

# Quantitative Musings on the Feasibility of Smartphone Clouds

Chen Chen, Moussa Ehsan, Radu Sion  
Stony Brook Computer Science  
{chen18,mehsan,sion}@cs.stonybrook.edu

**Abstract**—“Green” and its “low power” cousin are the new hot spots in computing. In cloud data centers, at scale, ideas of deploying low-power ARM architectures or even large numbers of extremely “wimpy” nodes [1,2] seem increasingly appealing. Skeptics on the other hand maintain that we cannot get more than what we pay for and no free lunches can be had.

In this paper we explore these theses and provide insights into the power-performance trade-off at scale for “wimpy”, back-to-basics, power-efficient RISC architectures. We use ARM as a modern proxy for these and quantify the cost/performance ratio precisely-enough to allow for a broader conclusion. We then offer an intuition as to why this may still hold in 2030.

## I. INTRODUCTION

**Energy.** Energy is a top 3 data center operating cost component [3] and “fully burdened power” (consumption+distribution) constitutes upwards of 30% of a large data center operating costs [4]. In 2011, Google’s data centers alone were consuming the equivalent electricity sufficient to power up 200,000 homes [5].

It is thus not surprising that so many research efforts have attempted to reduce data center power consumption, ranging from mechanisms for workload consolidation, virtualization (for increased utilization), decommissioning of unused servers, and the purchase of increasingly energy-efficient servers [6].

Overall, a large body of work deals with how to build more efficient and cost effective data centers. Two main strategies dominate: scale-up – increase individual nodes’ capacities and performance, and scale-out – cluster a larger number of lower-end nodes to perform the same amount of work at a fraction of the cost.

**ARM vs. x86.** In a scale-out model, another line of thought that seems to get traction recently suggests changing the processors altogether to one of the available low-power architectures to achieve sizable benefits in overall power consumption and cost. Ideas of deploying low-power ARM architectures or even large numbers of extremely “wimpy” nodes [1,2] seem increasingly appealing, and bold claims that “2014 will be the year in which ARM breaks the x86 monoculture” abound [7].

Skeptics on the other hand [8] caution that energy-efficiency “has long been a tough primary sell in business IT [...] partly because such “soft” solutions typically require customers to make stiff up-front capex investments in order to capture longer-term opex benefits. [...] broader market trends and events have measurably dimmed the likelihood that ARM will disrupt the IT infrastructure marketplace to the degree that proponents have hoped.” In other words, we get what we pay

for. A number of online blogs, startups’ marketing hype and heated discussions can be found debating this topic.

Here we explore these theses and quantify the cost / performance ratio precisely-enough to draw a broad conclusion without loss of generality. We then project into the future.

**Cost/Performance Ratio.** Data center cost analysis is difficult to get right. Numerous studies focus on different individual cost components: some on operation costs, others on ownership costs [9–12], yet others propose various lease-vs-buy trade-offs [13] etc.

The main insight we build upon is that the ultimate edge in such a dynamic cost-driven market is simply a better cost/performance ratio, including both *end-to-end* CAPEX and OPEX components (energy, hardware, cooling, staffing, space etc). Plainly, if the actual cost of owning and running a fixed unit of meaningful high-level data center work in a new architecture (e.g.,ARM) based data center will be lower than running the same unit of work in an x86 center, the new architecture is likely to slowly but surely take over, period. And vice-versa.

**Note: In the following we chose to use ARM as a proxy for any more “wimpy”, back-to-basics architecture with similar characteristics. We in no way want to imply that Intel/AMD are not going to produce a viable competitor in this space.**

However, to understand the cost/performance ratios of ARM vs. x86 platforms we require results that quantify *end-to-end* computing costs with enough precision to validate in actual current cloud vendor pricing such as the models of Chen et al. in [14]. They determine the total cost of a x86 computing cycle in modern infrastructures of different scale, and then prove its validity using current cloud provider retail price. Here we extend their model to ARM and derive the cost of a compute cycle in ARM-based data centers of large scale. We then factor in the fact that ARM (RISC) will require more cycles on average to achieve the same unit of high-level work when compared to x86 (CISC) and compare.

