Streaming Data Analytics- Background, Technologies, and Outlook

University of Oulu
Degree Programme on Information Processing Science
Master's Thesis
Adesh Chimariya
August 5, 2018

Abstract

Companies seek significant potential from the datasets they generate within their core processes. However, due to the variety and volatile nature of the data analytic landscapes, in terms of analytic models, programming environment, and software libraries, companies view analytics-related investment too risky. The aim of this dissertation was to gather empirical evidence on the differences in maturity level of the presented data analytical tools.

In this dissertation, we map the state of the art in Big Data and data analytic tools, following a top-down method using semi-structured interviews with companies. These interviews are analyzed with the help of Nvivo^{TM} and the results are compared against the state of the art.

Consequently, recommendations for principles and data analytics tool aimed at reducing the risk of investments, from company's point of view were provided. The maturity level of the technological landscape in streaming data analytics were also highlighted. By analyzing the interview results and reflecting on previous research, among the chosen streaming data analytics tools, Spark Streaming was deemed to have slight upper hand.

Additionally, the result from this dissertation is also intended at guiding the construction of a data analytics infrastructure, aimed at encouraging industry-academia collaboration. This dissertation sheds light on how the companies are not always aware of the state-of-the-art and there is a huge gap between practicalities and research findings. A lot of work has to be done in this area so that the gap does not widen further.

Keywords
Data analytics, Streaming data analytics, Big Data, Streaming engines, Flume, Kafka, Spark, Flink, Storm

Principal Supervisor
Professor Mika Mäntylä

Co-supervisor
Dr. Tech. Marko Jurmu

Foreword

Many individuals have contributed to make this thesis what it is. Some individual have given time and energy to this thesis, whereas some has given inspiration without even realizing it. My Co-Supervisor Mr. Marko Jurmu has played the most instrumental role in guiding me through every step and in ensuring the quality of the research, followed by my Professor Mika Mäntylä, whose guidance helped me to maintain the course in this research. My fellow researcher Mr. Prabhat Ram was kind enough to provide me with some help with many things along the line.

However, some of my colleagues and friends who directly or indirectly contributed to this research throughout the time will always be critical to this research. They helped me move on and gave me inspiration when needed. Mr. Raghu KC, Ms. Noémie Piquerez, and Mr.Puya Purbaba were always there to help me and were very instrumental throughout the whole process. My family deserves a special note of thanks as their wise council has always helped me move forward. Therefore, I collectively thank aforementioned people again for their contributions.

I hope you enjoy reading it as much as I enjoyed writing it.

Oulu, 07.06.2018

Adesh Chimariya

Contents

1. Introduction

1.1. Background

The emergence of social media, Internet of Things (IoT), multimedia, etc., has produced an overwhelming amount of structured and unstructured data. The amount of data created is also widely recognized by the term Big Data. It is characterized by three aspects (a) data are numerous, (b) data cannot be categorized into regular relational databases, and (c) data are generated, captured, and processed rapidly (Hashem et al., 2015). Big data has a staggering impact on healthcare, science, engineering, finance, business, and even the society itself.

Now, the analytic tools used in Big Data is far more powerful than any of the analytics tools used in the past. Now we can measure the data and therefore manage it more precisely than before. This, in turn, helps us make better predictions and smarter decisions. Big data is also becoming a major business asset, as the value of data is rising. A good example is a case where Facebook sold its data to Cambridge Analytica. The 2016 Presidential election in The United States of America was Big Data vs. Big Data, where one candidate's campaign took the data-driven psychometric micro-targeting strategy to a military grade (Albright, 2016). The result of the election was not the fault of Facebook, which collected a tremendous amount of user data, but it was how the data was used in finding people who can be influenced enough to go and vote. The two major factors here are data being sold, and data being used to influence behavior. Facebook was the source of psychological understandings that enabled Cambridge Analytica to target individuals and facilitated them to be conveyed on a large scale (Cadwalladr, 2017). McAfee et al., also claim how all the decisions, erstwhile, made by executives in many different areas relying on intuition could now be made by data and rigor. They conducted an experiment with the hypothesis that data-driven companies were better performers. Structured interviews with 330 public North American companies to enquire about their organizational and technology management practices, in addition to performance data from their annual reports and independent sources were collected. From this experiment, it became apparent that the companies claiming to be data-driven were better performers on financial and operational fronts. Some of these top companies were, on an average, 5% more productive and 6% more profitable than their competitors (McAfee, Brynjolfsson, Davenport, Patil, & Barton, 2012).

One example of data driven-model is the amount of time and money it saved for PASSUR Aerospace (McAfee et al., 2012). Earlier, airlines had to depend on the estimated time of arrivals (ETAs) provided by the pilot. A major US Airline learned from an internal study that 10% of the flights into its major hub had at least a 10-minute gap, and 30% had at least a 5-minute gap between ETAs given by the pilot and the actual arrival time. This gap meant a large number of ground staffs had to sit idly, which in the bigger picture meant loss of tremendous amount of time and money. PASSUR started its service named RightETA that calculated the time of arrival of planes by analyzing big data, and started providing its own arrival

estimates to airlines, saving millions for the companies in return. This proved, yet again, that using big data leads to better predictions and better decisions.

With the amount of sensor network, social media presence, and smartphones and devices, the amount of data being generated in real-time is immense. Twitter processes approximately 10,000 tweets per second before publishing them for the public (Shoro & Soomro, 2015). Before publishing, the company needs to analyze them and ensure the tweets follow their protocol of decency, and any vulgar or restricted words are filtered out. Streaming analytics system avoids putting data at rest, and processes it as soon as it is available to the system. This, in turn, significantly minimizes the time a single datum spends in a processing pipeline (Wingerath, Gessert, Friedrich, & Ritter, 2016). Due to massive increase in complexity of the data due to rapid growth in technology has today's world face many new challenges in the life of data management, a significant amount of such data that is produced is very volatile, which means it needs to be analyzed in real time as it arrives(Krempl et al., 2014; Shoro & Soomro, 2015). Big data analysis is a process of gathering data from different sources, having them organized in a meaningful way, and then analyzing these sets of data to uncover underlying facts and figures from that data collection.

In this era of Big Data, it is necessary to analyze large amount of data in small amount of time (H. Zhang, Chen, Ooi, Tan, & Zhang, 2015). Stream processing solutions have to solve many problems such as processing massive amount of streaming events, it should be real-time responsive to changing market conditions, it should increase its performance, and should be scalable with the ever-changing volume of data complexity. Furthermore, it should be good in analytics, such as live data discovery and monitoring, continuous query processing, and automated alerts and reactions (Namiot, 2015). Contrary to traditional offline data processing (Krempl et al., 2014), stream data analytics require fully automated data preprocessing method that can optimize the parameters and operate autonomously. Furthermore, the preprocessing model needs to update itself to evolve with the evolving data. Fault tolerance becomes of paramount importance, especially for continuously running computation, as new data is flowing through the system non-stop (Qian et al., 2013). Streaming data has been a topic of extensive research in the database community. Most of the data that a company receives are real time, and are most valuable at the time it arrives (Córdova, 2015). For instance, there is no point detecting a potential buyer after the user has left the e-commerce site or detecting fraud after the perpetrator has fled.

The type of data may differ depending on the application scenario, including discrete signals, monitoring information, event logs, video, etc. It is very vital to tell apart streaming data when it arrives at the processing system via, for instance, a log or queueing system, and intermediate streams of tuples resulting from the processing by system elements (Dias de Assunção, da Silva Veith, & Buyya, 2018). The latency gap between multi-core CPUs and mechanical hard disks are rising every year, and the computer architectures are increasingly imbalanced (Hey, Tansley, & Tolle, 2009). This, in turn, is making the data-intensive computing tougher to overcome. A very famous American computer scientist, Jim Gray postulated several unofficial rules or laws on how to approach data engineering challenges that are related to

large-scale data sets. The solution for this growing problem is “scale-out” architecture and the computations should be brought to the data rather than the other way around (Hey et al., 2009). This principle has led to interests in academic and industry-scale tools and software.

1.2. Objectives

The aim of this dissertation is to gather empirical evidence that provides the differences and the maturity level of the presented technology on data analytical tools that aim to lessen the risk of investment from the company’s point of view. This dissertation aims to provide recommendations for the companies to address their existing challenges by comparing our survey result with the problems faced by the companies and it will be a stepping-stone from an academic perspective for further research in this field. It also intends to map the current state-of-the-art in streaming data analytics. The research on current technology landscape and industry uptake of streaming data analytics are pursued. In addition, the focus is on mapping industry uptake on streaming analytics and their familiarity with it and preferences on analytics tools. This thesis also focuses on understanding the current tools that companies are using to tackle with their growing need for data analytics and management, their satisfaction or dissatisfaction with the current tools. Big Data/data analytics methods are seen as a key way for immaterial value creation in various domains, but the technology landscape is highly dynamic with varying levels of maturity. Hence, to understand the technology landscape better, discuss the models that seem to have significant benefit, and point out the flaws, if any, is one of the major factors for this thesis. We also try to understand and identify the current important principles of different paradigms of analytics and different data pipeline methods.

1.3. Research questions

Stream analytics is a relatively new topic for both companies and in itself, and requires sufficient assessment. Keeping this in mind, the research questions were broad and flexible in order to collect as much information from the people involved in this research. The broad questions, combined with a qualitative research method, can help formulate an overview that could pave the way for further research purposes. Thus, following research questions were formulated to assist us in our enquiry:

- RQ1: What are the reported types of data the ICT companies deal with on day-to-day basis and how familiar are they with streaming analytics?
- RQ2: With a number of streaming analytics tools on the market, how have you considered a suitable analytical tool to serve your company needs?
- RQ3: To what degree is the company’s decision-making driven by data analytics?

1.4. Method

The methodology used in this thesis is twofold: First, streaming analytics technology landscape was surveyed. The most recent and relevant solutions and tools were selected for comparison. Second, a select number of companies working in the domain of Big Data and/or data analytics were interviewed. Interview data were compared with our candidate technologies from the survey to deduce some conclusion. Semi-structured interviews were used to assess company awareness and readiness in streaming data analytics. The interviews were conducted to seek better understanding and/or create a hypothesis. Such an approach requires qualitative research where the interviewee shares rich description of phenomena, while leaving the interpretation to the investigator (DiCicco-Bloom & Crabtree, 2006). We then compare interview data with our candidate technologies to derive conclusion for the maturity of tools and on the industry uptake. This field of research is wide-ranging, and it requires an equally wide-ranging exploration. This research will also help us understand how companies these days make their decisions. Therefore, a semi-structured interview helps keep the questions specific and at the same time, there is plenty of room for the interviewee to elaborate if they want. Seven individuals representing seven companies were interviewed, either in person or over Phone/SKYPE. Factors such as age, gender, origin, etc. were not taken into consideration, as the companies that were selected . The interviews were recorded, transcribed, and analyzed. The companies we interviewed deal with significant amount of datasets that are generated within their core process, or while working with their customers. Due to variety and volatility of analytic models, it is not in their best interest to switch around these tools to figure out the best tools for them, for various reasons, risk of losing the data or corrupting it or due to financial loss.

1.5. Structure

This research is structured as follows: Chapter 1 covers the introduction part of the research, objective of the dissertation, and research questions also is included here. Chapter 2 mainly focuses on the background, delves deeper into the subject related field on streaming analytics, such as, key architecture consideration of streaming data, and the streaming engines that are taken into consideration in this dissertation. Chapter 3 covers the research methods and the approach taken to conduct the research. Chapter 4 covers the results and findings of the dissertation. Chapter 5 provides discussion of the research findings, limitations, threat to its validity, and suggests some potential future work. Chapter 6 provides an overall conclusion to this dissertation.

2. Streaming Data Analytics

2.1. Background and motivation

Traditional data analytics systems collect huge static volumes of data and periodically process those data. On the other hand, streaming analytics systems avoid putting data at rest, and process it as it is coming to the system or as it becomes available, cutting the data analytics time significantly and minimizing the time a single data spends in a processing pipeline (Wingerath et al., 2016).
In recent years, batch processing was thriving with offline data processing. Both Hadoop and Batch processing have progressed with time to become excellent offline data processing platforms for Big Data. Recently, things have changed drastically, as many use cases across various domains started calling for near-real-time/ real-time response on Big Data for faster decision making in their respective field. Hadoop was not suitable enough for those use cases. The rate at which organizations were analyzing their data was slow, and the data was losing its value exponentially over time (Bhattacharya & Mitra, 2013).

Many use cases require continuous and speedy analysis, such as the Government, Healthcare, Finance, Automotive, Telco, Insurance, Energy and Utilities, etc. (IBM Analytics, 2016).

