ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)

International Conference on Recent Trends in Computing and Communication (ICRTCC 2015)

Cochin College of Engineering & Technology

Vol. 4, Special Issue 1, June 2015

Big Data: Hadoop Cluster Deployment on ARM Architecture

Vijaykumar S[1], Dr. M. Balamurugan[2], Ranjani K[3]

Research Scholar, School of Computer Science, Engineering and Applications, Bharathidasan University, Trichy, TamilNadu, India[1]

Associate Professor, School of Computer Science, Engineering and Applications, Bharathidasan University, Trichy, TamilNadu, India[2]

Member technical staff, 6th sense, an advanced research and scientific experiment foundation, Kumbakonam, Tamilnadu, India[3]

Abstract: This Research work is the initiation of provisioning low power intensity, clustered architecture to ensure the calibre Big Data Analytics Framework. As a part of work, we made a trial on ARM Architecture that prolonging with compact single board computer called Raspberry-pi which had extensive supporting architecture towards embedded systems. Hence we initiate this proposal to bring an approach for Fault tolerance and reliable process with the help of Hadoop framework. Primitive we delivers a successful implementation of single node cluster on ARM. As a result, we can make high level data management provisioning system with low cost value.

Keywords: Include at least 4 keywords or phrases.

I. INTRODUCTION

Many companies have different definitions of Big Data [5]. In this modern era, the data extends in the numerous dimensions based on the Business model and requirement, as a result origination of V's in defining Bigdata. So far most commonly projected and demanded V's are Volume, velocity, Variety, Veracity, Value, etc. As per Accenture Analytic Survey, More than 1000 respondents from companies operating across 7 industries and headquarter in 19 countries that had completed at least one big data implementation Companies that had not completed at least one big data installation were not included in the results [5]. 92% of users are fully satisfied with their business outcomes. 94% of users reports that their implementation meeting their needs. 89% of users believe that the big data will revolutionize the same way as the internet did. The vast majority 92% of all users report that they are satisfied with business outcomes and 94% feel their big data implementation meets their need. Based on the needs and demands of this civic need we initiate this research to provide better architecture and framework to with low power multi Block infrastructure. For analysing Bigdata from multiple Dimension.

A. ARM

ARM is also an instruction set architectures used by processors depend on RISC architecture. Represent three cortex profiling for Application, Real-time, Microcontroller known as Cortex A, Cortex R, and Cortex M. Mainly Raspberry pi comes with ARM1176JZ-F undefined series but most of the properties are as same as ARM 11. Which is 32 bit ARM architecture, ARMv6 Architecture core. Especially those architecture are emit reduced heat when compare with previous models and lower heat risk and most compactable for real time process. Because most of the mobile phones are using this architecture. Series 1176 especially having security extensions. [9]

B. Hadoop

Hadoop is a platform that provides both distributed storage and computational capabilities. It brings support in two dimensions viz., HDFS for storage and map reduce for computational capabilities [7].

C. MAP REDUCE

MapReduce is a Programming model and an associated implementation for processing and generating large data sets [4]. Users specify the computation in terms of a map and a reduce function, and the underlying runtime system automatically parallelizes the computation across large-scale clusters of machines, handles machine failures, and schedules inter-machine communication to make efficient use of the network and disks. Programmers find the system easy to use: more than ten thousand distinct MapReduce programs have been implemented internally at Google over the past four years, and an average of one hundred thousand MapReduce jobs are executed on Google's clusters every day, processing a total of more than twenty petabytes of data per day[6][4].

II. PROBLEM STATEMENT

Identify the light weighted and low power consuming Architecture that supports Hadoop installation and it should meet basic hardware requirements. Which should bring high availability, durability, fault tolerance, supports variety and high data loads. Whether the chosen Architecture supports clustering and utilize maximum resources to contribute in improving the system

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)

International Conference on Recent Trends in Computing and Communication (ICRTCC 2015)

Cochin College of Engineering & Technology

Vol. 4, Special Issue 1, June 2015

performance. The minimum amount of instructions set that can be handled by the single board computer. Identify the source of power and power consumption with heat resistance. Then the challenges towards compatible software and its supporting versions specially designed for installing in the Architecture (JVM, Secure shell, HDFS, Job Tracker, Task tracker) and commands to install Hadoop on the single board computer.

III. RASPBERRY – PI

Raspberry Pi Foundation is an educational charity situated in UK which have a motto in finding of advance education system and technology for society. As their contribution they developed Credit card size light weight processing computer called Raspberry Pi [11]. And its technical information is mentioned in the below table 1.

