

Effects of the EPNs and FLPs on the Information
Node for ALICE
Cern

M.Q Puls

2018

Title	Effects of the EPNs and FLPs on the Information Node for ALICE
Name	Mitchell Quinn Puls
Student Number	500659986
Phone	+31615050310
Email	mitchpuls@upcmail.nl mitch.puls@hva.nl
Place	Amsterdam
Date	2018
University	University of Applied Sciences Amsterdam
Department	TI
Mentor	C. J. Rijsenbrij
Company	University of Amsterdam Software for Science Wibautstraat 2-4 Amsterdam 0205995555
Company Supervisor	Dr. Marten Teitsma
Period	Feb. 2018 - Jul 2018

Preface

Contents

1	Introduction	6
1.1	ALICE	6
1.2	Load Balancing	6
1.3	Research	7
1.4	Research Model	7
2	Framework	8
2.1	O^2 Balancer	8
2.1.1	Devices	8
2.2	Raspberry Pi	9
2.3	FairMQ	9
2.4	Zookeeper	9
2.5	Fail-over	9
2.6	Ansible	10
2.7	Blacklist Algorithm	10
3	Main Research Question	11
4	Experiments	12
4.1	Hardware	12
4.2	Software	14
4.3	Data analysis tools	14
4.4	Experiments	15
4.4.1	Experiment one	15
4.4.2	Experiment two	15
4.4.3	Experiment three	16
4.4.4	Experiment four	16
5	Results	17
5.1	Ex.1 2 FLPs 12 EPNs	17
5.2	Ex.2 2 FLPs 12 EPNs	17
5.3	Ex.3 2 FLPs 12 EPNs	17
5.4	Ex.1 3 FLPs 18 EPNs	17
5.5	Ex.2 3 FLPs 18 EPNs	18

5.6	Ex.3 3 FLPs 18 EPNs	18
5.7	Ex.1 4 FLPs 24 EPNs	18
5.8	Ex.2 4 FLPs 24 EPNs	18
5.9	Ex.3 4 FLPs 24 EPNs	18
5.10	Ex.4 4 FLPs 24 EPNs	18
6	Conclusion	19
6.1	Appendix	19
6.2	Planning as of 25/04/2018	19

Summary

Chapter 1

Introduction

This research is conducted as part of the final thesis of Mitchell Quinn Puls, Technical Computing student at the Amsterdam University of Applied Sciences. This research is about the effects of a high amount of computers, processing a vast amount of data. This research is in assignment from CERN in Switzerland.

1.1 ALICE

ALICE stands for A Large Ion Collider Experiment, and is a detector mounted on the Large Hadron Collider at CERN. CERN is a European organization for nuclear research, situated in Geneva Switzerland. ALICE's main function is to study matter at extreme energy densities, where matter turns into a form called quark-gluon plasma. Everything in the universe is made from protons, neutrons (except hydrogen which does not have any neutrons) and electrons. Protons and neutrons are then build up with quarks and bound together with something called a gluon. Quark-gluon plasma matter that appeared at the very start of the big bang, and is the thing that binds the gluons and quarks together. CERN wants to observe this matter. The way they achieve this, is by shooting two lead ions against each other. This produces a heat that is over 100,000 times hotter than the center of the Sun. This breaks the bounds between the quarks and the gluons and makes the quark-gluon plasma visible.

1.2 Load Balancing

The data stream that comes from ALICE is equal to about 1.1 Terabyte per second. All of this data comes in what is known as a heartbeat. This heartbeat gets distributed over 268 First Level Processors and funneled through 1500 Event Processing Nodes. The efficient distribution of this process is what is known as Load Balancing. All of these computers would be monitored by an Information Node.

1.3 Research

This research is a continuation of a previous research done by Heiko van der Heijden. His results show that of the two algorithms tested, Re-initialization and Blacklist, that the Blacklist algorithm has fewer Time Frames lost.

Even though the same ratio of FLPs to EPNs that is situated at CERN was used ($1/6$), there were fewer computers used than at CERN. Because of this it is not sure whether or not the Information Node is able to handle 1700+ computers as compared to the 15 computers used in the experiment. This research is focused on the capability of the Information Node to monitor a higher number of FLPs and EPNs and what the effects are on the results compared to the previous experiment.

