Cost-Efficiency Comparison of an ARM Cluster & Intel Server

Jaden D. Weiss
High Technology High School
Lincroft, NJ, USA

Abstract - *Data processing requirements can be met by using a small group of complex fast processors or a large number of extremely low-cost and low-power processors. This paper empirically compares the cost-effectiveness of these two computing paradigms and evaluates the hypothesis that the low-cost processor solution is more cost-effective. The control group consists of data obtained from a traditional Intel server (2 Intel Xeon CPUs) and the experimental group consists of data obtained from an ARM cluster—6 low-cost Orange Pi One single-board-computers using ARM processors. The ARM cluster was found to be significantly more cost-efficient than the traditional Intel server.*

Keywords: cluster computing, ARM, cost

1 Introduction

Computing needs can be met by using a variety of hardware solutions. Cost is typically a key factor and this is especially true for massive data processing tasks such as indexing the World Wide Web. In this paper we compare the cost efficiency of two very different computing solutions: a cluster of low cost ARM processors versus a much smaller number of fast, but more costly, Intel servers. Industry typically favors the Intel solution, but we hypothesize that the ARM solution is more cost efficient. Prior work has shown mixed results. One study indicated that ARM processors are not as efficient at High Performance Computing tasks [2]. However, another study demonstrated that some general purpose computing tasks are more efficient when using the ARM processors [3].

Companies such as Google need to constantly process webpages in order to maintain an up-to-date database to support web searches [4]. Processing this constant influx of new or modified web pages requires a large amount of processing power, and hence web page indexing is a relevant and important data processing task. In this paper we use a web indexing task as our processing benchmark for comparing the relative cost effectiveness of the ARM and Intel solutions.

For this study we constructed an ARM cluster using 6 Orange Pi One Single-Board Computers (SBCs) and compared its performance against that of an Intel Server. The benchmark web page indexing task was executed on both computing solutions and the cost efficiency for each solution was determined. Our results indicate that the ARM cluster is significantly more cost-effective than the Intel server solution.

This paper is organized as follows. Section 2 describes the two processor architectures that are evaluated as well as the cluster design. Section 3 provides a detailed description of the benchmark web indexing task and Section 4 provides our main results. Our conclusion are provided in Section 5.

2 Processor Architectures

In this Section we compare the current Intel and ARM Cortex-A7 microarchitectures, as well as describe the Single-Board Computer chosen for the comparison. The Orange Pi One (see Figure 1) was chosen as our low cost solution because each unit only costs $9.99, has reasonable performance due to its four ARM Cortex-A7 cores, and its 100M Ethernet provides decent networking capabilities.

Figure 1: Orange Pi One S~~b~~BC with ARM processor

Originally designed as a low power multicore architecture for Android smartphones/tablets, the ARM Cortex-A7 uses a minimalist pipeline in order to efficiently handle a large number of low activity background tasks. This is in contrast to the Intel processors, which are designed to handle a smaller number of high intensity foreground tasks, with a deep pipeline and individual L2 caches for each core. However, the deeply pipelined design of the Intel processor may decrease performance when handling a large number of tasks. Webpage

ISBN: 1-60132-458-8, CSREA Press ©

indexing involves a large number of simultaneously executing tasks—with hundreds or even thousands of threads—and hence the ARM processor might be better designed for such tasks. Table 1 provides some detailed information about both processors.

Table 1: Comparison of Intel vs. ARM Architectures

	INTEL	ARM CORTEX-A7
μOP/clock	8	2
SIMD	128-bit x2	64-bit shared
Pipeline Depth	20-24 stages	8 stages
L2 Cache	Individual	Shared 4 cores
Data Width	64-bit	32-bit

The ARM cluster uses 6 Orange Pi One Single Board Computers (SBCs). Each of these Orange Pi Ones has 4 ARM Cortex-A7 cores, assorted coprocessors, a Cat-5 networking port, 512 MB of RAM, and a GPU (unused). The Orange Pi Ones do not have fans; instead they are passively cooled. An 8 port networking hub is used to connect these SBCs together and to provide access to the Internet. The constructed ARM cluster has a total cost of $200. The Intel Server uses 2 Quad-Core Intel Xeon processors and is actively cooled. The Intel server used in this study has a total cost of $2,535. One important difference between the ARM and Intel solution is that for ARM solution the peripherals (most importantly the network controller) are integrated directly into the SoC of the Orange Pi One, whereas for the Intel server they are connected over PCIe. This gives the Orange Pi One and the ARM solution higher performance when accessing peripherals (i.e. networking), which is important for our web pages benchmarking task that requires intensive network access.