- In governance, streaming analytics performs continuous and speedy analysis to protect against threats and reduce fraud, by having smarter surveillance, identifying fraud and terroristic activities, discovering cyber-attacks and preventing it, being aware of street crime.
- It anticipates diseases onset and deliver patient data to make life saving judgements in healthcare, the care patients get shall be highly personalized and the treatment will be pre-emptive.
- It lowers the risk of cost and fraud at the same time supporting informed transactions and better revenue in financial sector.
- In automotive industry, it boosts operations, expand the driving experience and construct safer roads. This results in more interactive and safer driving experience, the integrated vehicle data, and improved quality and functionality.
- In regards to telecommunication sector, it increases customer satisfaction by processing all the call data, and timely marketing promotions. For the companies, it maximizes asset utilization and proactively preserves lucrative customers.
- The insurance industry has a lot to gain from streaming analytics as it increases service to the clients and servers by decreasing cost and fraud. It helps in protecting the cargo, and speed up fraud detection; It also helps in optimizing call centers.
- Streaming analytics optimizes energy usage and reduces power outages by predicting and detecting outages, load shedding and performing smarter analytics.

Familiarizing with big data is through the concept of 4V's: Volume, Velocity, Variety, and Veracity (Kune, Konugurthi, Agarwal, Chillarige, & Buyya, 2016). The way Real time data processing or Streaming data analytics has to handle all of the V's. Out of these, handling the velocity of the data is not an easy task. First, the system gathers real time data coming into the system at about a rate of millions of events per second. Secondly, it needs to deal with the parallel processing of the said data while the data is collected. Third, it should carry out event correlation using a Complex Event Processing engine to attain substantial information from the flowing stream. All these three steps should happen in fault tolerant and distributed way, with very low latency so that the computation can happen very promptly with close to real-time response capabilities (Bhattacharya & Mitra, 2013).

To keep up with the massive flow of data every minute, the tools must analyze the resulting interaction networks and graphs as the data arrives in high-volume, rather than analyzing in static snapshots. The streaming data flooded in social media and other applications are just too large to permit constructing streaming analysis from static snapshots (Ediger, Jiang, Riedy, & Bader, 2010). The omnipresence of smartphones and other wearable smart devices help in generating massive amount of data and if processed instantly, can lead to valuable information. In some cases, (Perera & Suhothayan, 2015) value of insights degrades very quickly with time, rendering the data almost useless. Stock market is a very good example of such case. Hence streaming data analytics can be very feasible when it comes to taking advantage of the data in hand instantly. Despite numerous studies on the streaming analytics as an individual technology, there seems to be a significant gap in understanding the different paradigms of analytics and different data pipelines methods. This has been the major motivation for this thesis, and also to understand and identify the current and important principles of data analytics.

2.2. Challenges of stream processing systems

Stream processing systems encounters various challenges that could be broadly classified into four categories (M. P. Singh, Hoque, & Tarkoma, 2016b):

Data Acquisition - It is a significantly challenging task to handle an enormous stream of continuous data. The sheer velocity of the incoming data can be overwhelming for the system to adapt to. The veracity of data that come to the system are Structured and Unstructured data. The structured data acts as an ideal input for stream processing system, whereas unstructured data should go through data pre-processing that involves filtering, extraction, and organization into structured format of data. The format of data plays a vital role in the latency of stream processing system. The accurate depiction of data and data acquisition strategies depends on the application built on top of the stream processing system.

Data Handling - Properly handling large volume of data is another challenge. The stream processing system needs to analyze the sensitivity of the data, and store it in long-lasting storage. The data handling and long lasting storage of data format varies

with the application requirement, and this needs to be evaluated very well by stream processing systems.

Data Modeling - Stream processing systems require in-memory processing capabilities to have low latency. Considering the 4Vs (velocity, veracity, volume, and variety) of data, the stream processing systems require predictive models and effective algorithms to extract applications linked to vital events from enormous data streams. It also needs data models to achieve complete analysis by combining all available data.

Data mining - Stream computing encompasses computational analysis and analytics around it. Stream processing needs new computational tools that can evaluate diverse data sets into suitable results. It entails data visualization and data analytics of huge data sets. The old-fashioned mining methodologies need to modify itself as per in-stream processing to give dynamic results.

2.3. Key Architecture Considerations

Here we briefly discuss few key architectural considerations of Streaming data analytics.

2.3.1. Master-Slave Vs Peer-to-Peer

A Master-Slave model, or also known as primary/replica model, is one of the simplest models of communication where one device has unidirectional control over one or more devices. Here, the device taking control is the Master and the devices that are under control are slaves (D. Singh & Reddy, 2015). A drawback of this architecture is that it is asymmetrical, with possible downtime or loss of data when master fails. This paradigm is fundamental and commonly used approach for parallel and distributed applications, and well matched as a programming model for applications targeted to distributed heterogeneous ‘grid’ resources (Shao, Berman, & Wolski, 2000). Master-slave tends to be of star topology, where each hub is connected directly to a central hub, creating a shape of a star. As long as the hub is active, network is not affected by node failure, which is one of the main advantages of star topology (Nair et al., 2015). In this topology, only the sink node, also known as central hub, is aware of the status of all the nodes as it centralizes control. One master can be connected with many slaves, whereas one slave can only be connected with one master (Nair et al., 2015).

Peer-to-Peer (P2P) architecture is about establishing communication networks/systems based on peer-to-peer models or resource sharing. The participants of such network/system are both resource providers and resource requesters (Schollmeier, 2001). As central dependency is excluded, failure of one peer in P2P techniques does not affect the functionality of other peers, which makes it more reliable (Navimipour & Milani, 2015). Some of the downsides of peer-to-peer architecture are no central system to manage and control the access of data, data recovery and back-up is very tedious, as each system should have its own back-up system, low security, and there is no guarentee about QoS (Navimipour & Milani,

2015; Schollmeier, 2001; D. Singh & Reddy, 2015). Peer-to-peer is one of the oldest, decentralized, and distributed computing platforms in existence, where the nodes, also known as peers, serve as well as consume resources.
With growing network size, synchronization overhead grows with it. In case of unstructured P2P networks, there are severe problems such as overhead of query forwarding and of storage and update of caches. Super-Peer, a hierarchical approach based on leveraging the heterogeneity of the participating peers, was introduced to address this issue (Kurve, Griffin, Miller, & Kesidis, 2015). Super-peer-based P2P architecture has at least two hierarchical levels, viz. the super peers and the ordinary peers. The super-peers node assumes higher responsibility, as they are the nodes with resource capabilities such as memory, computation, and bandwidth, which are much higher than ordinary peers (Kurve et al., 2015). Every peer is assigned a super-peer, and they route their queries via the super-peer they are assigned to. The super-peers are likely to become hubs of computation and communication, as they are directly included in content distribution protocol for all the peers that are reliant on them (Kurve et al., 2015).

To communicate and exchange data between peers, Message Passing Interface (MPI) is used as communication scheme. Hierarchical master/slave paradigm is another feature of MPI. The slave machine becomes the master of other processes when the MPI is deployed in one of the master-slave model (D. Singh & Reddy, 2015).

2.3.2. Execution semantics

Stream Processing Engines (SPEs) implement various processing models and standardized executions semantics have not yet created, this impedes usability and interoperability of SPEs. SECRET (ScopE, Content, REport, and Tick)(Dindar, Tatbul, Miller, Haas, & Botan, 2013), a model for analysis of the Execution Semantics (Affetti, Tommasini, Margara, Cugola, & Della Valle, 2017) of SPEs, could effectively capture and confront the behavior of all the systems it was applied to. In case of a failure, different SPEs offer different guarantees. Most of the SPEs offer exactly-once semantics. This implies no loss or duplicate processing are possible, and, therefore, failure doesn't affect the execution semantics. On the other hand, some platforms such as Storm offer at least once semantics, in which duplicating of some processing is allowed (Affetti et al., 2017).
There are three ways to analyze the execution semantics of individual SPEs, and compare the execution behavior of heterogeneous SPEs (Botan et al., 2010; Dindar et al., 2013):

Syntax heterogeneity indicates the differences in language clauses (keywords) used to define common constructs, since there is no standard language for stream processing.

Capability heterogeneity indicates the difference in SPEs, as they differ in their support of different query types, and exposes itself at the language syntax level.

Execution model heterogeneity refers to the dissimilarities in the underlying query execution models across SPEs. It is below the language level, hidden, and cannot be

influenced by the application developers. It is subtler than other heterogeneity and more confusing.

Therefore, it is of vital importance that companies getting started on SPEs understand the feature and semantics of it, as applications are not portable and can be a difficult task to build, even on a given SPE (Dindar et al., 2013).

2.3.3. Edge computing

More and more data is produced at the edge of a network, and so it only makes sense and would be more efficient to process the data at the edge of the network (Hu, Patel, Sabella, Sprecher, & Young, 2015). Edge computing (Satyanarayanan, 2017) is a new paradigm where substantial computing and storage resources, also denoted as cloudlets, micro datacenters, or fog nodes, are placed at the edge of the Internet in close proximity to the mobile device, sensors, or data source. The proximity of cloudlets helps in various ways:

Low-Latency Cloudlet Offloading - The physical proximity of a cloudlet to a mobile device makes it easier to achieve highly responsive, low end-to-end latency, and low jitters to services located on the cloudlet. This benefits massively to the applications such as augmented reality and virtual reality that are both computational intensive and latency-sensitive.

Scalability through Edge Analytics - Any application that transmits video feed from their smart device to cloud for content analysis will benefit massively from this. The cloudlets run any type of analytics near real time, and send the result along with metadata to the cloud, reducing the entry bandwidth to the cloud up to six orders of magnitude.

Enforce Privacy at the Edge Analytics/Filtering/Abstracting - There are many concerns arising from Internet of Things (IoT) system over centralization. Cloudlet can solve this problem, as it would be the first point of infrastructure contact for the sensor streams. Cloudlets can achieve scalable and secure privacy solution that fits well with the organizational boundaries of trust and responsibility.

Mask Core Cloud Outages - Our dependence on cloud grows every day. This also means the vulnerability to cloud outages grows. If or when the network is under cyber-attack, destroyed by natural disaster, or a developing country with weak-networking infrastructure, Cloudlet can act as a fallback service temporarily masking the cloud inaccessibility. It can serve as a proxy for cloud and carry out its important services.

Although with medium to low computational capacity compared with cloud computing, edge computing has unique advantages over cloud computing. Edge servers are relatively smaller, placed over multiple locations, whereas in cloud computing, the servers are considerably large and centralized. Edge computing is suitable for applications demanding low latency, real-time operations, and high quality of service (Sharma & Wang, 2017). As ultra-low latency is one of the

proposed characteristics for the future 5G network, edge computing is a natural partner of 5G network, as it guarantees ultra-low latency (Satyanarayanan, 2017). Therefore, introduction of edge computing in the conventional and centralized cloud-computing setup gives us the new opportunity to balance trade-off between centralized and distributed network architecture (Sharma & Wang, 2017). One of the examples where edge computing thrives is smart transportation, where a traffic alert system can alert the vehicle heading towards the incident (Cheng, Papageorgiou, & Bauer, 2016).

2.3.4. Serverless architecture

Serverless architecture represents a unique approach to designing applications on the cloud, not having to worry about managing servers. The name "serverless architecture" is confusing as it most certainly has a server, but entirely as a third party service (Crane & Lin, 2017). Here, the developer specifies functions with well-defined entry and exit points, and all the other phases of execution is handled by the cloud provider. It is serverless from the developer's perspective (Crane & Lin, 2017). As shown in Figure 1, the core capability of a serverless platform is that of an event processing system. The service must manage a set of user defined functions, receive an event from an event source or take an event that was sent from HTTP, determine the function(s) to which to forward the event, find an existing instance of the function or create a new instance, send the event to the function instance, wait for a response, gather execution logs, make the response available to the user, and stop the function when it is no longer needed (Baldini et al., 2017). The challenge in the architecture is to implement such functionality while keeping metrics such as cost, scalability, and fault tolerance into consideration.

As serverless architecture is still in its infancy, a well-known example of a commercial implementation is AWS Lambda. There is OpenWhisk that is open-source platform. Other competitions of AWS Lambda are Azure functions, Google cloud functions, but AWS Lambda is more mature and more efficient than the rest (Bila, Dettori, Kanso, Watanabe, & Youssef, 2017).

