as Huawei. The benchmarks abandon typical sequential and multithreaded workloads for scale-out [13]. workloads that are designed to better represent the distributed nature of data analytics.

The benchmarks themselves are designed around a set of commonly used workloads that represent common use cases of data analytics. Some examples of these are social network and search engine graph analysis, multimedia analytics and bioinformatics. The benchmarks are also implemented with different systems to test as broad a range of a system as possible. These implementations range from Apache Hadoop or Spark, to MySQL, to C-based programs that use MPI [13]. The suite also comes with its own data generation tool, designed to create huge data sets of different types under controllable generation rates [14].

3 Objectives

The objectives for my project are two-fold:

1. To understand if current benchmarking suites are suitable for hardware accelerated data analytics clusters.
2. To determine if accelerators can be used within data analytics with little modification to current software stacks or algorithm implementations.

For objective 1 it is important to define what the notion of suitability is. For the purpose of this project, a benchmarking suite is suitable if it meets the following criteria:

- The suite tests a variety of workloads with differing characteristics.
- The workloads may make use of accelerators if they are present in the system.
- The size the input data set for the workloads can be varied or scaled.

4 Methodology

My methodology for this project can be split into two parts: research and software development.

My software development methodology will be of a test-driven nature. This will allow for thorough testing of any (re)implementation of data analytics algorithms. The methodology will also be of an agile nature, allowing for changes to be made in the project during any software development stages that may arise [15]. The specific methodology to be used here will be Extreme Programming (XP) [15], allowing for the interleaving of development and testing of any software that is required. XP does require that the customer be involved every few weeks in the development cycle, but as this project is focused on research this is not feasible. Meetings shall also be set up during development on a regular basis with the project supervisor to make the most of this agile methodology.

For my research, I will be focusing on the Apache Hadoop software stack, which makes extensive use of the Java programming language though has support for other languages such as C++ [16]. I will also focus on GPGPU and MIC co-processors for the hardware accelerators in this project. These accelerators typically use C/C++ for programming, which may necessitate the use of the Java Native Interface (JNI) to execute any algorithm implementations designed for these accelerators.

My research methodology will start by selecting industry recognised benchmark(s) such as the BigDataBench benchmark for study. These benchmarks will have their data analytics workloads executed on the Chiron data analytics cluster to form a baseline for comparison. From there, I will attempt to make use of GPGPU and MIC accelerators within these benchmarks. From there I will benchmark the accelerated algorithms with the same data input sizes. Finally, I will compare the different benchmarks and draw conclusions with regards to the original objectives stated in section 3.

5 Project Management

Now that the discussion of the project's aims, objectives, requirements and methodologies has been completed, attention is now to be given to the project management aspect of the project. More specifically, this section will discuss the project's timetable, the resources required and give a risk assessment of

the project, providing insight into any risk mitigations or actions to be taken.

5.1 Timetable

Figure 1 provides an overview of the project timeline. It includes the official project deadlines, as well as outlines the core stages of the project: benchmark selection, benchmark testing and analysis, benchmark migration, and accelerated benchmark testing and analysis. The benchmark migration stage may also contain a sub-stage for software development which will run at the same time. The timetable also provides task dependencies in the form of directional arrows between stages.

- Benchmark selection: 5th October 2015 to 1st November 2015
- Benchmark testing and analysis: 2nd November 2015 to 30th November 2015
- Benchmark migration: 1st December 2015 to 31st January 2016
- Accelerated benchmark testing and analysis: 1st February 2016 to 14th February 2016
- Project specification write up: 5th October 2015 to 14th October 2015
- Project specification submission: 15th October 2015
- Progress report write up: 15th November 2015 to 28th November 2015
- Progress report submission: 30th November 2015
- Presentation preparation: 8th February 2016 to 7th March 2016
- Presentation: 7-9th March 2016
- Final report write up: 10th January 2016 - 21st April 2016
- Final report submission: 28th April 2016

5.2 Tools & Resources

Third party tools will be used wherever possible to speed up the process of research as well as its documentation. This will include:

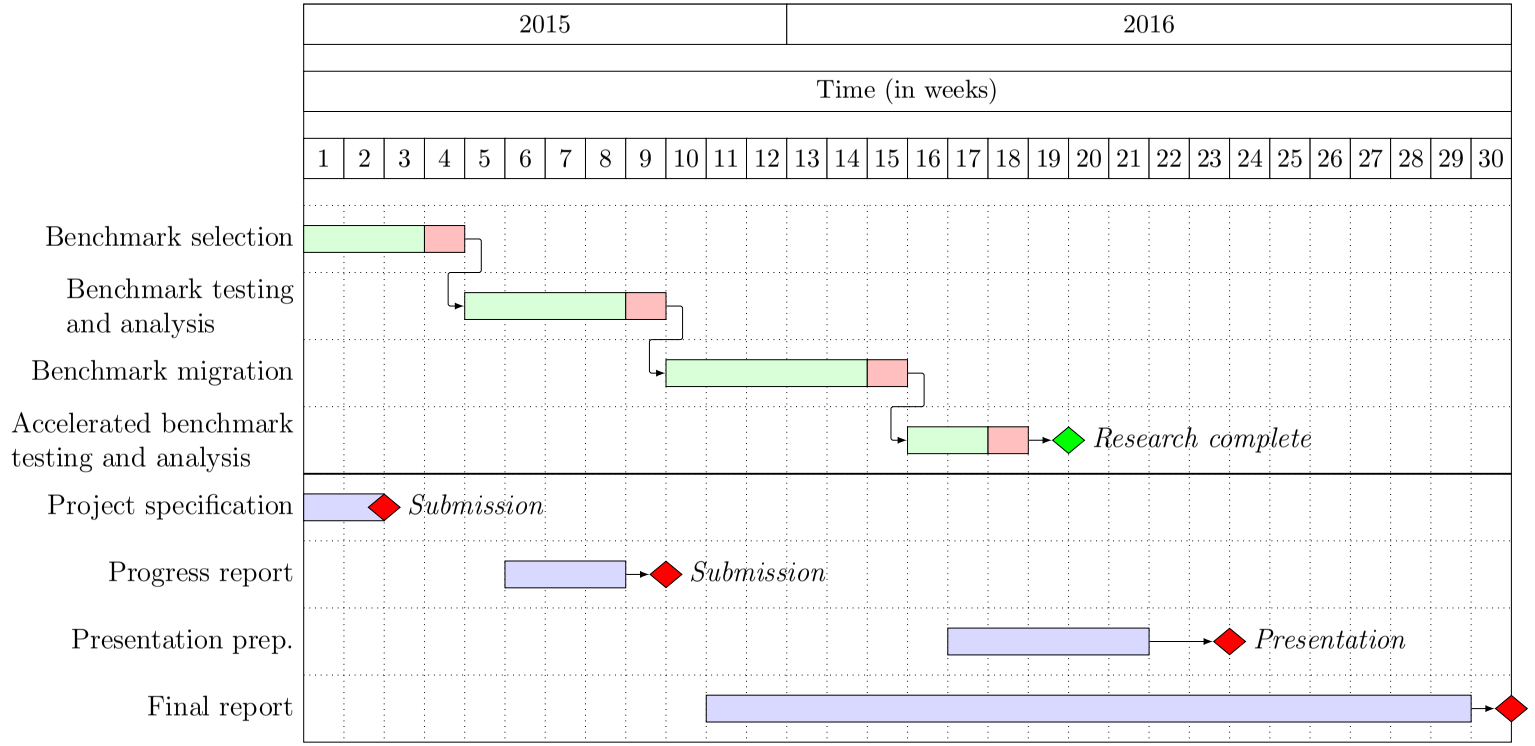

Figure 1: Project timetable from week 1 term 1 to week 1 term 3

Git VCS
: This version control system, along with GitHub, will provide a mechanism of recording previous and current versions of the project in a centralised location.

Nvidia CUDA Toolkit
: The Nvidia CUDA toolkit will be a vital tool for any development of data analytics algorithms for use with Nvidia Tesla GPGPUs. It provides a wide range of language bindings, giving the ability to use accelerators with ease in the Apache data analytics software stack.

IntelliJ IDEA
: This development IDE will prove to be very beneficial during the migration of benchmarks to accelerators. It has many language plugins as well as native test integration, aiding the test driven development methodology previously discussed.

LaTeX
: LaTeX is to be used as the means to produce the documentation for the project. It provides a way to create professional looking documents with easy, whilst also being powerful enough to display any extensive findings without the need for more 3rd party tools.

Along with these tools, the following resources will be crucial to perform the research:

Chiron
: Chiron is an Apache YARN-based data analytics cluster with accessible GPGPU and MIC accelerators. This is useful because it will allow the benchmarking of accelerated and un-accelerated programs on the same cluster.