**Markets.** To properly evaluate the cost of a hypothetical large ARM-based data center, many data points are required, most important of which being the server hardware cost (representing up to 57% of the data center costs [4]).

There exists no comparable mass-market for ARM servers yet. This is likely in part due to the major players (Intel, AMD) pushing the competing x86 platform and the fact that ARM has licensed designs to only a handful of (comparatively smaller)

players (e.g., IBM, HP).

It would not be sound to use today’s (sometimes astronomical) prices of ARM-based servers in any analytical computation. A good proxy for a pervasive ARM-based ecosystem may be today’s smartphone market. With billions of devices produced yearly [15] it is 2-3 orders of magnitude larger than server markets. This ensures a great competitive environment and significant downward price pressures.

We propose to thus rephrase the question slightly and more excitingly: “does it make economical sense to build a data center out of *actual* ARM-based smartphones or their SoC (system on chip) hardware and if so, by what margins?”

**Existing Work.** We are not the first and surely not the last to ask this question. In seminal work, Andersen et al. [2, 16] introduced FAWN, “a fast, scalable, and energy-efficient cluster architecture for data-intensive computing. A FAWN cluster links together a large number of “wimpy” nodes built using energy-efficient processors and small amounts of flash memory into an ensemble cluster that can perform the same amount of work as a traditional cluster but at a fraction of the power.” Further, Marinelli et al. [17] present a Hadoop-derived MapReduce system that supports cloud computing on a networked collection of Android mobile devices. Similarly, Dou et al. [18] also introduce a MapReduce framework targeting any device that supports Python and network connectivity. Jain et al. built a “virtual cloud computing platform, named MC<sup>2</sup>, using nearby mobile devices.” [19]. Arslan et al. [20] use idle smartphones being charged overnight to build a distributed computing infrastructure. Harizopoulos et al. [21] propose the “concept of a Micro-Cellstore (MCS), a stand-alone data-appliance housing dozens of recycled smartphones” etc.

This paper aims to augment this highly technical work with a cost feasibility analysis. To this end we take first steps towards understanding the ARM vs. x86 cost and performance trade-offs. Specifically, we (i) quantify the actual end-to-end dollar cost of an ARM-based data center CPU cycle and (ii) compare the cost of a unit of high-level work of ARM and x86 platforms at scale. We compute the costs of cycles for both ARM and x86 data centers at scale, by updating previous work of Chen et al. [14]. We then consider the RISC/CISC “cycle count” [22] (more RISC instructions are needed for the same task) and compute the cost of a given unit of high-level work on both ARM and x86 platforms. We also explore the case of multi-core servers.

**ARM processors vs. x86 Server processors.** Today’s ARM processors used in mobile devices are faster and orders of magnitude more energy-efficient than a 5 year old server.

For example, a Samsung Exynos 4412 CPU consumes only 2.845W [23], almost two orders of magnitude less than a typical Intel i5 Sandy Bridge processor (95W). Similarly, numerous studies favorably analyze the energy efficiency of smartphones and other ARM-based devices. Ou et al. [24] show that Pandaboard ARM-based clusters are 1.2 to 9.5 times more energy-efficient than Intel-based hardware.

Further, the Snapdragon 800 used in the Samsung Galaxy Note 3 features a 2.27 GHz quad-core CPU and the Primate

Labs’ Android benchmark chart shows that its single core performance scores are up to about 50% of an Intel Xeon E5-2609 core @ 2.4GHz. [25].

Moreover, wholesale manufacturers are advertising Android-capable SoCs for under \$15/unit in large quantities; ARM-based SoCs such as the BeagleBoard [26] and Raspberry Pi [27] can be had for \$25-45. Thus, overall, today, considering also OTS rackable enclosure and power supply hardware, a hypothetical SoC/PCB-mounted ARM core with supporting chipset is priced around \$85 (in a 4 core setup), whereas a modern x86 core (on a motherboard) at a similar frequency hovers around \$250.

## II. COST MODEL

In [14] Chen et al. explore the cost of data center computing (CPU cycles, networking, storage) in various environments by the amortized cost of an x86 CPU cycle, defined as “Cycle Cost”. They determine a boundary condition that defines when cloud computing becomes viable, i.e., when CPU cycle cost savings are enough to offset the client-cloud network distance. Here we summarize for more details refer you to [14].