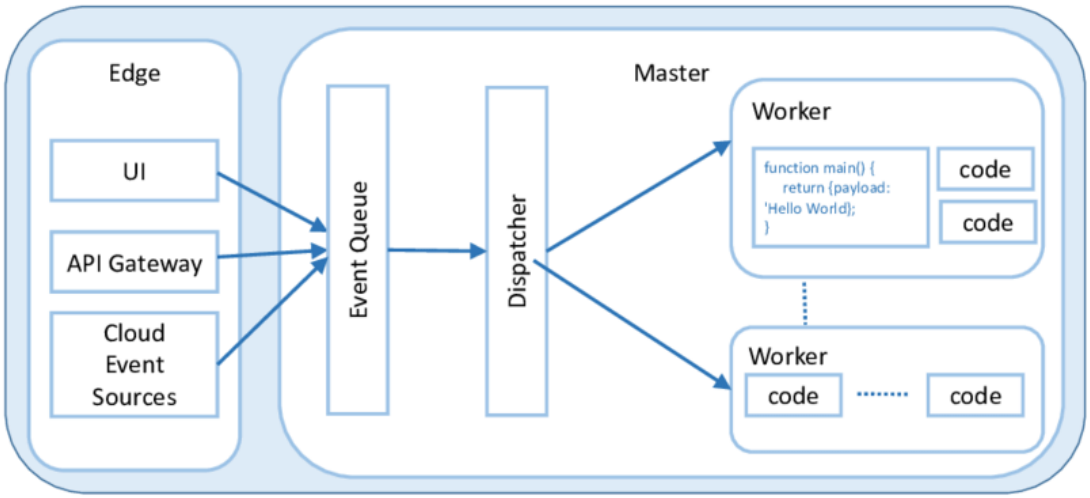

Figure 1. Serverless Platform Architecture (Baldini et al., 2017)

Traditionally, application developers had to spend significant development effort in latency, scalability, and elasticity. Now, they can focus on cost of their code when modularizing their applications (Baldini et al., 2017). There are significant advantages for both consumers and providers. From consumers' point of view, they no longer need to worry about provision and manage servers, VMs, etc. Instead, they focus solely on their functions on the applications that enable desired application behavior. This gives the provider more control over software stack allowing them to transparently deliver security patches and optimize the platform (Baldini et al., 2017).

2.4. Streaming engines

Needs for streaming analytics has given rise to numerous implementation initiatives. The concept of stream processing has evolved to real time analytics on massive stream of data (M. P. Singh, Hoque, & Tarkoma, 2016a). These represent differing maturity levels and focus areas, so it is important to survey the landscapes. Here, five candidate technologies are presented and discussed.

2.4.1. Apache Flume

Apache Flume is distributed, reliable, and available service for efficiently collecting, aggregating, and moving massive amounts of streaming data from different applications/webservers into Hadoop Distributed File System (HDFS)(Apache flume.June, 2015). Flume allows streaming data from multiple sources to be ingested into HDFS for storage and analysis. It also performs other features such as buffering storage platform. When the rate of incoming data exceeds the rate at which the data can be written into destination, it guarantees data delivery, and it can scale horizontally to ingest new data streams and additional volumes as needed (Apache flume.2017). Flume ensures guaranteed delivery of data as both sender and the receiver invoke the transaction.

A Flume agent's architecture can be illustrated in a simple diagram, as shown in Figure 2. An input is known as source and an output is known as a sink. A channel caters the glue between a source and a sink. This mechanism runs inside a daemon called agent (Hoffman, 2015). A basic payload of data transported by Flume is identified as an event. A source writes events to one or more channels. A Flume source ingests events provided to it by an external source, such as a web server, and then the source stores the event in one or multiple channels. The channel is like a passive store that stores the event until consumed by Flume sink. The sink removes the event from the channel and drops it into an external repository like HDFS, or sends it to next Flume source of the new Flume agent. A source writes events to one or more channels.

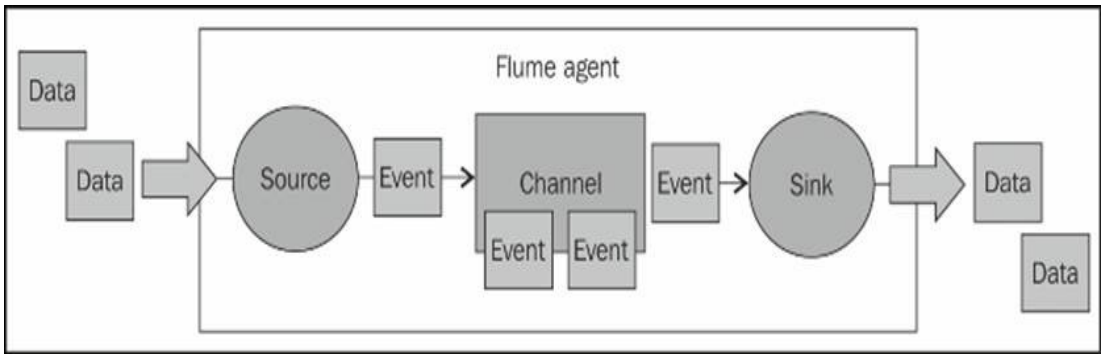

Figure 2. Apache Flume architecture (Hoffman, 2015)

The major domain areas for Apache Flume would be many enterprises (EDUBCA, 2018) such as e-commerce, online retail portals, social media, etc., to cope with high volume stream of data to reach in HDFS, as the typical source of the data here is log data, sensor and machine data, geo-location data, and social media (Apache flume.2017).

2.4.2. Apache Storm

Storm is a free and open source distributed real time computational system that does analysis on streams of data as they come in. The current version 1.0.0 has been developed since late 2010, and was open-sourced in September 2011 by Twitter, and then became a top-level project in Apache later by 2014 (Wingerath et al., 2016). It also simplifies working with queries and workers. It is scalable and fault-tolerant, and is benchmarked for processing millions byte data per second per node (Shoro & Soomro, 2015). It is compatible with almost any programming language, or it provides an adapter to write applications in any language, and it is written in Java and Clojure (M. P. Singh et al., 2016a). It defines workflows in Directed Acyclic Graphs (DAG's) called topologies. It uses ZeroMQ to pass the messages, which guarantees its architecture to pass the message at least once, and if a node does not process a record, it replays the records. If there is a failure of a task, the messages are reassigned by restarting the processing.

Storm topology is the overall calculation, represented visually as network of spouts and bolts. As shown in figure 3, spouts are the sources of data streams (an unbounded sequence of tuples) in a computation, and bolts are there to listen to the data, accept tuple, perform computation or transformation, such as filter, aggregate, or join data, query databases (Dias de Assunção et al., 2018).

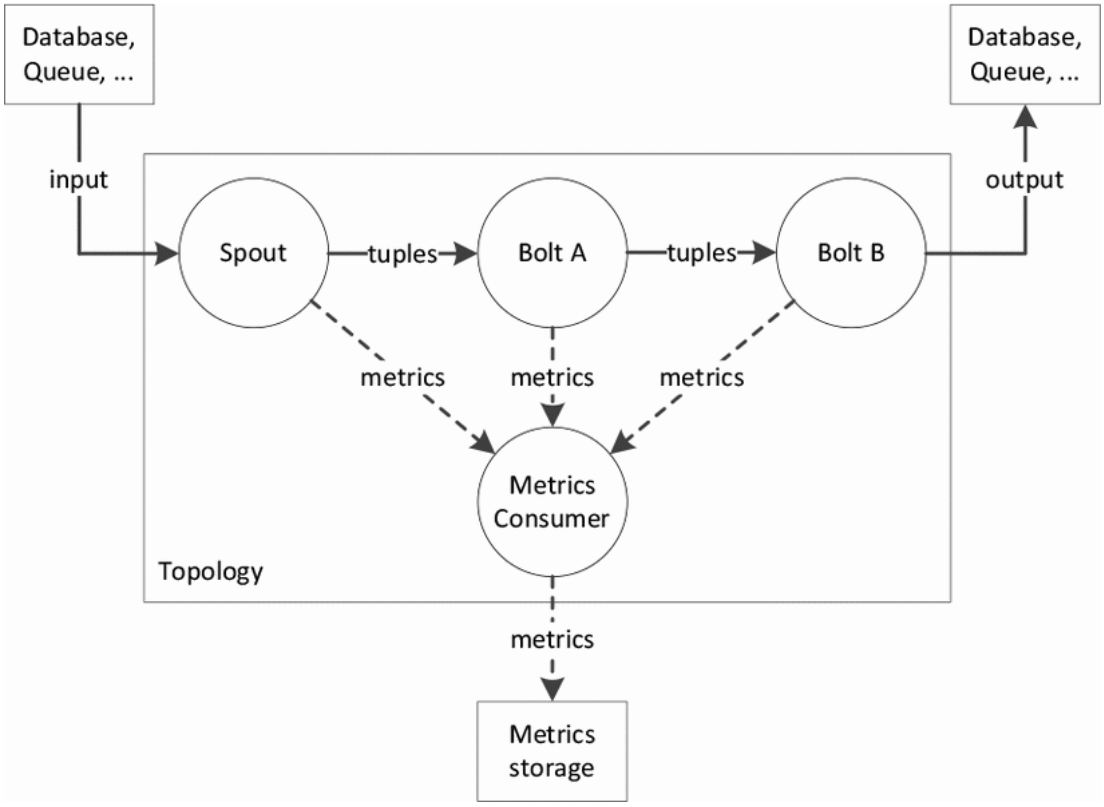

Figure 3. Storm topology (van der Veen, Jan Sipke, van der Waaij, Lazovik, Wijbrandi, & Meijer, 2015)

There are two different types of nodes in Strom cluster, the master node and the processing node. The master node runs a daemon called Nimbus that is responsible for distributing code around the cluster, assigning tasks, and monitoring their progression and failures. The master node also runs a daemon called UI that provides a website for users to view the status of the cluster and manage topologies (van der Veen, Jan Sipke et al., 2015). The processing node continue to process data in case there is a failure in master node, but reconfiguration of the cluster is not possible until the master node is restarted (van der Veen, Jan Sipke et al., 2015).

Storm (Apache storm.2018) is extremely fast and the ability to process over million records per second per node on a cluster of decent size makes it very useful to many domain areas. Some of the major domain areas that storm specializes are cyber security analytics and threat detection, customer service management, data monetization, etc.

2.4.3. Spark Streaming

Spark streaming is an extension of Spark that is an in-memory distributed data analysis platform. The spark framework is also often referred as unofficial successor of Hadoop (Zaharia, Chowdhury, Franklin, Shenker, & Stoica, 2010), with better and more concise API resulting in less extravagant application logic and significant performance improvements through in-memory caching. Since data is not necessarily

written to and loaded from disk between every processing step. It also provides a variation of machine learning algorithms out-of-the-box through the UC Berkley AMP Lab, which was based on 2007 Microsoft Dyrad paper. It is written in Scala, but also supports Java, Python, SQL, and R. It is one of the largest OSS communities in big data, with over 200 contributors in 50+ organizations. It was open sourced in 2010, and is a part of Apache software foundation since 2014. Spark's ability to call into existing Java libraries and to support Hadoop-supported storage system makes it a pragmatic choice to complement Hadoop for large-scale data analysis (Zaharia et al., 2012).

The core abstraction of spark is distributed and immutable collections called resilient distributed datasets (RDDs) that can only be manipulated through deterministic operations (Wingerath et al., 2016). Spark keeps track of its RDDs lineage, making it resilient to machine failures. Spark is very effective at iterative computations and it very compatible for the development of large-scale machine learning applications (Meng et al., 2016). Spark treats streaming computations as series of deterministic batch computations processed by spark's core on a minor time interval (Córdova, 2015).

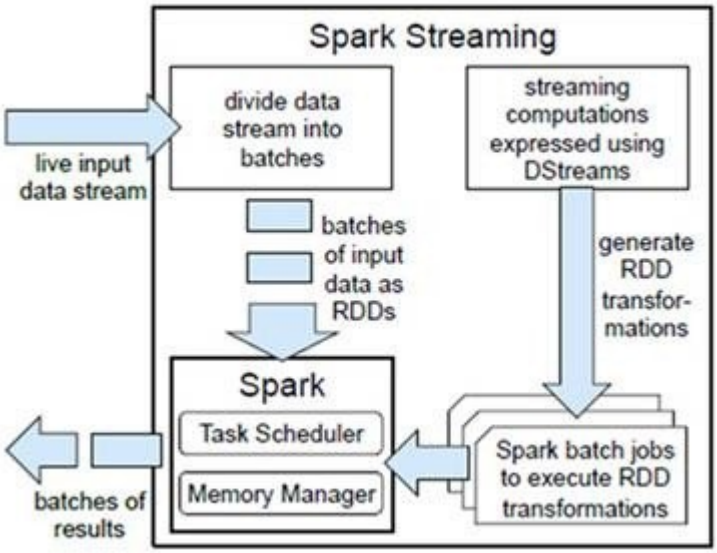

Figure 4. High-level overview of Spark Streaming system (Zaharia, 2016)

As shown in Figure 4., Spark streaming receives live input data streams, the data is divided into batches, and then stores them in Spark's memory as RDDs. Spark engine then processes the batches of input data to generate the final streams of results in batches (Spark streaming programming guide.2018; Zaharia, 2016)..