1.4 Research Model

Contrary to the previous research, which used a cluster of computers situated at Nikhef Amsterdam, this research will be conducted using a cluster of Raspberry Pi's. This decision has been made because of an unfortunate failure of communication from Nikhef about the availability of the cluster for this research period. The first step is to recreate the previous experiment which was focused around the various ticktimes of Zookeeper. This research is done to both validate the results from the previous experiment, but is to also validate the new Raspberry Pi cluster to confirm that this is a valid way to conduct this experiment. After recreating the first experiment, the same experiment will be conducted with a higher numbers of computers to see what the results are.

All technical documentation of what everything is will be explained in the next section including the definition of the experiments. The next chapter will explore more in depth of the prototype made for the experiment and it's difficulties that came with it. The following chapter looks at the results from the executed experiments. After this follows an analysis of the results of the experiment. Finally a conclusion and recommendations.

Chapter 2

Framework

This chapter will go more in depth about the several different kinds of hardware and software used. It will also more elaborate the different terms and names used for aspects of the research.

2.1 O^2 Balancer

O^2 Balancer is a framework used by CERN for simulation experiments for ALICE. The code is open source and licensed under the GNU General Public License V3.0.

2.1.1 Devices

The O^2 Balancer consist of a cluster of 1750 computers, divided in 250 First Level Processors (FLPs) and 1500 Event Nodes (EPNs) These computers are meant to process the data stream coming from ALICE. All of these computers are monitored using an Information Node.

First Level Processors

The FLPs are the first computers in the line. They receive the data stream (approximately 1.1TB/s) from ALICE and need to distribute that to the next line of computer. In order to do that it takes the data received between two heartbeats, and compresses that into something that's called a Sub Timeframe (STF). A heartbeat lasts for about 20ms. It will then send this STF to the next line of computers which are the EPNs. Every EPN needs to get the same amount of STFs at the same time for recreation purposes. These STFs can then be further examined from there.

Event Processing Nodes

The next line of computers are the EPNs. These receive the STFs from the FLPs and then compress them back into a time frame (TF). This compression reduces it's size by a factor of eight. These TFs are then stored for further use.

Information Node

There is one final computer which is the Information Node (IN). This computer keeps track of all the FLPs and EPNs that are online and makes sure that FLPs don't send data to offline EPNs.

2.2 Raspberry Pi

Raspberry Pi is a low cost small computer used for prototyping projects. These projects can reach go from small sensor applications, to bigger host-server applications. The Raspberry pi used for this research is the model 3 B+ variant. This is the latest version released, and is used for it's higher Ethernet connection with the onboard Ethernet connection.

2.3 FairMQ

The transport layer used for the O^2 Balancer is FairMQ. This is a transport layer from the larger framework FairRoot created by GSI Darmstadt. In order to accommodate the smaller processing size of the Raspberry Pi, a trimmed down version of FairRoot is used which is just FairMQ. This is a data transport layer used to send data in between the IN, FLPs and EPNs.

2.4 Zookeeper

Zookeeper is a program made by Apache to regulate the whole load balancing process. It is run on the Information Node and from there pings to all EPNs to check whether they are online or not. It then creates a list of online EPNs which it gives to the FLPs so that they know to what EPN to send data to. The frequency of these pings are called the Ticktime.

2.5 Fail-over

When an EPN goes offline it is called a Fail-over. When this happens, Zookeeper will know that it is offline and will notify the FLPs to not send any data to these EPNs anymore.

2.6 Ansible

Ansible is a deployment software used to create simple automation for large infrastructures. This is used to automate repetitive task for the experiment, and for deploying software stacks to every unit.

2.7 Blacklist Algorithm

The algorithm used in the previous experiment is a Blacklist Algorithm. This algorithm constantly keeps a list of online channels which is updated by the Information Node using Zookeeper. Once Zookeeper realizes that an EPN is offline, it will update the list so that the algorithm will skip that offline EPN. With this list of online EPNs, the algorithm uses a Round Robin approach to distribute the STF over the EPNs. A way to implement the Blacklist algorithm is shown in listing x

Chapter 3

Main Research Question

What effect does the blacklist algorithm have on the Information Node with an higher amount of First Level Processors and Event Processing Nodes for ALICE?

Chapter 4

Experiments

This chapter goes more in depth of the specifics on the experiments conducted. It describes the exact hardware specifications and software libraries used. It also defines all the different experiments and includes several figures for added information.