TABLE I: TECHNICAL INFORMATION

Chip	Broadcom BCM2835 SoC full HD multimedia applications processor
CPU	700 MHz Low Power ARM1176JZ-F Applications Processor
GPU	Dual Core VideoCore IV® Multimedia Co-Processor
Memory	512MB SDRAM
Ethernet	On board 10/100 Ethernet RJ45 jack
USB 2.0	Dual USB Connector
Video Output	HDMI (rev 1.3 & 1.4) Composite RCA (PAL and NTSC)
Audio Output	3.5mm jack, HDMI
On-board Storage	SD, MMC, SDIO card slot

IV. IMPLEMENTATION

Fig. 1. Implemented ARM Architecture

A. POWER SOURCE

There are two possible way to provide power source for our system. In this architecture we have chosen micro USB instead of GIPO for achieving quick stability based on available resource having capability for providing power to I/O components. 2A - 5 V is the power factor meets our requirement.

B. OPERATING SYSTEM

Code named wheezy is the one of the stable version from Debian, Linux distribution. With the future of multi-arch which support 32 bit runs on 64 bit Operating system and its feature extends to support arm [8]. So here in this work we choose it as one of the supporting system for Hadoop in ARM architecture. Therefore, we utilized Rasbian, Debian wheezy Linux operating system Kernel Version 3.12, and Released on 9 September 2014 from the Raspberry supporting site

C. JAVA

We need JVM for pi. Because Hadoop framework core depends on Java. So the following commands is lead to java installation from its source.

```
pi@raspberrypi ~ $ sudo apt-get install openjdk-7-jdk
pi@raspberrypi ~ $ java -version
java version "1.7.0_07"
OpenJDK Runtime Environment (IcedTea7 2.3.2) (7u7-2.3.2a-1+rpi1)
OpenJDK Zero VM (build 22.0-b10, mixed mode)
```

D. Hduser

Later we creates new user for hadoop for avoiding file system collision. It's also sudo user having rights to install applications and that user will later added into hadoop group to access it file system.

```
pi@raspberrypi ~ $ sudo addgroup hadoop
pi@raspberrypi ~ $ sudo adduser --ingroup hadoop hduser
pi@raspberrypi ~ $ sudo adduser hduser sudo
```

E. Hadoop Installation

```
hduser@raspberrypi ~ $ hadoop version
Hadoop 1.1.2
Subversion
https://svn.apache.org/repos/asf/hadoop/common/branches/branch-1.1 -r 1440782
```

F. SSH CONFIGURATION

SSH are secure shell widely using protocol to connecting system remotely. After Creating SSH key share that key with user to establish communication between its nodes. Then we start SSH Localhost.

```
hduser@raspberrypi ~ $ ssh localhost
Linux raspberrypi 3.12.28+ #709 PREEMPT Mon Sep 8 15:28:00 BST 2014 armv6l
```

G. HDFS CREATION & RELATED PROCESS

Those following commands are executed in the manner to create directory. After formatting that directory with HDFS file system it became Hadoop specialized distributed file system, which is apart from the Linux file system, for data protection concern we are creating new user to avoid some collision with regular Linux file system.

```
hduser@raspberrypi ~ $ sudo mkdir -p /fs/hadoop/tmp
```

This command brings ownership permission to hduser and its group for doing process on mentioned directory

```
hduser@raspberrypi ~ $ sudo chown hduser:hadoop /fs/hadoop/tmp
```

This step brings privilege to the user such that 750 it is the common type of permission where users can possibly process, read, write and execute (Traverse for directories).Also it limits the group users for doing the operations only read, execute and denies write operation.

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)

International Conference on Recent Trends in Computing and Communication (ICRTCC 2015)

Cochin College of Engineering & Technology

Vol. 4, Special Issue 1, June 2015

Extend to that it avoid data writing violations from other intrusions.

```
hduser@raspberrypi ~ $ sudo chmod 750 /fs/hadoop/tmp
hduser@raspberrypi ~ $ hadoop namenode –format
```

hadoop namenode -formate this command formating your file system at the location specified in hdfs-site.xml for example: hear my name node directory is /usr/local/hadoop/dfs/name

```
<property>
<name>dfs.name.dir</name>
<value>/usr/local/hadoop/dfs/name</value>
<final>true</final>
</property>
```

H. PROCESSES INVOCATION

After installing the required component in Linux, most commonly we need to start the process manually. Here the start-all.sh starts the required components of Hadoop such as name node, data node, secondary name node, job tracker and task tracker.

```
hduser@raspberrypi ~ $ start-all.sh
starting namenode, logging to
/usr/local/hadoop/libexec/../logs/hadoop-hduser-
namenode-raspberrypi.out
localhost: starting datanode, logging to
/usr/local/hadoop/libexec/../logs/hadoop-hduser-datanode-
raspberrypi.out
localhost: starting secondarynamenode, logging to
/usr/local/hadoop/libexec/../logs/hadoop-hduser-
secondarynamenode-raspberrypi.out
starting jobtracker, logging to
/usr/local/hadoop/libexec/../logs/hadoop-hduser-
jobtracker-raspberrypi.out
localhost: starting tasktracker, logging to
/usr/local/hadoop/libexec/../logs/hadoop-hduser-
tasktracker-raspberrypi.out
```