2.4.4. Apache Flink

Apache Flink is an open source framework for distributed stream processing that delivers accurate result, even if the data is out of order or has arrived late. In other words, it is an open source system for processing streaming and batch data. The

philosophy behind analytics, batch processing, machine learning, graph analysis, etc., can be stated and achieved as pipelined fault- tolerant data flows (Carbone et al., 2015). Memory-based processing has been dominated by Spark and Storm for few years, as both of them are extremely versatile and useful products. Apache Flink, with some of its features, offers some advantage over these tested and proven products (Lakhe & Lakhe, 2016). Flink guarantees exactly-once semantics for stateful computations. In this context, stateful means the kind of applications that can maintain an aggregation of data that has been processed over time. Here, Flink's checkpointing tool safeguards exactly-once semantics for an application's state in the event of failure. Flink is designed to be both DataStream API and DataSet API for stream analytics and batch analytics. Flink is primarily a stream-processing framework that can also perform batch processing like Spark streaming. Flink has an excellent optimization engine that analyzes input code (to the cluster) and decides on the best pipeline (it considers fit) to execute that code for a specific setup (Lakhe & Lakhe, 2016).

Flink is only a framework for distributed data analysis. The core of Flink has streaming iterative data flow engine. It uses two major APIs, the DataSet API and DataStream API. DataSet API is used for processing batch data, and DataStream API is for processing event streams. These two APIs are on the top of the core engine to provide versatile functionality of processing different data with equal ease (Lakhe & Lakhe, 2016).

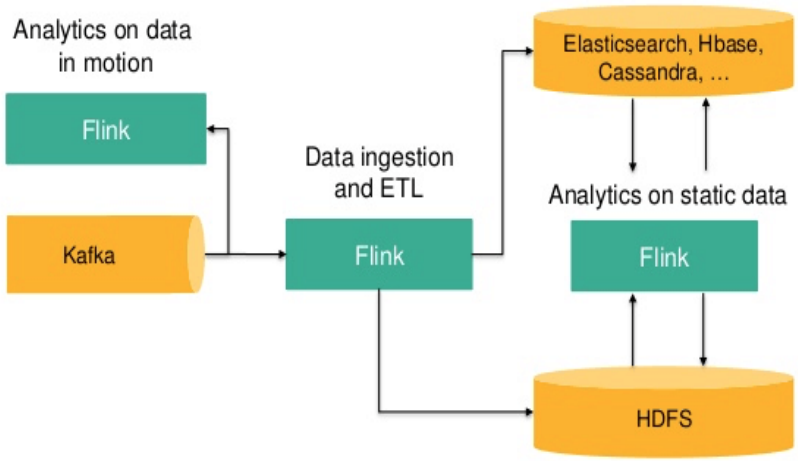

Figure 5. Apache Flink (Janakiram, 2016)

Since Flink does not have its data storage system, the input data can come from various sources. It consumes data from Kafka for stream data processing, or from HDFS or HBase for batch processing (Janakiram, 2016).

2.4.5. Kafka Streams

Kafka streams, compared with other streaming engines, are relatively new. It is fast, scalable, fault tolerant, and offers high throughput. Mainly, it is used for log processing. It provides an API similar to messaging system, and allows applications to consume log events in real time (Kreps, Narkhede, & Rao, 2011). It is a client library to build applications and micro services. Here, the input and output of the data are stored in Kafka clusters. It is elastic, highly scalable, and fault-tolerant system, which is feasible for small, medium, or large use case, and only processes semantics once. It is fully integrated with Kafka security and does not require a separate processing cluster. It has very different hardware profile than others, and is not widely supported or used. The few recognized companies/organizations that use Kafka streams (Apache kafka a distributed streaming platform.2018) are The New York Times, Pintrest, Zalando, Trivago, Rabobank, etc. As shown in Figure 6., the producer API allows an application to publish a stream of records to one or more topics (streams of records in categories). The Consumer API permits an application to subscribe to one or more topics, and process the stream of records that are produced to them (Apache kafka a distributed streaming platform.2018).

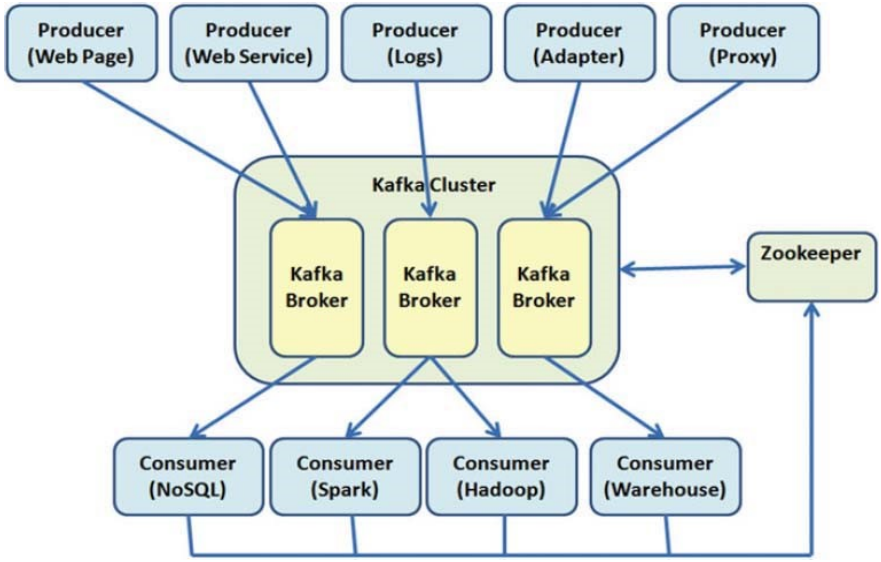

Figure 6. Apache Kafka typical scenario (Estrada & Ruiz, 2016)

Apache Kafka quickly routes real-time information to consumers, and is a message broker that provides seamless integration. The two collateral objective of it is to not block producers and to not let the producers know who the final consumers are (Estrada & Ruiz, 2016). The few examples of Apache Kafka use cases are, it commits logs, does log analysis, stream processing, record user activity, etc. Well-reputed companies such as LinkedIn, Uber, Twitter, Netflix, Spotify, and Yahoo are using Apache Kafka for various data analytics processes (Estrada & Ruiz, 2016).

2.5. Summary and comparison of chosen technologies

The five technology chosen here in this thesis are by no means the only ones in Streaming analytics field. The sharp increase in Big Data and data analytics has led to various technology landscapes with varying level of maturity. Real-time processing overcomes all the drawbacks and limitations of batch processing (Bhattacharya & Mitra, 2013). Among the aforementioned technological landscapes of streaming data analytics, Kafka and Flume are data ingestion tools and play a very important role in data analytics. Flume has peer-to-peer communication (Apache flume.2017; Apache storm.2018) whereas Spark, Storm, and Flink are Master/slave communication (Spark streaming programming guide.2018; Carbone et al., 2015), and Kafka streams is Publish/Subscriber topology (Apache kafka a distributed streaming platform.2018). The exactly once execution semantics of Spark, Flink, and Kafka can ensure the event is just processed exactly-once, where the events are being processed exactly once (Carbone et al., 2015; Yadranjiaghdam, Pool, & Tabrizi, 2016). On the other hand, Storm and Flume execution semantics is at least once, where there is a chance of events being processed more than once (Apache flume.2017; Apache storm.2018).

Some of the attributes of the chosen technologies are presented in the table below:

Table 1. Comparison of the chosen technology on communication, execution semantics and programming language

Name	Support Edge Analytics	Topology	Execution semantics	Programming language
Flume	N/A	Peer2Peer	at least once	Java
Storm	Early/Beta support	Master/Slave	at least once	Java, Clojure, Scala, Python, Ruby
Spark Streaming	Early/Beta support	Master/Slave	exactly once	Java, Scala
Flink	Early/Beta support	Master/Slave	exactly once	Java, Scala, Python, SQL
Kafka Streams	Early/Beta support	Peer2Peer	exactly once	Java

As discussed in section 2.3.3, Edge Computing is suitable for applications demanding low latency, real-time operations, and high quality of service, and is attracting more and more attention (Cheng et al., 2016). Previously, it has been attempted to implement edge computing to Storm, Spark, Kafka, and Flink (Renart, Diaz-Montes, & Parashar, 2017; Q. Zhang, Zhang, Zhang, Shi, & Zhong, 2016). A prototype of SpanEdge was implemented in Apache Spark, in Apache Beam project, Apache Spark and Apache Flink has been used as processing back-ends (Cheng, Papageorgiou, Cirillo, & Kovacs, 2015; Dias de Assunção et al., 2018; Sajjad, Danniswara, Al-Shishtawy, & Vlassov, 2016). Cheng et al. points out “existing stream processing platforms such as Storm, Spark streaming, and S4, are well

designed to process stream data within a cluster in the Cloud, but are not suitable for highly distributed IoT systems in which data are naturally geo-distributed and low latency analytics results are mostly expected." (Cheng et al., 2015). A set of experiments to compare a proposed (locality-aware stream processing overlay network) edge stream processing approach with traditional approach using Apache Kafka and Apache Storm were performed successfully (Renart et al., 2017). As the number of edge nodes can reach up to over thousand distributed nodes in a smart city, it is beyond any current stream-processing platform (Cheng et al., 2015). Despite the entire real-time stream processing is in the state of the art, there is still a void yet to be filled to meet the requirements of IoT edge analytics (Cheng et al., 2016).

3. Research method

The objective of this thesis is to map industry uptake in Streaming data analytics. As indicated in Chapter two, Streaming data is becoming relevant. However, the emergence of solutions and analytics engines does not yet reveal when and how industries are actually going to utilize streaming data analytics. For this reason, a series of semi-structured interviews were conducted. The aim of the interviews was to find out what software experts, or people that are involved in a company where data plays a vital role in their day-to-day work, think about Streaming data analytics in general, and what stream analytics engine do they prefer, if any. Texts based on interviews and observations are formed from an interaction between researcher and participant and can be seen as an act of communication, and every text has messages to be interpreted and described (Graneheim & Lundman, 2004). Hence, Qualitative Research method seems to be most favorable in this situation to explore a conclusion on the overview of specificity on some of the topics. This research focuses on what kind of data software practitioners work with on day-to-day basis, and how familiar they are with big data and streaming data. Semi-structured interviews were preferred over structured interview, and they were conducted to gather information on said topic, as with structured interviews there was more chance in missing cursory details (Graneheim & Lundman, 2004).

3.1. Interview questions

The interview questions were primarily based on the publications used in this research. The questions were mostly open ended so the interviewee could elaborate on the particular question the way they wanted. The goal was to make the interviewee relaxed and open freely about the topic, feeling comfortable. The first two questions focused on knowing the interviewee and what their role was in their specific field of work. It also was meant to understand their familiarity with the topic that was going to be discussed. All of the interviewee were asked to freely add anything at any duration of the interview without hesitation.

Basic layout of the questionnaire (slight variations from interview to interview)

- Q: What are the domain (ICT or something else) areas the company operates?
- Q: How would you characterize your data? On the basis of 4V’s in big data?
- Q: When you deal with all these data, which analytical engine/tool you use? Seeing the characteristics of the data you deal with, which analytical engine you deploy or use?
- Q: Do you currently have any problems or limitations in your models in your company?
- Q: What are the few questions you want your datasets to answer?
- Q: How familiar are you with Streaming data analytics engines? And if yes, which of the streaming engine do you currently use in your company? Do you have a favorite?
- Q: Is your decision-making a data-driven process? If yes, to what degree?

3.2. Interview tools and setup

Interviews are the most commonly used method of data collection (DiCicco-Bloom & Crabtree, 2006). The first stage of the interview setup requires identifying the proper and right candidate for the interview (Whiting, 2008). The sample interviewees should be homogenous and share critical similarities related to the research questions (DiCicco-Bloom & Crabtree, 2006). The candidates were selected very carefully by knowing their qualification in software/data analytics field. One of the interviews was done face-to-face, two interviews via SKYPE™, and four of the interviews were done over telephone. The interviews conducted were according to the preference/convenience of the interviewee, in their work place, home or in the University of Oulu. The interviews (face-to-face interview, the skype interview and telephone interview) were recorded using tape recorder and by in-built phone recorder in the smartphone as a backup. Both the recording tools were tested prior to the interviews to check if they worked properly. The recorded interviews were transcribed manually without the assistance of any software. The transcribed data were then managed using the software Nvivo ™, which is a tremendous help in the data analyzing process (DiCicco-Bloom & Crabtree, 2006).

3.3. Interviewees

All the participants for the interview were selected and contacted together with the author and supervisor. The necessary condition for the selection of the interviewee was at least 5 years of experience and knowledge in dealing with large amount of datasets at their workplace. This screening process ensured that every interview yielded useful results. All the interviewees were male, age ranging from 25-55.