5.3 Risk Assessment

Figure 2 provides an overview of possible risks throughout the project, as well as their severities and any possible actions that could be taken to mitigate or prevent them from occurring.

Risk	Severity	Likelihood	Mitigating Action(s)
Chiron unavailable	Severe	0.01%	Locate suitable replacement for use in testing — replacement should have similar feature set to Chiron.
Benchmark code unavailable	Severe	5%	Check internet archives for possible location of older version. Find other suitable benchmarks.
Networking failure	Moderate-Severe	5%	Temporarily locate to different area to use a different network.
Project leader falling ill	Moderate	10%	Do work that can be done without further risk to health.

Figure 2: Risk matrix that associates possible risks with severity and mitigating actions

6 Legal, Ethical, Social and Professional Issues

For the project there are a few professional and legal issues that must be addressed. There are, however, no social or ethical issues to be discussed.

6.1 Professional Issues

The project must be completed to a professional standard to ensure it can be extended in further research as well as used commercially. This will involve thorough testing of any new code, along with complete documentation through code commenting. Project documentation should also be of a high standard.

6.2 Legal Issues

The project will be utilising free, open-source software (FOSS) under varying licenses such as the Apache 2 license. Any software development libraries used within the project, such as the Nvidia CUDA toolkit, will also be under their own license agreements. These agreements will need to be adhered to.

7 Conclusion

The underpinning aim of this project is to investigate the use of hardware accelerators within data intensive compute. Acting towards this goal, a set of methodologies, management practices and test plans have been established and adopted to best attain the achievement of the stated goal. Furthermore, the refinement of the defined requirements is to be considered throughout the early stages of the project.

The progress towards the goals of this project will be documented in the next deliverable; the progress review. This document will be completed for and available on the 30th November 2015.

References

[1] Richard T. Kouzes, Gordon A. Anderson, Stephen T. Elbert, et al. "The changing paradigm of data intensive computing". In: *IEEE Computer* (Sept. 2009).

[2] Mario Cannataro, Domenico Talia, and Pradip K. Srimani. "Parallel data intensive computing in scientific and commercial applications". In: *Parallel Computing* (May 2002).

[3] Borko Furht and Armando Escalante. *Handbook of Cloud Computing*. Springer, 2010.

[4] EMC Education Services. *Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data*. John Wiley & Sons, Mar. 2015.

[5] The Apache Software Foundation. *Apache Storm Homepage*. June 2015. URL: `https://storm.apache.org/`.

[6] The Apache Software Foundation. *Apache Spark Homepage*. Oct. 2015. URL: `https://spark.apache.org/`.

[7] Aisling O'Driscoll, Jurate Daugelaite, and Roy D. Sleator. "'Big data', Hadoop and cloud computing in genomics". In: *Journal of Biomedical Informatics* 42 (Oct. 2013).

[8] S. Huang, S. Xiao, and W. Feng. "On the energy efficiency of graphics processing units for scientific computing". In: *IEEE International Symposium on Parallel & Distribued Processing* (May 2009).

[9] F. Bensaali, A. Amira, and A. Bouridane. "Accelerating matrix product on reconfigurable hardware for image processing applications". In: *Circuits, Devices and Systems, IEE Proceedings* 152 (June 2005).

[10] Ari Harju, Topi Siro, Filippo Federici Canova abnd Samuli Hakala, et al. "Computational Physics on Graphics Processing Units". In: (2013).

[11] Jr. Gerald A. Hanwerck. *Accelerating Quantitative Financial Computing with CUDA and GPUs*. Mar. 2013. URL: `http://on-demand.gputechconf.com/gtc/2013/presentations/S3373-Accelerating-Quantitative-Financial-Computing.pdf`.

[12] Warwick Institute for the Science of Cities. "Chiron: Data Intensive Computing for Warwick". Original slides detailing Chiron. 2015.

[13] Chinese Academy of Sciences ICT. *BigDataBench Homepage*. Oct. 2015. URL: `http://prof.ict.ac.cn/BigDataBench/`.

[14] Zijian Ming, Chunjie Luo, Wanling Gao, et al. "BDGS: A Scalable Big Data Generator Suite in Big Data Benchmarking". In: (2013).

[15] Ian Sommerville. *Software Engineering*. Pearson, Aug. 2015.

[16] The Apache Software Foundation. *Apache Hadoop MapReduce Tutorial*. June 2015. URL: http://hadoop.apache.org/docs/current/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html#Overview.