I. JPS

The jps tool lists the instrumented HotSpot Java Virtual Machines (JVMs) on the target system. The tool is limited to reporting information on JVMs for which it has the access permissions [10]. The numeric value represented before the instrumented JVM are its identification number. Listing the instrumented JVMs on the local host:

```
hduser@raspberrypi ~ $ jps
3051 Jps
2612 NameNode
2816 SecondaryNameNode
2710 DataNode
2999 TaskTracker
2892 JobTracker
```

V. RESULT

```
hduser@raspberrypi /usr/local/hadoop $ hadoop jar
hadoop-examples-1.1.2.jar pi 5 50
Number of Maps  = 5
Samples per Map = 50
Wrote input for Map #0
Wrote input for Map #1
Wrote input for Map #2
Wrote input for Map #3
Wrote input for Map #4
Starting Job
14/12/13 11:48:09 INFO mapred.FileInputFormat: Total
input paths to process : 5
14/12/13 11:48:21 INFO mapred.JobClient: Running job:
job_201412131131_0001
14/12/13 11:48:22 INFO mapred.JobClient: map 0%
reduce 0%
14/12/13 11:52:33 INFO mapred.JobClient: map 20%
reduce 0%
14/12/13 11:55:36 INFO mapred.JobClient: map 40%
reduce 0%
14/12/13 11:55:49 INFO mapred.JobClient: map 40%
reduce 13%
14/12/13 11:57:43 INFO mapred.JobClient: map 60%
reduce 13%
14/12/13 11:57:55 INFO mapred.JobClient:  map 60%
reduce 20%
14/12/13 11:58:04 INFO mapred.JobClient:  map 80%
reduce 20%
14/12/13 11:58:18 INFO mapred.JobClient:  map 80%
reduce 26%
14/12/13 11:59:15 INFO mapred.JobClient:  map 100%
reduce 26%
14/12/13 11:59:27 INFO mapred.JobClient:  map 100%
reduce 33%
14/12/13 11:59:46 INFO mapred.JobClient:  map 100%
reduce 100%
14/12/13 12:00:40 INFO mapred.JobClient: Job complete:
job_201412131131_0001
14/12/13 12:00:41 INFO mapred.JobClient: Counters: 30
14/12/13 12:00:42 INFO mapred.JobClient:   Job Counters
14/12/13 12:00:42 INFO mapred.JobClient: Launched
reduce tasks=1
14/12/13 12:00:42 INFO mapred.JobClient:
SLOTS_MILLIS_MAPS=976959
14/12/13 12:00:42 INFO mapred.JobClient: Total time
spent by all reduces waiting after reserving slots (ms)=0
14/12/13 12:00:42 INFO mapred.JobClient: Total time
spent by all maps waiting after reserving slots (ms)=0
14/12/13 12:00:42 INFO mapred.JobClient: Launched
map tasks=6
14/12/13 12:00:42 INFO mapred.JobClient: Data-local
map tasks=6
14/12/13 12:00:42 INFO mapred.JobClient:
SLOTS_MILLIS_REDUCES=421500
14/12/13 12:00:42 INFO mapred.JobClient:  File Input
Format Counters
14/12/13 12:00:42 INFO mapred.JobClient: Bytes
Read=590
14/12/13 12:00:42 INFO mapred.JobClient: File Output
Format Counters
14/12/13 12:00:42 INFO mapred.JobClient: Bytes
Written=97
14/12/13 12:00:42 INFO mapred.JobClient:
FileSystemCounters
14/12/13 12:00:42 INFO mapred.JobClient:
FILE_BYTES_READ=116
```

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)

International Conference on Recent Trends in Computing and Communication (ICRTCC 2015)