Table 2. General information about the interviewees

Interviewee (IE)	Country of Residence	Gender	Domain of Expertise	Title of Position	Experience
IE01	Finland	Male	Smart IoT	Board member	10+ years
IE02	Finland	Male	IT and Services	Director, Post-Processing product	10+ years
IE03	Finland	Male	Video Artificial Intelligence	CEO and Founder	10+ years
IE04	Finland	Male	Analytics and Algorithms	Chief Executive Officer (CEO)	10+ years
IE05	Finland	Male	IT, Data and Analyst	Architect Information management and Analyst	13+ years
IE06	Finland	Male	Software Development and Data Mining	CEO and Founder	12+ years
IE07	Finland	Male	Software Development, IoT Development, Analyst	Chief Technology Officer (CTO)	10+ years

As shown in Table 2, all the interviewees were male and resident of Finland. As per necessity, all the interview candidates had more than 10 years of expertise in IT sector. The table also shows the title of position of the interviewee.

4. Results and Findings

The result of the research is presented here in the order of research questions. All the conducted interviews were analyzed in NVIVO software (Castleberry, 2014). Each interviewee was assigned a number and their interviews were assigned an ID. So, for interview number 1, the corresponding ID would be IE01, for interview number 2, the corresponding ID would be IE02, and so on and so forth.

4.1. Types of data and familiarity with streaming analytics

Here we dive into the research question 'RQ1: What are the reported types of data the Information and Communication technology (ICT) companies deal with in day-to-day basis and how familiar are they with streaming analytics?' To understand this better, the interviewees were asked questions such as, 'what are the domain (ICT or something else) areas does the company operate?', 'When you deal with all these data, which analytical engine/tool you use?', and 'How familiar are you with Streaming data analytics engines?' The first question aimed for the general understanding of the type of data the company operated on. The second question was to understand what sort of tools are used by the company to deal with the data in hand, and the third question was aimed to know how familiar interviewees are with streaming analytics, if at all.

ICT domain

All the interviewees were associated with IT companies. Some of them were very thorough in explaining what their respective companies dealt with.

"We operate in the telecommunications. That's the industry way we are, more specifically our main customer segment is the mobile operators, inside the operator we are mainly interacting with the departments that are responsible for the radio access networks."- IE02

"It is a video AI company basically, and we are doing cloud based as well as own produced software products that involve video artificial intelligence, meaning that we are doing recognition system understanding system as well as, applications that are related to video data management or re-proportion or reuse."- IE03

Data analysis, data visualization, IoT systems, telecommunication, and software engineering were the most common answers among others. One of the company admitted on working with video data management, with terabytes of data.

"With industrial data, industrial processes where the production process has lots of sensors that are integrated into the process and then there is information that is available before starting/producing the goods, and information after producing the goods, so, this is producing a lot of data, not very heterogenic data but any ways, big amounts of data…..Then in sales and marketing, the data are sometimes quite unstructured, you can have observations from many different things...Then in telecom industry, you have again side of the industry that is producing the

equipment, so that is kind of fabrication or different factory, like steel industry, but producing telecom equipment, and then there is the research and development side where the equipment is tested and lots of data throughput is observed, how does it perform? How does the equipment perform? What are the limits of the equipment? And how does the signal perform in the equipment."-IE04

"Basically we are Telco (Tele-communication), as you know, but we don't stick with telco alone. We are also communication providers. We have services like, IPTV, then we have Digital library services, video conferencing services and cloud services, so there are lot more, but primarily a telco service provider."-IE05

"Data analysis, visualization, work consulting, troubleshooting, problem solving, training, of course to do all these, we do software development, (laughs), yea we are like data service company" -IE06

"I think we are kind of building IoT systems and analytics based on the collective data and the actual business domain is, we are doing things from the electronics to the code level." – IE07

Regardless of various ways of explaining, the common theme of the ICT domains the interviewees dealt with on the daily basis was, Big Data. Some of the interviewees were more familiar with data analytics and streaming data analytics. From the answers, it is abundantly clear that these software practitioners were more than familiar with Big Data and dealt with large sum of data on daily basis.

Types of data

All the interviewees admitted on dealing with Big Data one-way or the other. Either the company they were associated with produced immense amount of data or the work they do in the company involved dealing with a large quantity of data. Some of the interviewee simply answered 'yes' when asked if they deal with Big data and some of the interviewees go into details about what kind of data they deal with in their work place. Below, some of the snippet of the interview transcripts are presented where the interviewees discuss about the types of data they handle in day-to-day basis, and source of said data.

"The scenario is such that we are collecting data from the network itself and there data is being generated terabytes per day and in largest scale we are operating within few tens of terabytes of data." – IE02

"So, I think content data is big data, so there are lots of video material alongside of the other kind of data sources. So, big data is not only this, kind of transaction or event based gathering or time series but it can also be huge repositories of content to be produced." – IE03

"But in a similar way, we also have a lot of unstructured and semi-structured data. Like in semi-structured, we have XML, CSV's and all those stuffs, JSON's for example. We don't have much of unstructured data, but there are unstructured data like log files. There are some cases we also store documents in the big data platform.

So, there the variety is a bit more and if you look at the velocity, there are also many use cases where we do real-time integrations. So, when it comes to real-time integrations, we usually go to big data platform. We don't often go to traditional data warehouse, so the big data platform is used, for example one use case what we are handling, it is almost handling 10,000 messages per second. We also do real-time analytics on top of that when we are streaming the data in."- IE05

"Actually, we are using, well big Data is a little bit wide concept, we are using the weather data and we are using the video data from the measurement sites. And also, what we are doing there is, we are collecting the vibration data from the road itself. From the sensors it is kind of non-stop stream basically." –IE07

There was one interviewee who admitted on not knowing the content of the data as it is their customer's data, but the size of the data was large. This is inferred from the following snippet:

"Definitely for the reporting but we don't know what was the data size, okay we know for example 50 bytes or 180 bytes, we don't know what the data had. We don't know the content of the data. Basically, we can say publicly about 1 Billion messages per month we are relaying. Our customers they have the data but we are just handling it on their behalf, so we don't have the visibility or in a sense access to that data."- IE01

One of the interviewees admitted on not necessarily working on Big Data at the moment, but said there were projects lined up in the future where it is required.

"We do Process data, like for Process industry like paper machines and chemical processes…. we are not a big data company, but our stuff usually fits into MySQL, but there are cases where we need this Big Data stuff, to give you some idea about our Big Data stuff, I have done only one project with Big Data, but now we are working on another one…. now we are working with them (Helsinki Region Transport) on an open source platform which would be collecting passenger movements, from of course who accepts in where they are in public transport, and that would be a big data. That would be like one millions rows of data a day." –IE06

From the responses of the interviewees, it can be clear that the participating companies are already handling large amount of structured, semi-structured, or unstructured data. Six of these companies handle Terabytes of data per day, among which, some handle large amount of video material along with other data sources, and some deal with non-stop streaming data. One company that do not deal with Big Data now, admit on getting a potentially large project involving Big Data in the very near future. The responses from the interviewees here clearly indicate that companies interviewed deal with large amount of data and are potentially right candidates for this study. Table 2 summarizes this section in an extensive way.

Table 2. A summary of ICT domain/types of data the interviewees dealt with

Topic	Meaning unit	Condensed meaning	Theme
ICT domain/Types of data	"…We operate in the telecommunications… We operate in the telecommunications"	Data analysis, data visualization, IoT services, Telecommunication, large amount of structured and unstructured data, etc., openly admit, and understand they deal with big data.	Currently working with big data or big data projects lined up in near future.
	"we are doing cloud based as well as own produced software products that involve video artificial intelligence… there are lots of video material alongside of the other kind of data sources"		
	"…industrial processes where the production process has lots of sensors that are integrated into the process… not very heterogenic data but any ways, big amounts of data…"		
	"We are also communication providers… we have Digital library services, video conferencing services and cloud services… we also have a lot of unstructured and semi-structured data… We don't often go to traditional data warehouse, so the big data platform is used"		
	"Data analysis, visualization, work consulting, troubleshooting, problem solving, training, of course to do all these, we do software development, (laughs), yea we are like data service company"		
	"…we are kind of building IoT systems and analytics based on the collective data… we are using, well big Data is a little bit wide concept, we are using the weather data and we are using the video data from the measurement sites"	Handle large amount of data themselves or on behalf of someone without knowing the content. Future projects lined up include big data projects.	
	"…We don't know the content of the data…Our customers they have the data but we are just handling it on their behalf, so we don't have the visibility or in a sense access to that data… we can say publicly about 1 Billion messages per month we are relaying…"		

4.2. Data/Business needs

Here, we delve into the business needs or the data needs in the interviewees respective businesses. To understand this topic in a more specific way, the questions that was asked to the interviewee were on the line of "What would you say your current needs (business or technical) from the data analytics tools are?"

"Well, our needs are, we need performance and scalability for large volumes of data and we also need practical scenarios of these important points, we need easy deployment, so we need something that we can quickly deploy and set up the system without long integration project to integrate into our customer systems.
We want our data sets to answer multiple questions, let me give you few examples, we want to know, what is the performance of the network in function of time and place? We want to know, what are the root causes of bad performance?"- IE02

"It has to be scalable, it has to be able to have scale up capabilities, so that the processing time can be flexible and can in-line with the increase of users, number of users that are using the analytics. Then it has to provide highly accurate models, and have fast set-up and training time. So, the processing time for the data what need to be preferably very quick, as quick as possible."- IE03

Scalability, and integration time and training time were one of the major focuses in two of these companies. Retaining the existing customer base and expanding their business without interruptions to the services were also mentioned.

"So you have to retain your customers from leaving your service…Next Best Service that is something you want to do the upselling and cross selling for our customers…we have to be very careful in providing services to our customers without any interruptions. So, that's the reason we constantly monitor all these base stations and the core networks and the radio networks. And basically you evaluate, there are models which constantly getting inputs from these instruments. We have various way of calculating the thresholds, and we take appropriate actions…"-IE05

In one interview, the interviewee was very specific about a project his company was involved in. The project was about collecting streaming data from the sensors on a road that would give them insight on many things regarding the condition of the road, vehicle travelling on the road, any sort of disturbance that would create any problem with that particular road, etc. This is the project where the prospect of edge computing was being looked into.

"I think the simplest question is that; we want to know is 'Is the road okay?'. And the other question is that, what's on the road? I think these are the simplest questions, and I think the answers for those are a little bit difficult and a little bit more complex." –IE07

The most common theme for this question was retaining the customers, or keeping the customers happy while steadily growing as a company. Few interviewees wanted to extract more information from their already existing data sets, increase the

performance of the network. Table 3 summarizes this section in a comprehensive way.

Table 3. A summary of Data/business needs of the interviewees

Topic	Meaning unit	Condensed meaning	Theme
Data/business needs	"…we need performance and scalability for large volumes of data… we need something that we can quickly deploy and set up the system without long integration project to integrate into our customer systems…"	Scalable, less time consuming, easy training, very quick data processing	Customer satisfaction, increasing their business, scalable and less time-consuming services.
	"It has to be scalable… flexible and can in-line with the increase of users, number of users that are using the analytics… provide highly accurate models, and have fast set-up and training time. So, the processing time for the data what need to be preferably very quick, as quick as possible"		
	"…providing services to our customers without any interruptions… upselling and cross selling for our customers"	Customer satisfaction and gaining more customers	
	"…we want to know is 'Is the road okay?'. And the other question is that, what's on the road?"	Project specific satisfaction to the customers	

4.3. Data analytics method

To understand what sort of data analytics method (if any) the interviewees in their respective companies implemented. The question such as "What kind of data analytics model/tools, based on the data characteristics you just mentioned, does your company use?" or "What kind of data analytic models/tools your company is using right now?" were asked. Six interviewees responded with the data analysis methods used in their respective company.