Feature-wise, the ARM and Intel solutions are very similar. Both support hardware-accelerated virtualization. All ARM Cortex-A processors have supported this feature—even those designed for smartwatches and set-top boxes. The SoC used in the Orange Pi One and the Intel CPU both support acceleration for the AES encryption algorithm. The SoC in the Orange Pi One also supports several other common cryptographic algorithms.

3 Benchmark

In order to compare the cost effectiveness of the two computing solutions, a benchmark program was written to index Wikipedia pages. Wikipedia was chosen for its relatively uniform page structure and reliable ping times, as well as its wealth of available URLs for extraction. The indexing program was written in Google Go [1] because it has an efficient HTML tokenizer and an efficient concurrency system that allows it to handle the large number of pages that must be indexed concurrently.

This indexing process, which is shown in Figure 2, occurs in the following sequence. First, the link is sent to the indexer (second rectangle from the bottom in the second column in Figure 2). Then the indexer extracts the title of the web page, a list of words as well as their frequency of occurrence, a list of headings, a list of the text paragraphs, and a list of hyperlinks along with the associated text describing them. The resulting page data is then sent to the page recurser which iterates through the links and sends the Wikipedia links to the hash distributer. The hash distributer then hashes the links and the hash value is used to route the links to the appropriate node (either the Intel server or one of the Orange Pi Ones). Next, the deduplicators store a list of links in order to prevent them from being repeatedly reindexed. The links that were not previously indexed are placed into a buffer. The output of this buffer is automatically distributed among all of the nodes. These links are then sent to the indexers and the process continues. All channels (the labelled arrows in Figure 2) are buffered in order to prevent any part of the indexing pipeline from needing to wait for more work to do.

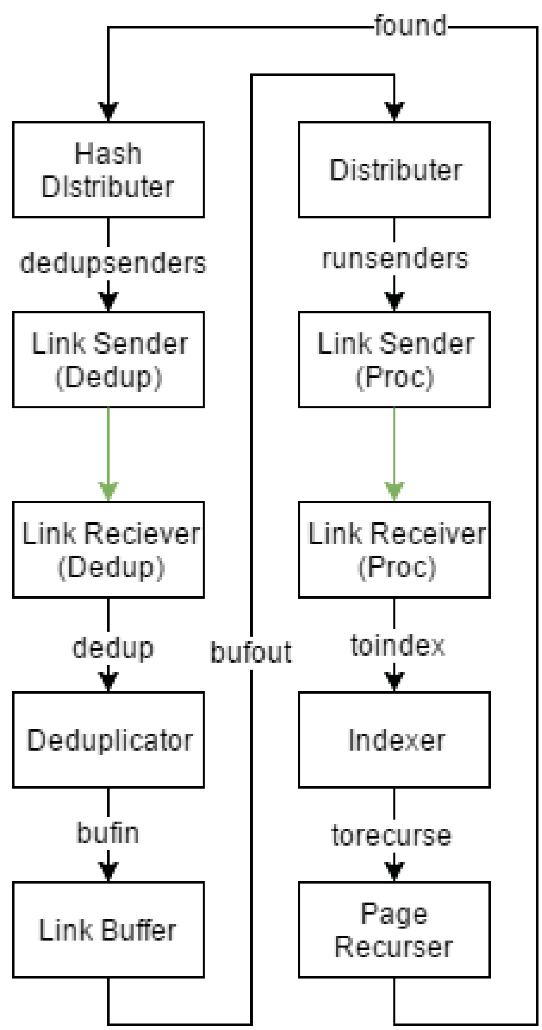

Figure 2: Flowchart of Web Page Indexing Benchmark

ISBN: 1-60132-458-8, CSREA Press ©

The indexer was started and then allowed to run for 2 minutes. After the 2 minutes, the number of links processed was recorded for each node and sent to the control node, which saved it to a file. All data sent over the network was encoded with gob (a high performance binary protocol for Google Go).

The benchmark program is executed 12 times on both the Intel server and the ARM cluster. On the Intel server, 200 pages were indexed concurrently per core, while on the ARM cluster 10 pages were indexed concurrently per core. These values were selected because we determined that they were the highest values that did not lead to connection timeouts.