Cochin College of Engineering & Technology

Vol. 4, Special Issue 1, June 2015

14/12/13 12:00:42 INFO mapred.JobClient: HDFS_BYTES_READ=1210
14/12/13 12:00:42 INFO mapred.JobClient: FILE_BYTES_WRITTEN=305403
14/12/13 12:00:42 INFO mapred.JobClient: HDFS_BYTES_WRITTEN=215
14/12/13 12:00:42 INFO mapred.JobClient: Map-Reduce Framework
14/12/13 12:00:42 INFO mapred.JobClient: Map output materialized bytes=140
14/12/13 12:00:42 INFO mapred.JobClient: Map input records=5
14/12/13 12:00:42 INFO mapred.JobClient: Reduce shuffle bytes=140
14/12/13 12:00:42 INFO mapred.JobClient: Spilled Records=20
14/12/13 12:00:42 INFO mapred.JobClient: Map output bytes=90
14/12/13 12:00:42 INFO mapred.JobClient: Total committed heap usage (bytes)=1021792256
14/12/13 12:00:42 INFO mapred.JobClient: CPU time spent (ms)=73380
14/12/13 12:00:42 INFO mapred.JobClient: Map input bytes=120
14/12/13 12:00:42 INFO mapred.JobClient: SPLIT_RAW_BYTES=620
14/12/13 12:00:42 INFO mapred.JobClient: Combine input records=0
14/12/13 12:00:42 INFO mapred.JobClient: Reduce input records=10
14/12/13 12:00:42 INFO mapred.JobClient: Reduce input groups=10
14/12/13 12:00:42 INFO mapred.JobClient: Combine output records=0
14/12/13 12:00:42 INFO mapred.JobClient: Physical memory (bytes) snapshot=726032384
14/12/13 12:00:42 INFO mapred.JobClient: Reduce output records=0
14/12/13 12:00:42 INFO mapred.JobClient: Virtual memory (bytes) snapshot=2161262592
14/12/13 12:00:42 INFO mapred.JobClient: Map output records=10
Job Finished in 758.142 seconds
Estimated value of Pi is 3.14800000000000000000

VI. DISCUSSION

This proposed system is basically now installed in the heavy weighted operating system. It decreases the performance of the system. Next our team planned to implement this architecture in the lightweight operating system which is also compatible to run hadoop framework. And the java version used in this implementation is the common java platform that is compatible for Linux system architecture. So need to introduce byte code compiler to increase the performance of this system. This system currently having infrastructure to equip minimum amount of data volume to process in a parallel manner. But this is the major initiation towards multi node Lightweight compact distributed architecture for various demand based Big data process.

VII. CONCLUSION

The Result proven that the deployment of single node cluster on ARM Architecture successfully executed Pi task in Single board compact portable computer Raspberry-pi. And this system is constructed with in the cost less than 3000 INR. Equivalent to 48$ approximately. This covers the primary unit of the single node architecture excluding the I/O & Displays.

VIII. FUTURE WORK

As a continuity of this work, we have planned to introduce multi node cluster on latest ARM Architecture. With Lightweight high performance and capable to load and distribute data on cost effective model. Introducing Metric for analysing performance and stability of Bigdata analytics in ARM Architecture.

ACKNOWLEDGMENT

Authors **Vijaykumar S.** and **Ranjani K.** are Grateful to Research, Technical and Advisor members of "6TH SENSE" Research Foundation, Kumbakonam, Tamilnadu, South India.

REFERENCES

[1] VijayKumar S, Saravanakumar S.G., Revealing of NOSQL Secrets. CiiT Journal.vol2,no10 (Oct.2010), 310314. URL=http://www.ciitresearch.org/dmkeoctober2010.html
[2] VijayKumar S, Saravanakumar S.G., "Implementation of NOSQL for robotics", Publisher: IEEE (Dec.2010). DOI=10.1109/INTERACT.2010.5706225
[3] Vijay Kumar S, Saravanakumar S.G., Future Robotics Memory ManagementFuture Robotics Memory Management, Publisher : Springer Berlin Heidelberg.Year 2011. DOI=10.1007/978-3-642-24055-3_32
[4] Jeffrey Dean and Sanjay Ghemawat , "MapReduce: Simplifed Data Processing on Large Clusters", OSDI 2004.
[5] Accenture, "Big Success with big data" April 2014 Available at URL : http://www.accenture.com/us-en/Pages/insight-big-success-big-data.aspx
[6] Jeffrey Dean and Sanjay Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters". Communications Of The Acm January 2008/Vol. 51, No. 1.
[7] Alex Holmes, "Hadoop in Practice'. Publisher: Manning, Shelter Island. Year: 2012. ISBN: 9781617290237
[8]]Debian Operating system . URL : https://www.debian.org/releases/stable/
[9] ARM Architecture. URL : http://arm.com/
[10] Java, JVM, JSP. URL:https://docs.oracle.com/javase/7/docs/technotes/tools/share/jps.html
[11] Raspberry PI Foundation. URL: http://www.raspberrypi.org/

BIOGRAPHY

Dr. M. Balamurugan pursued his undergraduate and postgraduate degrees in Computer Science from Bharathidasan University, Tiruchirappalli and he completed his Master of Philosophy Degree in Computer Science from MS University, Tirunelveli. He was awarded doctoral Degree Ph.D from Bharathidasan University, Tiruchirappalli. He has presented research papers in 20 international and national conferences. He has published 10 research papers in national and international journals. Now, he is Associate Professor at Bharathidasan University , Tiruchirappalli (India). His research interests are mainly focused on the area of Data Science and Big Data. He has supervised several research scholars in these areas.