**Computing Environments.** The cost of computing is a function of scale and varying environment complexity: home (H), small (S), mid-size (M) and large size data centers (L).

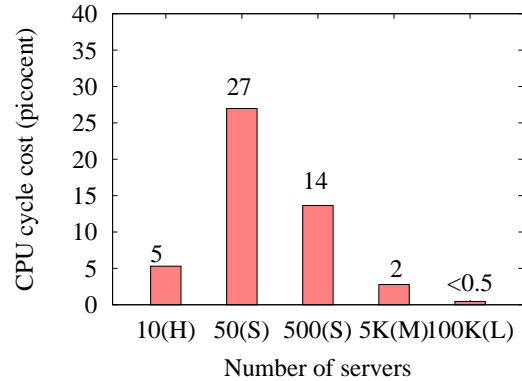


Fig. 1. x86 cycle costs.

**Cost factors.** A number of cost factors come into play across all of the above levels. These can be divided into inter-dependent vectors, including: hardware (servers, networking gear), building (floor space leasing), energy (running hardware and cooling), service (administration, staffing [28], maintenance), and network service. Other breakdown layouts of these factors are possible [29].

Considering an analytical end-to-end cost model integrating these cost factors, the amortized cost of a CPU cycle (x86) in various environments was computed and validated with actual cloud provider pricing data at the time of publication. We succinctly describe this process below and note that the cost model can be simplified by equation (1):

$$CycleCost = \frac{Hardware + Energy + Service + Floor}{Total Cycles} \quad (1)$$

Using Table I notations, this becomes equation (2).

$$\begin{aligned}
\text{CycleCost} &= \frac{\text{Server} + \text{Network} + \text{Energy} + \text{Service} + \text{Floor} + \text{Infrastructure}}{\text{Total Cycles}} \\
&= \frac{\frac{\lambda_s}{\tau_s} + \frac{\lambda_w \cdot N_w}{\tau_w \cdot N_s} + (w_p \cdot \mu + w_i \cdot (1 - \mu)) \cdot \text{PUE} \cdot \lambda_e + \frac{\lambda_p}{\alpha} + \frac{(w_p \cdot \mu + w_i \cdot (1 - \mu)) \cdot \text{PUE} \cdot \lambda_f}{\beta} + \frac{(w_p \cdot \mu + w_i \cdot (1 - \mu)) \cdot \text{PUE} \cdot \lambda_i}{\tau_i}}{\mu \cdot v \cdot \eta \cdot \sigma} \\
&= \frac{\frac{\lambda_o \cdot \eta \cdot \sigma \cdot v + C_s}{\tau_s} + \frac{\lambda_w \cdot N_w}{\tau_w \cdot N_s} + ((w_o^p \cdot \eta \cdot \sigma + C_o^p \cdot \sigma + C_p) \cdot \mu + (w_o^i \cdot \eta \cdot \sigma + C_o^i \cdot \sigma + C_i) \cdot (1 - \mu)) \cdot \text{PUE} \cdot \lambda_e + \frac{\lambda_p}{\alpha} + \frac{((w_o^p \cdot \eta \cdot \sigma + C_o^p \cdot \sigma + C_p) \cdot \mu + (w_o^i \cdot \eta \cdot \sigma + C_o^i \cdot \sigma + C_i) \cdot (1 - \mu)) \cdot \text{PUE} \cdot \lambda_f}{\beta} + \frac{((w_o^p \cdot \eta \cdot \sigma + C_o^p \cdot \sigma + C_p) \cdot \mu + (w_o^i \cdot \eta \cdot \sigma + C_o^i \cdot \sigma + C_i) \cdot (1 - \mu)) \cdot \text{PUE} \cdot \lambda_i}{\tau_i}}{\mu \cdot v \cdot \eta \cdot \sigma} \quad (3)
\end{aligned}$$

[14] reports the resulting cost (Figure 1) of a CPU cycle to range from 0.45 picocents ( $1 \text{ US picocent} = 10^{-14} \text{ USD}$ ) for large data centers to 27 picocents for small enterprises. [14] then validated these results using the pricing of main cloud providers: Amazon [30], Google [31], Microsoft (Figure 2).