4 Results

The benchmark was run 12 times on the ARM cluster and Intel server in order to obtain a sample size sufficient for statistical analysis. After the benchmark runs were completed, the results were recorded and analyzed. The results for the 12 runs are displayed in Table 2. The "Pages" field specifies the total number of pages indexed in the 2 minute period. The speed in pages per second (Pages/s) was then calculated by dividing the total number of pages indexed in this two minute period by 120. Finally, the cost effectiveness ($/Page/s) was calculated by dividing the cost of the computing solution by the speed in Pages/s. Recall that the cost of the ARM cluster is $200 and the cost of the Intel server is $2,535.

Table 2: Web Indexing Benchmark Results

	ARM			INTEL		
Run	**Pages**	**Pages/s**	**$/Page/s**	**Pages**	**Pages/s**	**$/Page/s**
1	7,739	64.49	3.10	21,286	177.38	14.30
2	6,636	55.30	3.62	20,193	168.28	15.07
3	8,104	67.53	2.96	19,550	162.92	15.57
4	6,458	53.82	3.72	21,442	178.68	14.19
5	8,223	68.53	2.92	19,658	163.82	15.48
6	8,441	70.34	2.85	16,773	139.78	18.14
7	8,538	71.15	2.81	21,331	177.76	14.27
8	7,710	64.25	3.12	17,202	143.35	17.69
9	8,774	73.12	2.74	17,636	146.97	17.26
10	7,116	59.30	3.38	17,426	145.22	17.46
11	8,273	68.94	2.90	19,768	164.73	15.39
12	8,614	71.78	2.79	20,607	171.73	14.77
Avg.	**7,886**	**65.71**	**3.08**	**19,406**	**161.72**	**15.80**

The results in Table 2 show that, despite being approximately 13 times ($2535/$200) as expensive as the ARM cluster, the Intel server is on average only ~2.5x faster (161.72/65.71). Therefore, the ARM cluster is 5.13 (15.80/3.08) times as cost effective as the Intel server. In order to match the Intel server's cost effectiveness, the ARM cluster would need to be upgraded to accommodate 15 Orange Pi One SBCs—which would cost only $362.13.

Table 3: Summary Data Table for Cost per Unit Work ($/Page/s)

	ARM	**INTEL**
n	12	12
Mean	3.08	15.80
Standard Dev	0.31	1.39

Table 3 shows the summary cost-effectiveness statistics over the 12 runs. Clearly the mean cost per unit work for the ARM solution is much lower than for the Intel solution. A two-tailed t-test indicates that the results are statistically significant with $p=8.3\times10^{-13}$.

5 Conclusion

This research shows that for general performance computing tasks (which involve processing of network data rather than intense mathematical processing) the ARM Cortex-A7 is more cost effective than Intel processors. Industry may be finally ready to exploit this advantage. Very recently (while this study was being conducted), a cloud computing company called Scaleway added ARM severs to their datacenters in order to deliver cheaper VMs and bare-metal servers than their competitors. Scaleway sells ARM servers with 4 CPU cores, 2GB of memory, and 50GB of SSD for €2.99/month ($3.27/month at the time of writing), either as bare-metal or a virtual machine. However, Scaleway only has datacenters in Paris and Amsterdam, and very few other large cloud hosting companies have adopted ARM servers.

There are also many other characteristics of ARM processors which may work well at large scale—including low power consumption, low heat output, and low hardware maintenance requirements. ARM has also released a new core (ARM Cortex-A35) which is faster and more efficient than the Cortex-A7. This will likely make the ARM even more cost effective than the Intel solution, but no devices are available using this core at the time of writing this. There are also many other ARM SBCs and cores which can be compared.

6 References

[1] I. Balbaert. The Way to Go: A Thorough Introduction to the Go Programming Language. IUniverse, 2012.

ISBN: 1-60132-458-8, CSREA Press ©

[2] M. F. Cloutier, C. Paradis, and V. M. Weaver. Design and Analysis of a 32-bit Embedded High-Performance Cluster Optimized for Energy and Performance. In *Proceedings of the 1st International Workshop on Hardware-Software Co-Design for High Performance Computing*, pp. 1-8, New Orleans, LA, IEEE, 2014.

[3] Z. Ou, B. Pang, Y. Deng, J. K. Nurminen, A. Yla-Jaaski, and P. Hui. Energy- and Cost-Efficiency Analysis of ARM-Based Clusters. In *Proceedings of the 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pp. 115-123, Washington, DC: IEEE Computer Society, 2012.

[4] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank Citation Ranking: Bringing Order to the Web, Stanford InfoLab, 1999.