4.1 Hardware

The hardware used for this experiment are specially designed clusters made with Raspberry Pi's 3 model B+. These Pi's are assigned using the same ratio of FLPs and EPNs at CERN (1:6) and 1 Information Node. The Pi's are modified with an extra Ethernet port. The exact specifications are in table 5.1:

Processor	Cortex-A53 1.4GHZ
RAM	1GB LPDDR2 SDRAM
Network	1 300MbE 1 10/100MbE
Operating System	Raspbian Stretch (Debian 9)

Table 4.1: Specifications modified Raspberry Pi 3 model B+

Everything is setup and built from a single server unit. The IP addresses are also managed from this unit. Specifications are in table 5.2:

Processor	Intel Xeon
RAM	
Network	
Operating System	Raspbian Stretch (Debian 9)

Table 4.2: Specifications management server

The network configuration consists of a bidirectional connection of :

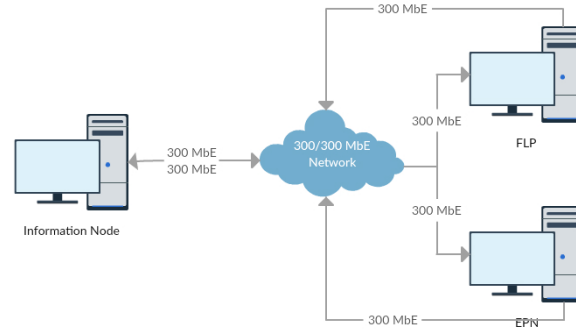


Figure 4.1: Network setup

The clusters of pi's are build using a

4.2 Software

The software used is found at <https://github.com/SoftwareForScience/O2-Balancer>. It consists of 3 executable programs to represent an FLP, EPN and Information Node. These programs are send to the devices to represent their respective units. Apart from that it uses a slightly modified version of FairRoot ¹. Since only the FairMQ ² part is needed for the programs to run it has been split off from this code.

The Infomation Node serves two purposes. At first it generates the TFs that are send to the FLPs using FairMQ. It also receives notifications from the EPNs when they have received the full TF from the FLPs again. The configuration is done using YAML files. A diagram of the connections and dependencies are as followed.

Library/Tool	Version
FairMQ	1.1.5
ZeroMQ	4.2.1
Zookeeper	3.4.9
Cmake	3.11.0
Boost	1.66.0
Yaml-cpp	0.5.2
Compiler	gcc 6.3.0 20170516 (Raspbian 6.3.0-18+rpi1+deb9u1)

Table 4.3: Dependencies software

4.3 Data analysis tools

In order to stay consistent with the previous experiment, the same tools will be used to create and visualize the data. This can be found at <https://github.com/valvy/BalancerScripts>. These scripts are written in Python and ROOT to generate graphs and histograms from the log files received from the software. The dependencies are specified in table 5.4.

¹<https://github.com/FairRootGroup/FairRoot>

²<https://github.com/FairRootGroup/FairMQ>

Tool	Version
Raspbian	Centos 7
ROOT	6.13.02
Python	2.7.13
Ansible	2.2.1

Table 4.4: Dependencies for the analysis scripts

4.4 Experiments

In order to check whether or not the Information Node has any issues with increased numbers of FLPs and EPNs, the same experiments need to be done as described in the previous report (van der Heijden, 2018, p23-p27) but will be briefly summarized again. The experiments are done in two steps. At first the experiments need to be run using the same amount of FLPs and EPNs to verify whether or not the new setup of Pi clusters are able to give the same result. After that the experiments need to be run again using more FLPs and EPNs to check if it indeed does have an effect on the Information Node.

4.4.1 Experiment one

Ticktime influence on the Blacklist algorithm with one fail-over.

The first experiment uses a fixed sample size to be sent from the Information Node, and will have 1 fail-over during its run. This sample size is set to hundred-forty kilobyte and the heartbeat rate is set to twenty milliseconds. This heartbeat is set at the same rate used at CERN. This sample size is set to accommodate the slower Ethernet and processing speed of Raspberry Pi as compared to units used in the previous experiment (van der Heijden, 2018, p.23). This experiment will disable one EPN when it receives the first STF from heartbeat 3.000. Then special scripts will parse the logfiles between heartbeat 2.000 and 10.000 to have enough of a buffer to read from. A flowchart can be found in figure 5.3.