Provider	Picocents
Amazon EC2	0.93 - 2.36
Google AppEngine	up to 2.31
Microsoft Azure	up to 1.96

Fig. 2. Today's pricing.

**Infrastructure.** Hardware costs include computing devices, network equipment, and infrastructure. The network equipment cost can be estimated by evaluating the number of required switches and routers. The infrastructure includes all the other equipments such as power equipment, UPS, cooling equipment, etc.

$$\text{Hardware} = \text{Server} + \text{Network} + \text{Infrastructure} \quad (3)$$

Since infrastructure was not thoroughly considered in [14] we further define it here. The cost of infrastructure ( $\lambda_i$ ) is reported in dollars per watt (\$/W) and depends directly on the power draw of the data center [32]. The lifespan of infrastructure (typically 12-15 years) is significantly longer than that of computing devices and network equipment [4]. In a modern data center,  $\lambda_i$  averages about 10 \$/W [32].

Symbol	Definition
$N_s, N_w$	number of servers, switches
$\alpha$	server : administrator ratio
$\beta$	watt per sq ft
$\lambda_s, \lambda_w$	server, switch price
$\lambda_p, \lambda_f$	personnel, floor cost/sec
$\lambda_e$	electricity price/(watt-sec)
$\mu$	CPU utilization
$v$	CPU frequency
$\eta$	number of CPU cores per socket
$\sigma$	number of sockets per server
$\tau_s, \tau_w, \tau_i$	servers, switches, infrastructure lifespan
$w_p, w_i$	server power at peak, idle
$\lambda_i$	infrastructure cost/watt
$w_o^p, w_o^i$	power per core at peak, idle
$\lambda_o$	price per core
$C_s$	the constant base price per server
$C_p, C_i$	the constant base power per server at peak, idle
$C_o^p, C_o^i$	the constant base power per socket at peak, idle
$\rho$	ARM server cores #/x86 server cores #

TABLE I  
NOTATIONS.

**Multi-core, multi-socket servers.** Multi-core multi-CPU servers are the current norm. For example, in the Xeon line, Intel released the Beckton processors in 2010 with up to eight cores [33]. Further, single socket servers are only found in entry-level configurations and most of the x86 servers in the market today contain multiple CPU sockets. For example, the HP ProLiant DL series of rack servers include products with 2, 4 or 8 processors (Table II).

DL100 series	DL500 series	DL900 series
1U	4U	8U
1-2 processors	2-4 processors	4-8 processors
2-8 cores per processor	4-10 cores per processor	6-10 cores per processor
\$3000-8000	\$9000-29000	\$22000-98000

TABLE II  
SPECS FOR HP PROLIANT SERVERS

**CPU Cycles.** After incorporating these additional considerations in the cost model we now proceed to determine the current cost of a CPU cycle (x86).

We note that some of the parameters in Table I are dependent of each other or correlated with server configuration data. Specifically, an important relationship exists between the number of cores within a server  $\eta \cdot \sigma$  and parameters  $\lambda_s$  (server price),  $w_p$  (server peak power),  $w_i$  (server idle power). W.l.o.g., for illustration purposes, consider a specific server line – the HP ProLiant rack servers [34, 35] – which provides validated data points on server power consumption for custom server configurations, using its online "HP Power Advisor" [34] tool.

The reported baseline server power consumption before adding any CPU and DRAM considerations lies in the 45-60W ballpark consistent also with our in-house observations at idle with minimal RAM. As the number of cores is increased (and 2GB DRAM per additional core are added), we observe the behavior illustrated in Figure 3. As can be seen, when averaging across available processor flavors (for each given number of cores) an almost perfect linearity ensues.

Similarly, Figure 4 illustrates the price dependency on the product between the number of server cores and their CPU frequency (effectively modeling server computation power). When averaging again across processor flavors for each given number of cores, the behavior is linear.