"One important thing and the basic thing is just to visualize the data. So, data visualization is, one important thing, without actually further processing the data, big part of the data analysis is just to visualize the data….the actual data analytics, aggregation, calculating average, and minimum, and maximum values for the kps, that's important. Because we are dealing with terabytes of data, we always usually have to aggregate it some level to reduce the amount of data so that it can be handled more easily…we do not use streaming data analysis right now"- IE02

"talking about big data analytics frameworks, which typically entail some kind of map reduce derived processing paradigms, where the basic paradigm is to distribute the data and put the analysis where the data is and then aggregate the result. So, in that kind of framework, we are little bit similar but not really 100% following that paradigm…We have our own architecture in place, and we are interested in better

understanding stream processing frameworks, like Spark or any new stream processing"-IE03

"We use all the typical tools in the field, like, SQL, adobe in some cases, then R language, Java script, etc. So, the thing that we do is, we select a tool that are most suitable for the particular problem, then work on from there... It is rather so that, you don't first have analytical model and then try to fit that into something, but you must learn what is the problem that you are studying and then select the proper mode…We have tested and developed with spark… we tried using spark and storm about an year ago, and we ended up choosing spark… storm didn't support so many different languages or it was kind of difficult to, there was some kind of limited support for the tools in storm than in spark" -IE04

"If you look at the analytics model, from the IT department, we are more supporting descriptive and diagnostic analytics, so there we are end to end service provider, but when it goes to advance analytics, like predictive or prescriptive in nature… More about time series analytics, very traditional analytics like descriptive or diagnostic in nature, we already provide end-to-end service for our users. At least I can give you these information, we are using CHURN modelling, Next Best Service model, and there are also some predictive models, particularly on the areas of financial management, there are predictive incident management models. Basically, if you look at MAPR streams, that's an enhanced version of Kafka. Because, basically what MAPR does is, it takes all these open source components and then enhances it, and then sells it as a product. MAPR streams is very similar to Kafka, but they have made some modifications to it. So, that's the streaming engine what we are using… But in ELISA we also use FLUME, for example, that the another one, but it's not real time streaming"-IE05

"We use two kinds of tools. When we need to deal with big data we use Apace Spark and when we work with MySQL kind of data then we just do, we are running batches, we are running like 50,000 rows at a time, yea other way we work with large files." –IE06

"Actually, we are using the Cyclone, it is based on the Intel AI environment, it is kind of mixing Watson and R and some other maintenance functionality, so it is kind of mix up things… We also use Ansible to handle our analysis; it is an environment for the server maintenance." –IE07

There was variety of answers for this part. One of the interviewees talked about the importance of data visualization and then proceeding to data analytics. Aggregation, descriptive, and diagnostic analytics seemed standard with these companies. MYSQL, R, Apache Spark, MAPR, Ansible, Storm, Mango, Cyclone, etc. were some of the tools mentioned by the interviewees mentioned. Among them, Apache Spark and MYSQL seemed to be most commonly used. Table 4 summarizes this section.

Table 4. A summary of Data analytics method discussed by the interviewees

Topic	Meaning unit	Condensed meaning	Theme
Data analytics method	"…big part of the data analysis is just to visualize the data… the actual data analytics… we always usually have to aggregate it some level to reduce the amount of data so that it can be handled more easily… we do not use streaming data analysis right now"	mostly working with batch processing and not using stream data analytics	Except for one company, most using some type of streaming data analytics tools.
	"…map reduce derived processing paradigms, where the basic paradigm is to distribute the data and put the analysis where the data is and then aggregate the result… We have our own architecture in place, and we are interested in better understanding stream processing frameworks, like Spark or any new stream processing"	own architecture in place but interested in understanding stream data analytics better	
	"We use all the typical tools in the field, like, SQL, adobe in some cases, then R language, Java script, etc…, We have tested and developed with spark… storm didn't support so many different languages or it was kind of difficult to, there was some kind of limited support for the tools in storm than in spark"	worked with most tools, prefers Spark over Storm	
	"we are more supporting descriptive and diagnostic analytics… we are using CHURN modelling…MAPR streams is very similar to Kafka, but they have made some modifications to it. So, that's the streaming engine what we are using… But in ELISA we also use FLUME, for example, that the another one, but it's not real time streaming"	Prefer MAPR streams to Kafka, also uses Flume	
	"When we need to deal with big data we use Apace Spark and when we work with MySQL kind of data then we just do, we are running batches"	works with Apache Spark and MySQL	
	"we are using the Cyclone, it is based on the Intel AI environment, it is kind of mixing Watson and R and some other maintenance functionality, so it is kind of mix up things… We also use Ansible to handle our analysis; it is an environment for the server maintenance"	Using Cyclone, and using Ansible to handle analysis on server maintenance	

4.4. Limitations of current tools

We asked questions such as “Would you say you are satisfied with your analytics model?”, “Do you happen to know any problems or limitations in current model your company is using?” to know and understand more about the problems or limitations (if any) the company is facing with the data analytic model currently in use.

“We are not satisfied. With the current analytics we have, the main reason is our customers are never fully happy, they always want something more, and what they want is, what we call ‘can’t analytics’, so something available off the shelves, automated data analytics routines that would detect the problems that are seen on the data and also would detect the root cause of these problems, by analyzing the data. So, this is what we are being asked more and more from our customers.”- IE02

The general theme of the response involving this question was that they all wanted to know their options, and were open to new ideas and innovation that could help increase the potential of their company to meet the never-ending customer need.
Some of the interviewees preferred easy-to-use tools and requires quick training. This point can be inferred from the following snippet:

“One of the challenges on deep learning based machine learning solutions is the training time, training is slow, it requires lots of data so that’s one of the challenges we need resolve in order to made more adaptive system… training the models machine.”- IE03

In one of the interviews, the interviewee pointed out how they were having issues with Spark, as the process was non-responsive after having memory issue. Another issue was the limitation of creating only one content at a time, as there was an issue with running multiple projects or streams.

“Yeah, there’s been some memory issues, some inbuilt limitations to how you can allow spark to use memory, so if it says it is lot of memory, the process can just kill itself, and it doesn’t do anything at that point. And then there was the issue of running multiple projects or streams, that’s quite limiting in how you can only create one content at a time.”- IE04

I have not found any limitations with Spark but of course we have limitations with our current main model (MySQL). Because it is quite often that the data does not fit into the memory of the system in one go. So, we do have problems where we need to use something else than just having it all in the memory. – IE06

“I think the current limitation is, it is pretty damn expensive, so we are trying to make it a little bit cheaper…I think our problems are more mechanical than data side. The future would be, we do most of the analytics inside the box and we send kind of refined data, like this is the data we actually want and these other data is noise and we drop that. And sending the data to the cloud for accurate and detailed analytics. The future would be, we do most of the analytics inside the box and we send kind of refined data, like this is the data we actually want and these other data is noise and we drop that we are using this edge computing more and more” –IE07

One of the interviewees using Spark for few Big Data related projects said they have not found any flaws with it yet. They found flaws with MYSQL that they use on regular basis for other smaller (not necessarily Big Data) projects. One interviewee complained about the mechanical device they are currently using was expensive. This was a mechanical limitation rather than data analytics. However, they were planning to tackle this limitation by adapting edge computing in the future. This was the first instance where an interviewee talked about edge computing, and how beneficial it was going to be for their company. Table 5 summarizes this section.

Table 5. A summary of Data analytics method discussed by the interviewees

Topic	Meaning unit	Condensed meaning	Theme
Limitations of current tools	"We are not satisfied… automated data analytics routines that would detect the problems that are seen on the data and also would detect the root cause of these problems, by analyzing the data"	Easy and quick to train solutions, wanted to know about more options and innovating ideas to make customers happy.	Difficult to operate tools, never ending customer needs, expensive to operate tools.
	"…training is slow, it requires lots of data so that's one of the challenges we need resolve in order to made more adaptive system… training the models machine."		
	"there's been some memory issues, some inbuilt limitations to how you can allow spark to use memory… then there was the issue of running multiple projects or streams, that's quite limiting in how you can only create one content at a time"	Some memory issue with Spark	
	"I have not found any limitations with Spark but of course we have limitations with our current main model (MySQL). Because it is quite often that the data does not fit into the memory of the system in one go"	No limitations with Spark but with MySQL and in the other case mechanical limitations and looking forward on using Edge computing in the future to solve this issue.	
	"…it is pretty damn expensive, so we are trying to make it a little bit cheaper…I think our problems are more mechanical than data side…The future would be, we do most of the analytics inside the box and we send kind of refined data, like this is the data we actually want and these other data is noise and we drop that we are using this edge computing more and more"		

4.5. Awareness and familiarity of Streaming analytics

Here, we asked the interviewees their familiarity with streaming analytics. It is safe to say, some of them were unaware about it, whereas some of them were accustomed to it and had their own share of reaction to it.

Here are some of the responses where the interviewees were not familiar to streaming analytics in their work place. When asked why, they said they were still relying upon traditional data analytics or batch processing.

"Well actually not to that extent, no."- IE01

"I am not familiar at all and as far as I know we do not use streaming data analysis right now. The whole Keysight I have no idea but speaking of, as far as I know the network testing business unit, where I am working, as far as I know we do not use such tools currently."- IE02

Here are some of the responses of the interviewees that were familiar with Streaming analysis, and they were the one to talk in detail about streaming analytics that they use in their day-to-day workspace.

"We have tested and developed with spark."- IE04

"We have our own architecture in place, and we are interested in better understanding stream processing frameworks, like Spark or any new stream processing...we are interested on the spark because we believe that there might be solution but we would be interested in evaluating different frameworks, like which one is the most useful for distributed clusters"- IE03

"For streaming, what we are using is MAPR streams, MAPR is a HADOOP based vendor and they are providing real time streaming component called MAPR streams. So, we are using that product in ELISA for streaming. And for doing analytics on the streaming data, we are using SPARK."- IE05

"The thing I did for Norway used Spark streaming technology, but we don't have a business on that. It was just a project, but I think if and when we proceed with Helsinki Regional Transport (HSL) then we will be using the streaming tools of Spark… the other main one is quite manually working with MySQL. Case by case we need to solve this things. So what you are looking for Spark is to reduce this manual work to make you know, lighter." –IE06

"We are using so many different analytics tools in software environment. I think the easiest way to say would be, we don't use Google's environment (long pause) yet (laughs)… Yea, we are using Spark for anomaly detection, so in that way, yes. But Watson was the easiest way, and IBM was giving us that play money, that's why Watson (laughs). Storm we are using kind of a data streams. Storm is handing the data collection level and then all the data receiving level… We are using Spark and Mango, and all these aggregation models Mangos is providing and so many things, this will take a whole day (laughs)." –IE07

Two of the interviewees claimed they were aware of Streaming analytics, but had not personally worked with it. They admitted on still working on Hadoop or HDFS for batch processing of data. The rest of them were very familiar with streaming analytics, and had worked in one or more projects with it. Some of the interviewees said they have their own architecture to work with when it comes to streaming data

analytics. One of the interviewees mentioned how they had to work with IBM's Watson analytics, as IBM was giving the 'play money' for them to work on Watson. Among the technologies that was chosen in this dissertation to discuss, Spark streaming seemed to be more favorable to the interviewees in their companies, among the technologies. Some of the other streaming analytics tools claimed to be used by the interviewees were Storm, Mango, MAPR, MySQL, and Watson. Table 6 summarizes this section.

Table 6. A summary of Awareness and Familiarity of Streaming Analytics

Topic	Meaning unit	Condensed meaning	Theme
Awareness and Familiarity of Streaming Analytics	"Well actually not to that extent, no"	Aware but not familiar with many data streaming tools, but certainly keen to understand more.	All of them were aware of Streaming analytics and with the ones that worked with Streaming analytics, Spark streaming seems to be more popular, among others.
	"I am not familiar at all and as far as I know we do not use streaming data analysis right now"		
	"We have our own architecture in place, and we are interested in better understanding stream processing frameworks… we are interested on the spark because we believe that there might be solution but we would be interested in evaluating different frameworks, like which one is the most useful for distributed clusters"		
	"We have tested and developed with spark"	Familiar with Streaming analytics, worked with tools like Spark, Storm, Kafka, MAPR streams, Mango	
	"For streaming, what we are using is MAPR streams… for doing analytics on the streaming data, we are using SPARK"		
	"The thing I did for Norway used Spark streaming technology… if and when we proceed with Helsinki Regional Transport (HSL) then we will be using the streaming tools of Spark… the other main one is quite manually working with MySQL…"		
	"We are using so many different analytics tools in software environment… we are using Spark for anomaly detection… Watson was the easiest way… Storm we are using kind of a data streams. Storm is handing the data collection level and then all the data receiving level…We are using Spark and Mango, and all these aggregation models Mangos is providing and so many things"		

4.6. Data in decision-making

Here, we asked the interviewees to elaborate on how their company makes its decisions, or to what degree the company's decision is driven by data analytics. Three out of seven interviewees admitted on data playing an integral role in the company's decision-making, yet it was not based on just data analytics but also

intuitions and experiences of the people involved in the company, or the one in the position of making the decisions.

"I think it is about both, and if you think about we are using the data or certain text analysis…. So, it's basically both meaning that testing and analyzing the data and also knowledge that we have been collecting all these years."- IE01

"It is based on experience and intuition, but in reality it is something in between."- IE02

"Basically, if you look at the company, I would say it is 50-50. We want to be 100% data driven, but we are not yet there. In certain cases, yes, we still reply on the data, but in certain cases where we don't have that competency, so we just go by industry trends, best practices, what's there in the market, vendor's proposal and reference cases."- IE05

There was one answer where the decisions made by the company was almost solely dependent data analytics.

"It is based on data streams, definitely, we have work around video data and all the solutions are kind of revolving in that data type."- IE03

Two of the interviewees admitted on being a small company with very little moving parts and just about seven employees. They said, they rely on experience and intuitions to make decisions rather than data analytics, as they do not produce huge amount of their own data, and the owner of the company knows everything inside out.

"Because we have so little data or moving parts in our company that I know all the data that is affecting the decision making inside out, so I don't need any big data systems to handle our data. But basically, we know what the tools are that we are using, we don't produce any data ourselves, so of course we track the complexity of some tasks or projects or estimate how much time something will take to develop, and that kind of data we use in our decisions."- IE04

"It is not data-driven, it is just old fashioned management, talking with people and stuff." –IE06

One interviewee admitted on every now and then coming up with some crazy idea and trying it out, instead of relying on anything.