4.4.2 Experiment two

Ticktime influence on the Blacklist algorithm with all but one fail-over.

The same heartbeat rate and sample size are used from experiment one. For this experiment the first EPN will be disabled at heartbeat 2.000. After that every 1.000 heartbeats an additional EPN will be disabled until there is only 1 left. Scripts will parse all the logs to check how many TFs were lost during this progress. A flowchart can be found in figure 5.4.

4.4.3 Experiment three

Ticktime influence on the Blacklist algorithm with all but one fail-over with random sample size.

The same heartbeat rate is used from experiment one. A random sample size will be generated from the FLPs between .. & .. . Apart from that the same configuration will be used as in experiment two. The first EPN will be disabled at heartbeat 2.000 and after that every 1.000 heartbeats an additional EPN will be disabled. A flowchart can be found in figure 5.5.

4.4.4 Experiment four

Ticktime influence on the Blacklist algorithm with all but one fail-over at once.

The same heartbeat rate is used from experiment one. For this experiment all but one EPN will be disabled at heartbeat 3.000. After that the system will run for another 7.000 heartbeats to get enough of a buffer to read from. A flowchart can be found in figure 5.6.

Chapter 5

Results

5.1 Ex.1 2 FLPs 12 EPNs

Ticktime influence on the blacklist algorithm with one fail-over.

5.2 Ex.2 2 FLPs 12 EPNs

Ticktime influence on the blacklist algorithm with all but one fail-over.

5.3 Ex.3 2 FLPs 12 EPNs

Ticktime influence on the blacklist algorithm with all but one-fail over with random sample size.

5.4 Ex.1 3 FLPs 18 EPNs

Ticktime influence on the blacklist algorithm with one fail-over.

5.5 Ex.2 3 FLPs 18 EPNs

Ticktime influence on the blacklist algorithm with all but one fail-over.

5.6 Ex.3 3 FLPs 18 EPNs

Ticktime influence on the blacklist algorithm with all but one-fail over with random sample size.

5.7 Ex.1 4 FLPs 24 EPNs

Ticktime influence on the blacklist algorithm with one fail-over.

5.8 Ex.2 4 FLPs 24 EPNs

Ticktime influence on the blacklist algorithm with all but one fail-over.

5.9 Ex.3 4 FLPs 24 EPNs

Ticktime influence on the blacklist algorithm with all but one fail-over with random sample size.

5.10 Ex.4 4 FLPs 24 EPNs

Ticktime influence on the blacklist algorithm with all but one fail-over at once.

Chapter 6

Conclusion

Bibliography

DIT IS NOG NIET APA WEET IK

<https://home.cern/about/experiments/alice>

<https://home.cern/about>

<https://fairroot.gsi.de/?q=about>

<https://github.com/FairRootGroup/FairRoot>

<https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>

<https://www.ansible.com/overview/it-automation>

6.1 Appendix

6.2 Planning as of 25/04/2018

Name	Description	Deliverable	Est.	Complete
Create scripts for all experiments	Create all the scripts for the deployment of experiments	Experiment scripts	6 hours	19/04/2018
Experiment 1 (2/12)	Ticktime influence on the blacklist algorithm with one fail-over	Compressed Ex.1 2/12 results	105 hours	30/04/2018
Experiment 2 (2/12)	Ticktime influence on the blacklist algorithm with all but one fail-over	Compressed Ex.2 2/12 results	19 hours	01/05/2018

Work out Ex.1 (2/12)	Compile results of Ex.1 (2/12)	Compiled re- sults of Ex.1 (2/12)	2 hours	30/04/2018
Write chapter 6.1	Write down the re- sults for chapter 6.1 about the results of Ex.1 (2/12)	Chapter 6.1	3 hours	30/04/2018
Experiment 3 (2/12)	Ticktime influence on the blacklist algorithm with all but one-fail over with random sam- ple size	Compressed Ex.3 2/12 results	19 hours	02/05/2018
Work out Ex.2 (2/12)	Compile results of Ex.2 (2/12)	Compiled re- sults of Ex.2 (2/12)	2 hours	01/05/2018
Write chapter 6.2	Write down the re- sults for chapter 6.2 about the results of Ex.2 (2/12)	Chapter 6.2	3 hours	01/05/2018
Experiment 2 (3/18)	Ticktime influence on the blacklist algorithm with all but one fail- over	Compressed Ex.2 3/18 results	19 hours	03/05/2018
Work out Ex.3 (2/12)	Compile results of Ex.3 (2/12)	Compiled re- sults of Ex.3 (2/12)	2 hours	02/05/2018
Write chapter 6.3	Write down the re- sults for chapter 6.3 about the results of Ex.3 (2/12)	Chapter 6.3	3 hours	02/05/2018
Experiment 2 (4/24)	Ticktime influence on the blacklist algorithm with all but one fail- over	Compressed Ex.2 4/24 results	19 hours	04/05/2018
Work out Ex.2 (3/18)	Compile results of Ex.2 (3/18)	Compiled re- sults of Ex.2 (3/18)	2 hours	03/05/2018
Write chapter 6.5	Write down the re- sults for chapter 6.5 about the results of Ex.2 (3/18)	Chapter 6.5	3 hours	03/05/2018