We model the observed overall linearity of  $\eta \cdot \sigma$  in the following equations where  $\lambda_o, w_o^p, w_o^i$  represent the price, peak power and idle power for a single CPU core respectively and  $C_s, C_p, C_i$  are three constants:

$$\lambda_s = \lambda_o \cdot (v \cdot \eta \cdot \sigma) + C_s \quad (4)$$

$$w_p = w_o^p \cdot (\eta \cdot \sigma) + C_o^p \cdot \sigma + C_p \quad (5)$$

$$w_i = w_o^i \cdot (\eta \cdot \sigma) + C_o^i \cdot \sigma + C_i \quad (6)$$

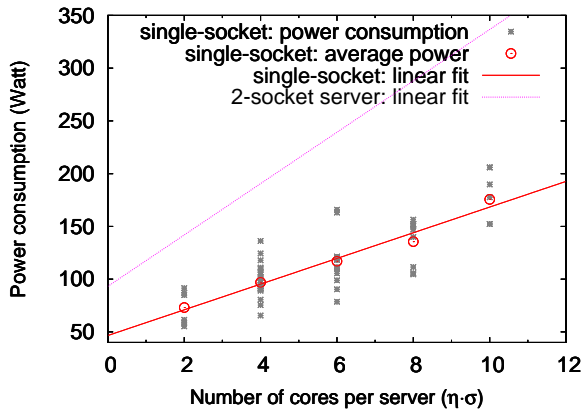


Fig. 3. The relationship between the number of cores and power consumption of a single socket server plotted for a large variety of processor types (HP ProLiant servers). As can be seen the averages can be fitted to mode a linearity with a correlation larger than 93%!

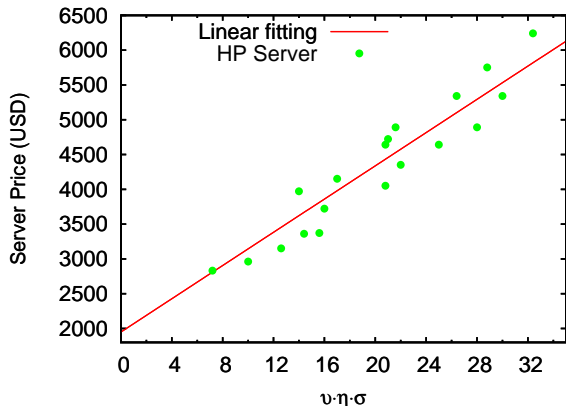


Fig. 4. Server price ( $\lambda_s$ ) with increasing server computation power ( $v \cdot \eta \cdot \sigma$ ).

Note that, for servers with the same number of sockets, equations (5) and (6) can be simplified to linear functions of the number of per-server cores with the slope-defining constants changing for different number of sockets.

For the currently most common data center configuration – a 1U server with 2 CPU sockets [36] – the result of a trade-off between cost, performance, system management etc. – the following values result:  $w_o^p \approx 25W$ ,  $C_o^p \cdot \sigma + C_p \approx 100W$ ,  $w_o^i \approx 5W$ ,  $C_o^i \cdot \sigma + C_i \approx 50W$ ,  $\lambda_o \approx 50 \sim 110USD$ ,  $C_s \approx 1930USD$ . In the following, unless specified otherwise we consider a 1U rack server with two Intel E5-2603 processors as a baseline.

Finally, equation (2) illustrates the dependency of a CPU cycle cost from the number of cores  $\eta \cdot \sigma$ . Figure 5 illustrates how at scale, cycle cost decreases with increasing server computation power ( $v \cdot \eta \cdot \sigma$ ) – increasing number of per-server cores or increasing core frequency. As expected, increasing either (or both) increases overall compute density which translates in more efficient CPU cycles. The Figure also shows point of diminishing return, beyond a certain compute density threshold, after which other baseline cost and consumption factors start to dominate equation (2).

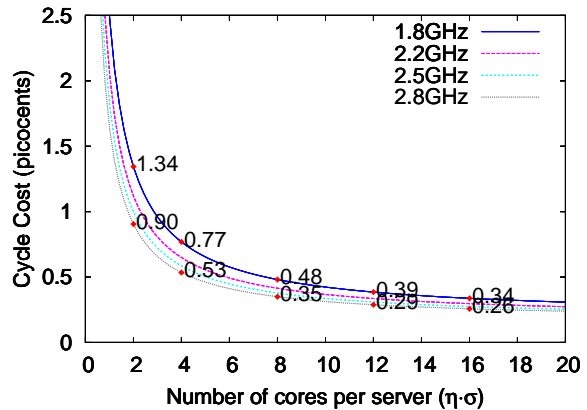


Fig. 5. CPU cycle cost for x86 data center with increasing per-server core number and core frequency. As expected, increasing either (or both) – and thus increasing density of computing – reduces the data center CPU cycle cost. A point of diminishing return is reached beyond a certain density, once other baseline cost and consumption factors start to overcome in equation (2).