"We get an idea, we test it and if it works then we sell the idea to the guy who needs it. So, that's our decision making process." –IE07

This is a very complex topic and to present it here without any more detail or research in this particular topic, would be doing it injustice. We have asked one question to all the interviewees about their decision-making process in their company and did not go into details. Out of seven interviewees, four admitted on their company having some sort of data-driven way of decision-making approach, whereas

three interviewees admitted on having traditional way of decision-making. Saying that, there are too many moving parts surrounding this topic, and to simplify this very complex topic about decision making in any company with one question would not be fair in any way.

4.7. Frequency table

In this section, we have put down some of the keywords words or phrase mentioned by the interviewees to give an overview on the number of interviewees mentioning certain terms or phrases. Some of the keywords chosen for the frequency table are the analytical tools that we have selected in this thesis as candidate technology. This is done in an effort to inform the reader about the extent of knowledge or familiarity interviewees had with the chosen technology. Merely mentioning the term or phrase does not imply the interviewee were using the tool itself, but were aware of the tool. During the interview, the number interviewees mentioned a certain word or phrase while discussing the questions. This table could provide reader some impression on the findings and the number of interviewees mentioning it. This table was generated by NVIVO^{TM}. Some of the most repeated words, that has been omitted from this table are ‘data’, ‘types’, ‘challenge’, ‘business’, ‘tools’.

Table 7. A keyword frequency table

Keyword	Keyword mentioned/Total number of interviewee
Big data	6/7
Data analytics	6/7
Streaming data	7/7
Flume	1/7
Kafka	1/7
Flink	3/7
Storm	3/7
Spark	6/7

From Table 7, it is abundantly clear that all the interviewees were familiar with the terms Streaming Data, Big Data, Data analytics. Almost all of them used the term in various ways throughout the interviews. It also speaks volumes about the interviewee’s familiarity with the data analytics tools and the ones they all mention. Spark was mentioned by 6 out of 7 interviewees, Storm and Flink was mentioned by 3 out of 7 interviewees, and Flume and Kafka were mentioned by 1 out of 7 interviewees. This gives us some indication on the interviewees’ familiarity of the data analytics tools.

5. Discussion

The results of this research are subjective in nature. The outcome of the research has provided a unique insight into how the participating companies analyze and utilize data, both in their operations as well as in their products and services. In this chapter, a parallel is drawn between the subjective results of the research and what has been reported in the existing literature.

5.1. Reaching the objectives

This research focused on gathering insights into the chosen technology to get macro view from the company's perspective. The aim was to gather as much information from the software practitioner's point of view on the sort of data these companies deal with on daily basis, and the current problems faced by them with their existing solution. The insight provided from this research could help the companies understand which of the existing technology could address their issues better. In this research, to what extent the companies are familiar with the technological landscapes in Streaming data analytics that are available in the current maturity level was recognized. We also compare the chosen technologies, and point out the similarities and differences among them. We were able to find out to what extent the companies were familiar with the technological landscape available today. We were able to inquire about to what extent the candidate company's decisions were data driven. Although we were able to acquire valuable information and insights from the candidate company interviews, there is significant variance in how companies understand Streaming analytics. As some of the interviewees were unaware of streaming data analytics tools, we could not go into much detail about it. From the response of the interviewees and the interests they expressed in understanding more about Streaming data and analytics tool, it was obvious that they were contemplating of adopting it in the near future. With the diverse response in the interviews about Streaming Analytical tools, and each of the tools having their own competitive edge, it is extremely difficult to say, conclusively, which is better among the existing technologies.

The technologies mentioned in Table 1, viz. Spark, Storm, and Flink are open source processing frameworks that can be used for Stream analytics. Kafka and Flume are data ingestion tools and have a vital role in streaming data analytics (Yadranjiaghdam et al., 2016). Due to high performance of Spark and Flink, they both can be used for batch processing and stream processing. Storm is the streaming analytics engine that supports micro batching, which is s special kind of batch processing where the batch size is orders smaller. Despite having a latency of few seconds, Spark has more maturity, performs well on dedicated clusters when entire data can fit in the memory (Yadranjiaghdam et al., 2016). Apache Flink, despite having a slight edge among other analytical tools with very low latency and exactly-once semantics, was not the obvious favorite, as it had maturity issues. However, Spark streaming has good performance, and maturity wise it is more practical despite few seconds of latency. Spark streaming was one of the favorites among the interviewees, out of the chosen five technologies. Table 7 helps shed more light on familiarity and preference for Spark among other analytical tools. Considering

theoretical evaluation and the interviews, Spark Streaming seems to have a slight upper hand in regard to the familiarity and maturity level as well. There are still plenty of work to be done with Spark, as some of its flaws, such as issues of running multiple projects or streams and memory issues, were pointed out. Most of these companies still have reservations about these tools, and which one of the existing tools would be suitable for their business. Some of the companies involved in this research, despite their involvement in streaming data analytics, were not fully ready to accept the new data analytics tools and are still dependent on traditional tools like Hadoop and HDFS. One aspect that was very clear from the interviews was that the companies that did not use Streaming data analytics yet, were very keen to know the best solution for them in terms of their needs. The companies that were already using various data analytics tools were keen on knowing more information on the applicability of the different analytical tools and the solutions. The interviewees were aware of the fact that they need to embrace the data analytics tools, and it is the step forward for their company's future.

5.2. Limitations

Similar to every research method, the qualitative interview research method has its limitations. Despite of the efforts to increase the quality of data collection, there is always a doubt about the nature of the analysis. Unlike quantitative analysis, at the core of qualitative analysis is a creative process that almost entirely depends on the insights and conceptual capabilities of the analyst (Patton, 1999). The major factor that could come in play as one of the biggest limitation of this research would be the interviewees not disclosing information. One of the interviewees openly admitted that he was not at liberty to share certain information, as it would be against the company's policy.

One could argue that there were some limitations in the survey, as the questions were limited and similar for all the interviewees, irrespective of their job title or description. The structure of the interview questions varied from interview to interview according to the answers provided. Interviewees that were familiar to different Streaming Analytical tools were asked relatively more questions than those that were not familiar with it. There were limitations regarding the number of interviewees and the geographical location of the interviewees. The number of interviewees that participated in this research was limited, and most of them were from Finland. This limits the research scope within one country and the result might not be applicable elsewhere. There is no empirical benchmark in this thesis, as there is no verifiable observation or experiment, and no standard point of reference for all the involved technologies. However, we compared the technology through a theoretical framework consisting of architectural details.

5.3. Threats to validity

In qualitative research, trustworthiness is vital and trustworthiness will increase if the findings are presented in such a way that lets the reader look for alternative interpretation and conclusions (Graneheim & Lundman, 2004). We have included a snippet of all the interview answers while making any sort of claim, to allow the

reader to look for their own interpretation. Validity in qualitative study is quite a challenging task on many levels. However, qualitative inquirers need to exhibit their research are trustworthy (Creswell & Miller, 2000). One of the most important tasks of the interviewer is to obtain information while listening and encouraging the interviewee to speak more open and freely (DiCicco-Bloom & Crabtree, 2006). If interrupted while the interviewee is commencing, the interviewee may lose his/her chain of thought that might result in losing valuable information. The study by Creswell and Miller (2000) suggests that two perspectives govern the choice of validity procedures. First one is the lens researchers use to validate their studies, and the second one being researchers' paradigm assumption (Creswell & Miller, 2000). Keeping this is mind, additional precaution were taken during the interviews to not have leading questions, so that the interviewee would not feel like the interview was a pretense to prove a preconceived theory.

To make this research transparent, we have discussed all the threats and validity in this particular research. Maxwell (1992) says there are five broad categories that could help understanding validity of any qualitative research. The categories are descriptive validity, interpretive validity, theoretical validity, generalizability, and evaluative validity (Maxwell, 1992). Descriptive validity means if the interviewee is in fact telling the truth and not making it up. This is the most important aspect of validity as every other validity depends upon this (Maxwell, 1992). In our interview, all the interviewees appear to be telling the truth, as they had no valid reason to lie or make up any of the things they said in the interviewee. Interpretive validity means the way the researcher interprets the gathered information is innately a matter of interpretation from the words and actions of participants (Maxwell, 1992). To avoid this as much as possible, snippets of interview transcripts have been provided. Since it is a qualitative study and entire research was a subjective study on Streaming data analytics – background, technologies and outlook, an objective observation and conclusion could not be guaranteed.

Theoretical validity is theoretical explanations that are developed from the studies, which fits the data that are gained during the process of collecting information. This also means the validity of the blocks where a researcher builds a model (Maxwell, 1992). We have tried to avoid this conflict as much as possible, as our study points out the differences in our theoretical technology analysis and our interview analysis.
Generalizability means the extent to which interviewer generalizes one study or case to other even if the extension is not possible or absolute. In qualitative research, there are two aspects of generalizability: generalizing within a group or a community studied to persons, events, or situation that were not interviewed or observed (Maxwell, 1992). Here, we have avoided any type of generalizing altogether, as all the interviews were conducted in a same manner, and with no relation with previous interviews. Therefore, we can say generalizability does not apply in this research. Evaluative validity means if the researcher was able to understand and interpret the interviewee without making any sort of preconceived judgements (Maxwell, 1992). Being a qualitative research, it is evaluative in nature but a little to no preconceived judgements were in play while doing the evaluations in this thesis.

5.4. Future work

For future research, findings from this thesis could be used to further study and develop a joint project with some of the companies, and design and implement a streaming analytics solution for them. The entire process could be put together as a case study, examined and analyzed from the economic point of view as well. This will give us a unique understanding to the investment cost of implementing such technology, to a potential future growth of a company, as many companies are skeptical about the initial investment and deem it as a too big of a risk. To understand this topic in a broad way, empirical benchmarking with the presented technologies in co-operation with few companies can be done. The data-driven decision making question in our interview was not enough to understand the data driven aspect of company's decision-making process. An extended interview or a survey with larger company pool to understand, 'How decision-making is becoming data-driven?' could be one of the many topics for further exploration. The observation made here on ethical misconduct of data analytics, for example voting behavior, also opens up new study possibility on ethics and good conduct/governance in utilizing streaming data analytics and 'how to manage this in practice?'. How companies that collect tremendous amount of data are not just responsible for the safety of the data but also responsible for ethical misuse of it. The findings and observations of this dissertation can be used as inputs in various ways for further research purposes.

6. Conclusion

The rapid growth in technology and complexity of data in today's context led to data analytics in real time, which in-turn led to this thesis. In contrast to the traditional batch processing of data, stream data analytics processes a large amount of continuous data. In this thesis, we have chosen streaming data analytics as a topic to discuss its background, involved technology, and outlook. This was achieved by analyzing the theoretical aspects of Streaming data analytics, and what sort of principles and data pipelines it has. Five existing streaming data analytics tools were chosen, compared, and analyzed. Qualitative research method was used in this thesis to learn more about this topic. Interviews with seven IT companies that specialized in data analytics were conducted. Nvivo™ software was then used to assist in analyzing the transcripts of the interviews.

There are very few existing research that have compared the streaming data analytics tools by comparing company's perspective on the matter as well. Taking this as a motivating factor and aiming to contribute on this topic by analyzing the streaming data analytics technology individually and getting the insight from the companies that are involved in data analytics. From the theoretical review of the chosen technology, Spark Streaming seems to have slight edge among other streaming data analytics tools, with more maturity in technology despite having few seconds of latency. The result from the interviews was not very conclusive, but was in slight favor of Spark streaming as well. Some of the companies, despite their involvement in data analytics, are not fully ready to accept the new data analytics tools, and are still dependent on traditional tools like Hadoop and HDFS. All the companies that were still using the traditional analytics tools admitted on being interested in the new streaming analytics tools, and seem to be very curious in getting more information on the applicability of solutions.

The comparison of the existing technology is a big issue and is a part of an existing problem. An attempt has been made in this dissertation, to study it, learn from it and present the findings. The responses of the companies were fascinating to form an opinion. The approach that was taken in this dissertation, gave us valuable insight in how some of the companies are very hesitant on adopting the state of the art. There has always been a gap between practicalities and research findings, and our aim is to lessen this gap. Companies are not always aware of the state of the art, and this issue must be tackled by forming a kind of collaboration between industries and academics or researchers.

7. References

Apache flume (June, 2015). Retrieved from https://cwiki.apache.org/confluence/x/4xfVAQ

Affetti, L., Tommasini, R., Margara, A., Cugola, G., & Della Valle, E. (2017). Defining the execution semantics of stream processing engines. *Journal of Big Data, 4*(1), 12.