Experiment 3 (3/18)	Ticktime influence on the blacklist algorithm with all but one-fail over with random sample size	Compressed Ex.3 3/18 results	19 hours	09/05/2018
Work out Ex.2 (4/24)	Compile results of Ex.2 (4/24)	Compiled results of Ex.2 (4/24)	2 hours	04/05/2018
Write chapter 6.8	Write down the results for chapter 6.8 about the results of Ex.2 (4/24)	Chapter 6.8	3 hours	04/05/2018
Experiment 3 (4/24)	Ticktime influence on the blacklist algorithm with all but one-fail over with random sample size	Compressed Ex.3 4/24 results	19 hours	10/05/2018
Work out Ex.3 (3/18)	Compile results of Ex.3 (3/18)	Compiled results of Ex.3 (3/18)	2 hours	09/05/2018
Write chapter 6.6	Write down the results for chapter 6.6 about the results of Ex.3 (3/18)	Chapter 6.6	3 hours	09/05/2018
Experiment 1 (3/18)	Ticktime influence on the blacklist algorithm with one fail-over	Compressed Ex.1 3/18 results	105 hours	14/05/2018
Work out Ex.3 (4/24)	Compile results of Ex.3 (4/24)	Compiled results of Ex.3 (4/24)	2 hours	11/05/2018
Write chapter 6.9	Write down the results for chapter 6.9 about the results of Ex.3 (4/24)	Chapter 6.9	3 hours	11/05/2018
Experiment 1 (4/24)	Ticktime influence on the blacklist algorithm with one fail-over	Compressed Ex.1 4/24 results	105 hours	19/05/2018
Work out Ex.1 (3/18)	Compile results of Ex.1 (3/18)	Compiled results of Ex.1 (3/18)	2 hours	15/05/2018
Write chapter 6.4	Write down the results for chapter 6.4 about the results of Ex.1 (3/18)	Chapter 6.4	3 hours	15/05/2018

Compile + write appendixes	Compile and write down everything that needs to go in the appendix, which include the bibliography, email documentation, and glossary	Appendixes	4 hours	16/05/2018
Work out Ex.1 (4/24)	Compile results of Ex.1 (4/24)	Compiled results of Ex.1 (4/24)	2 hours	21/05/2018
Write chapter 6.7	Write down the results for chapter 6.7 about the results of Ex.1 (4/24)	Chapter 6.7	3 hours	21/05/2018
Write a summary in 2/3 languages	Write a summary of the thesis in English, Dutch and German	Summaries	16 hours	18/05/2018
Experiment 4 (4/24)	Ticktime influence on the blacklist algorithm with all but one fail-over at once	Compressed Ex.4 4/24 results	19 hours	22/05/2018
Work out Ex.4 (4/24)	Compile results of Ex.4 (4/24)	Compiled results of Ex.4 (4/24)	2 hours	23/05/2018
Write chapter 6.10	Write down the results for chapter 6.10 about the results of Ex.4 (4/24)	Chapter 6.10	3 hours	23/05/2018
Write a preface	Write a small preface	Preface	1 hours	24/05/2018
Write an introduction	Write an introduction for the thesis	Introduction	2 hours	24/05/2018
Write chapter 7.1	Write down the conclusions based on the results of the previous experiments	Chapter 7.1	3 hours	25/05/2018
Write chapter 7.2	Write down the recommendations based on the results of the previous experiments	Chapter 7.2	3 hours	25/05/2018