### III. ARM VS. X86 DATA CENTERS

Equipped with a general understanding of x86 CPU cycle costs, we now explore the attainable cost of ARM cycles in a cost-efficient ARM-based data center of equivalent scale. To properly evaluate these however, as outlined in the introduction, a number of data points are required, most important of which being the cost of server hardware (representing up to 57% of the data center monthly costs [4]).

Yet, since there exists no comparable mass-market for ARM servers yet – according to even the most optimistic sources, ARM-based servers market share percentage lies in the single-digits – it would not be sound to use today’s (sometimes astronomical) prices of ARM-based servers in the analytical computation a baseline for the future.

We posit however, that a good proxy for a pervasive ARM-based ecosystem is today’s smartphone market. With billions of devices produced yearly [15] it is 2-3 orders of magnitude larger than server markets and aligns well with the move towards cheap, short life-cycle, high-density SoC-based solutions that some of the current vendors are aiming for in the cloud market.

**Base Costs.** Under these considerations, one approach to computing overall ARM cycle cost would be to directly estimate absolute values of the components impacted by the main differences between ARM SoC’s and equivalent x86 hardware and then plug these back into the above cost model summarized by equation (2).

However, driven by the insight that we are ultimately really interested in the ARM/x86 ratio  $\frac{CycleCost_{ARM}}{CycleCost_{x86}}$ , we observe we can take a more direct shortcut, as follows.

The main model parameters that feature differences between ARM and x86 can be grouped in three categories: server price ( $\lambda_s$ ), energy consumption ( $w_p \cdot \mu + w_i \cdot (1 - \mu) \cdot PUE$ ) and utilized CPU cycles ( $\mu \cdot v \cdot \eta \cdot \sigma$ ). Now, consider the following ARM/x86

parameter sub-ratios:

$$\begin{aligned}
 R_S &= \frac{\lambda_s^{ARM}}{\lambda_s^{x86}} \\
 R_E &= \frac{w_p^{ARM} \cdot \mu^{ARM} + w_f^{ARM} \cdot (1 - \mu^{ARM})}{w_p^{x86} \cdot \mu^{x86} + w_f^{x86} \cdot (1 - \mu^{x86})} \\
 R_C &= \frac{v^{ARM} \cdot \mu^{ARM} \cdot \eta^{ARM} \cdot \sigma^{ARM}}{v^{x86} \cdot \mu^{x86} \cdot \eta^{x86} \cdot \sigma^{x86}} \quad (7)
 \end{aligned}$$

Then, if  $p_x$  denotes the percentage that  $x \in \{server, energy, floor, infrastructure, network, service\}$  represents of the x86 cycle cost – e.g.,  $p_{server,x86}$  is the percentage of x86 server hardware of the cost of a cycle [4, 14] – the main ARM/x86 ratio of interest becomes

$$\begin{aligned}
 \frac{CycleCost_{ARM}}{CycleCost_{x86}} &= \frac{R_S}{R_C} \cdot p_{server,x86} + \frac{1}{R_C} \cdot (p_{network,x86} + \frac{\alpha_{x86}}{\alpha_{ARM}} \cdot p_{service,x86}) + \\
 &\quad \frac{R_E}{R_C} \cdot (p_{energy,x86} + p_{floor,x86} + p_{infrastructure,x86}) \quad (8)
 \end{aligned}$$

**Scale.** For comparison purposes, we will assume a similar scale (cloud-sized) and its associated architecture-independent parameters in [14] unchanged: overall data center utilization, PUE, the watt-per-sqft coefficient  $\beta$ , the number of networking switches per node, the unit price of switches, floor rental pricing, personnel hourly rates, electricity, infrastructure and the lifespan of most non-server hardware.