Albright, J. (2016). How Trump's campaign used the new data-industrial complex to win the election. *USApp–American Politics and Policy Blog,*

Apache flume. (2017). Retrieved from https://hortonworks.com/apache/flume/

Apache flume. (2018). Retrieved from http://flume.apache.org/_images/UserGuide_image00.png

Apache kafka a distributed streaming platform. (2018). Retrieved from http://kafka.apache.org/intro

Apache storm. (2018). Retrieved from https://hortonworks.com/apache/storm/#section_1

Baldini, I., Castro, P., Chang, K., Cheng, P., Fink, S., Ishakian, V., . . . Slominski, A. (2017). Serverless computing: Current trends and open problems. *Research advances in cloud computing* (pp. 1-20) Springer.

Bhattacharya, D., & Mitra, M. (2013). *Analytics on big fast data using real time stream data processing architecture* EMC Corporation.

Bila, N., Dettori, P., Kanso, A., Watanabe, Y., & Youssef, A. (2017). (2017). Leveraging the serverless architecture for securing linux containers. Paper presented at the *Distributed Computing Systems Workshops (ICDCSW), 2017 IEEE 37th International Conference on,* 401-404.

Botan, I., Derakhshan, R., Dindar, N., Haas, L., Miller, R. J., & Tatbul, N. (2010). SECRET: A model for analysis of the execution semantics of stream processing systems. *Proceedings of the VLDB Endowment, 3*(1-2), 232-243.

Cadwalladr, C. (2017). The great british brexit robbery: How our democracy was hijacked. *The Guardian, 20*

Carbone, P., Katsifodimos, A., Ewen, S., Markl, V., Haridi, S., & Tzoumas, K. (2015). Apache flink: Stream and batch processing in a single engine. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 36*(4)

Carbone, P., Katsifodimos, A., Ewen, S., Markl, V., Haridi, S., & Tzoumas, K. (2015). Apache flink: Stream and batch processing in a single engine. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 36*(4)

Castleberry, A. (2014). *NVivo 10 [Software Program].Version 10.QSR International; 2012.,*

Cheng, B., Papageorgiou, A., & Bauer, M. (2016). (2016). Geelytics: Enabling on-demand edge analytics over scoped data sources. Paper presented at the *Big Data (BigData Congress), 2016 IEEE International Congress on,* 101-108.

Cheng, B., Papageorgiou, A., Cirillo, F., & Kovacs, E. (2015). (2015). Geelytics: Geo-distributed edge analytics for large scale iot systems based on dynamic topology. Paper presented at the *Internet of Things (WF-IoT), 2015 IEEE 2nd World Forum on,* 565-570.

Córdova, P. (2015). Analysis of real time stream processing systems considering latency. *University of Toronto Patricio@ Cs.Toronto.Edu,*

Crane, M., & Lin, J. (2017). (2017). An exploration of serverless architectures for information retrieval. Paper presented at the *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval,* 241-244.

Creswell, J. W., & Miller, D. L. (2000). Determining validity in qualitative inquiry. *Theory into Practice, 39*(3), 124-130.

de Assuncao, M. D., da Silva Veith, A., & Buyya, R. (2018). Distributed data stream processing and edge computing: A survey on resource elasticity and future directions. *Journal of Network and Computer Applications, 103*, 1-17.

Dias de Assunção, M., da Silva Veith, A., & Buyya, R. (2018). *Distributed data stream processing and edge computing: A survey on resource elasticity and future directions* doi:https://doi.org/10.1016/j.jnca.2017.12.001

DiCicco-Bloom, B., & Crabtree, B. F. (2006). The qualitative research interview. *Medical Education, 40*(4), 314-321.

Dindar, N., Tatbul, N., Miller, R. J., Haas, L. M., & Botan, I. (2013). Modeling the execution semantics of stream processing engines with SECRET. *The VLDB Journal, 22*(4), 421-446.

Ediger, D., Jiang, K., Riedy, J., & Bader, D. A. (2010). (2010). Massive streaming data analytics: A case study with clustering coefficients. Paper presented at the *Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW), 2010 IEEE International Symposium on,* 1-8.

EDUBCA. (2018). 5 most important difference between apache kafka vs flume. Retrieved from https://www.educba.com/apache-kafka-vs-flume/

Estrada, R., & Ruiz, I. (2016). *Big data SMACK: A guide to apache spark, mesos, akka, cassandra, and kafka* Apress.

Graneheim, U. H., & Lundman, B. (2004). Qualitative content analysis in nursing research: Concepts, procedures and measures to achieve trustworthiness. *Nurse Education Today, 24*(2), 105-112. doi:10.1016/j.nedt.2003.10.001 [doi]

Hashem, I. A. T., Yaqoob, I., Anuar, N. B., Mokhtar, S., Gani, A., & Khan, S. U. (2015). The rise of "big data" on cloud computing: Review and open research issues. *Information Systems, 47*, 98-115.

Hey, T., Tansley, S., & Tolle, K. M. (2009). *The fourth paradigm: Data-intensive scientific discovery* Microsoft research Redmond, WA.

Hoffman, S. (2015). *Apache flume: Distributed log collection for hadoop* Packt Publishing Ltd.

Hu, Y. C., Patel, M., Sabella, D., Sprecher, N., & Young, V. (2015). Mobile edge computing—A key technology towards 5G. *ETSI White Paper, 11*(11), 1-16.

IBM Analytics. (2016). Top industry use cases for streaming analytics. Retrieved from https://www.slideshare.net/IBMBDA/top-industry-use-cases-for-streaming-analytics

Janakiram, M. (2016). All apache streaming projects: An exploratory guide. Retrieved from https://thenewstack.io/apache-streaming-projects-exploratory-guide/

Krempl, G., Žliobaite, I., Brzeziński, D., Hüllermeier, E., Last, M., Lemaire, V., . . . Spiliopoulou, M. (2014). Open challenges for data stream mining research. *ACM SIGKDD Explorations Newsletter, 16*(1), 1-10.

Kreps, J., Narkhede, N., & Rao, J. (2011). (2011). Kafka: A distributed messaging system for log processing. Paper presented at the *Proceedings of the NetDB,* 1-7.

Kreps, J., Narkhede, N., & Rao, J. (2011). (2011). Kafka: A distributed messaging system for log processing. Paper presented at the *Proceedings of the NetDB,* 1-7.

Kune, R., Konugurthi, P. K., Agarwal, A., Chillarige, R. R., & Buyya, R. (2016). The anatomy of big data computing. *Software: Practice and Experience, 46*(1), 79-105.

Kurve, A., Griffin, C., Miller, D. J., & Kesidis, G. (2015). Optimizing cluster formation in super-peer networks via local incentive design. *Peer-to-Peer Networking and Applications, 8*(1), 1-21.

Lakhe, B., & Lakhe. (2016). *Practical hadoop migration* Springer.

Marcu, O., Costan, A., Antoniu, G., & Pérez-Hernández, M. S. (2016). (2016). Spark versus flink: Understanding performance in big data analytics frameworks. Paper presented at the *Cluster Computing (CLUSTER), 2016 IEEE International Conference on,* 433-442.

Maxwell, J. (1992). Understanding and validity in qualitative research. *Harvard Educational Review, 62*(3), 279-301.

McAfee, A., Brynjolfsson, E., Davenport, T. H., Patil, D., & Barton, D. (2012). Big data: The management revolution. *Harvard Business Review, 90*(10), 60-68.

Meng, X., Bradley, J., Yavuz, B., Sparks, E., Venkataraman, S., Liu, D., . . . Owen, S. (2016). Mllib: Machine learning in apache spark. *The Journal of Machine Learning Research, 17*(1), 1235-1241.

Nair, K., Kulkarni, J., Warde, M., Dave, Z., Rawalgaonkar, V., Gore, G., & Joshi, J. (2015). (2015). Optimizing power consumption in iot based wireless sensor networks using bluetooth low energy. Paper presented at the *Green Computing and Internet of Things (ICGCIoT), 2015 International Conference on,* 589-593.

Namiot, D. (2015). On big data stream processing. *International Journal of Open Information Technologies, 3*(8)

Navimipour, N. J., & Milani, F. S. (2015). A comprehensive study of the resource discovery techniques in peer-to-peer networks. *Peer-to-Peer Networking and Applications, 8*(3), 474-492.

Patton, M. Q. (1999). Enhancing the quality and credibility of qualitative analysis. *Health Services Research, 34*(5 Pt 2), 1189-1208.

Perera, S., & Suhothayan, S. (2015). (2015). Solution patterns for realtime streaming analytics. Paper presented at the *Proceedings of the 9th ACM International Conference on Distributed Event-Based Systems,* 247-255.

Qian, Z., He, Y., Su, C., Wu, Z., Zhu, H., Zhang, T., . . . Zhang, Z. (2013). (2013). Timestream: Reliable stream computation in the cloud. Paper presented at the *Proceedings of the 8th ACM European Conference on Computer Systems,* 1-14.

Ranjan, R. (2014). Streaming big data processing in datacenter clouds. *IEEE Cloud Computing, 1*(1), 78-83.

Renart, E. G., Diaz-Montes, J., & Parashar, M. (2017). (2017). Data-driven stream processing at the edge. Paper presented at the *Fog and Edge Computing (ICFEC), 2017 IEEE 1st International Conference on,* 31-40.

Sajjad, H. P., Danniswara, K., Al-Shishtawy, A., & Vlassov, V. (2016). (2016). SpanEdge: Towards unifying stream processing over central and near-the-edge data centers. Paper presented at the *Edge Computing (SEC), IEEE/ACM Symposium on,* 168-178.

Satyanarayanan, M. (2017). The emergence of edge computing. *Computer, 50*(1), 30-39.

Schollmeier, R. (2001). (2001). A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications. Paper presented at the *Peer-to-Peer Computing, 2001. Proceedings. First International Conference on,* 101-102.

Shao, G., Berman, F., & Wolski, R. (2000). (2000). Master/slave computing on the grid. Paper presented at the *Heterogeneous Computing Workshop, 2000.(HCW 2000) Proceedings. 9th,* 3-16.

Sharma, S. K., & Wang, X. (2017). Live data analytics with collaborative edge and cloud processing in wireless IoT networks. *IEEE Access, 5*, 4621-4635.

Shoro, A. G., & Soomro, T. R. (2015). Big data analysis: Apache spark perspective. *Global Journal of Computer Science and Technology,*

Singh, D., & Reddy, C. K. (2015). A survey on platforms for big data analytics. *Journal of Big Data, 2*(1), 8.

Singh, M. P., Hoque, M. A., & Tarkoma, S. (2016). Analysis of systems to process massive data stream. *Corr, abs/1605.09021,*

Singh, M. P., Hoque, M. A., & Tarkoma, S. (2016). A survey of systems for massive stream analytics. *ArXiv Preprint arXiv:1605.09021,*

Spark streaming programming guide. (2018). Retrieved from https://spark.apache.org/docs/latest/streaming-programming-guide.html

Tene, O., & Polonetsky, J. (2012). Big data for all: Privacy and user control in the age of analytics. *Nw.J.Tech.& Intell.Prop., 11*, xxvii.

van der Veen, Jan Sipke, van der Waaij, B., Lazovik, E., Wijbrandi, W., & Meijer, R. J. (2015). (2015). Dynamically scaling apache storm for the analysis of streaming data. Paper presented at the *Big Data Computing Service and Applications (BigDataService), 2015 IEEE First International Conference on,* 154-161.

Whiting, L. S. (2008). Semi-structured interviews: Guidance for novice researchers. *Nursing Standard (through 2013), 22*(23), 35.

Wingerath, W., Gessert, F., Friedrich, S., & Ritter, N. (2016). Real-time stream processing for big data. *It-Information Technology, 58*(4), 186-194.

Yadranjiaghdam, B., Pool, N., & Tabrizi, N. (2016). (2016). A survey on real-time big data analytics: Applications and tools. Paper presented at the *Computational Science and Computational Intelligence (CSCI), 2016 International Conference on,* 404-409.

Zaharia, M. (2016). *An architecture for fast and general data processing on large clusters* Morgan & Claypool.

Zaharia, M., Chowdhury, M., Das, T., Dave, A., Ma, J., Mccauley, M., . . . Stoica, I. (2012). Fast and interactive analytics over hadoop data with spark. *Usenix Login, 37*(4), 45-51.

Zaharia, M., Chowdhury, M., Franklin, M. J., Shenker, S., & Stoica, I. (2010). Spark: Cluster computing with working sets. *Hotcloud, 10*(10-10), 95.

Zhang, H., Chen, G., Ooi, B. C., Tan, K., & Zhang, M. (2015). In-memory big data management and processing: A survey. *IEEE Transactions on Knowledge and Data Engineering, 27*(7), 1920-1948.

Zhang, Q., Zhang, X., Zhang, Q., Shi, W., & Zhong, H. (2016). (2016). Firework: Big data sharing and processing in collaborative edge environment. Paper presented at the *Hot Topics in Web Systems and Technologies (HotWeb), 2016 Fourth IEEE Workshop on,* 20-25.