#### A. Strawman Scenario

To illustrate, first consider the strawman scenario of a server with a single processor ARM SoC (a smartphone) and a mid-line HP ProLiant server with one socket as discussed in the following. Later we explore the more interesting scenario of a multi-SoC ARM server.

**Hardware.** Examples of popular ARM SoCs include Qualcomm Snapdragon which runs its own ARM v7 instruction-set CPU “Krait”, and the Samsung Exynos SoC with a licensed ARM Cortex-A\* line of CPUs. In the following, w.l.o.g. We consider the popular, widely deployed and benchmarked ARM Cortex-A9 as run in a Samsung Galaxy S III GTI9300 (S3) on a Samsung Exynos 4412 SoC (with 4 cores). Its detailed specs [23, 35] are listed in Table III. Noting that we use the smartphone spec as the ARM server spec here.

	Samsung Galaxy S3 (Exynos 4412)	HP ProLiant DL160 Gen8 (Intel Xeon E5-2603)
Cores num( $\eta \times \sigma$ )	4	4
Peak power ( $w_p$ )	$\approx 7W$	$\approx 160$
Idle power ( $w_i$ )	$\approx 0.8W$	$\approx 70$
CPU frequency ( $v$ )	1.4GHz	1.8GHz
Price( $\lambda_s$ )	\$370	\$2000

TABLE III  
CONSIDERED HARDWARE SPECS.

The sub-ratio values in equation (7) for the ARM Cortex-A9 are listed in Table IV. As can be seen all are less than 1. Yet their orders of magnitude vary. Since the energy efficiency of smartphones is much higher,  $R_E$  is one order of magnitude smaller than the other two ratios.

$R_S$	$R_E$	$R_C$
0.185	0.016	0.778

TABLE IV  
ARM/x86 SUB-RATIOS.

Finally, to account for the architectural differences, we apply the “cycle count” method introduced in [22] – RISC processors will run more instructions (than their CISC peers) to execute a given amount of higher level work. Cycle counts for ARM exceed 2.1 (2.1 for ARM v7 Cortex-A9 and 2.5 for ARM v7 Cortex-A8) for the representative “CloudSuite” workload [37] when normalized to a third generation Intel Core. Table V summarizes the resulting ARM CPU cycle costs.

	Cycle Cost (picocents)	Cycle Count (x86 normalized)	Equivalent Cycle (picocents)
ARM Cortex-A9	0.25	2.1	0.525
x86 Intel Core	0.628	1	0.628

TABLE V  
“EQUIVALENT CYCLE” COST

The results surprise and are highly supportive of the ARM hypothesis. The cost of performing the same amount of work in an ARM data center composed of weak single CPU smartphone is virtually less than the cost of the work in an x86 data center!

However, it is important to note that the numbers used in the comparison are ARM-favorable. Specifically, the considered powerdraw and ARM chipset costs are associated with smaller amounts of DRAM and other components and peripherals than the x86 comparison servers. Thus, in next section, we considered the hypothetical multi-SoCs ARM server.

#### B. Multi-SoC ARM servers

We have identified above the fact that an infrastructure composed of smartphones as single-SoC ARM servers features virtually lower *equivalent cycle costs* (cost of same unit of high level work [37]) when compared with a specific single socket x86 server as defined in Table III.

Yet what happens in the more realistic scenario of modular, multi-SoC ARM servers [38, 39]? To answer this, we apply the above cost model to ARM servers with increasing numbers of multi-core Exynos 4412 SoCs, under the assumption that the supporting baseline costs (e.g., server rack costs, enclosures, etc) in equations (4), (5) and (6) do not change <sup>1</sup> (Section II).

The baseline cost can be simply regarded as the sum of individual supporting component costs in the server. According to the online Power Calculators provided by either server manufacturers or other organizations [40–42], the power consumption and price for each supporting component are listed in Table VI. They are consistent with the numbers derived in the 2 sockets x86 server models.

As illustrated in [36], the two sockets x86 server is the most commonly used in current data center. We also use the x86

<sup>1</sup>Note that this is a somewhat cautious conservative scenario since in practice, due to their low power and efficient heat dissipation profiles, baseline costs for ARM SoC servers will likely be significantly cheaper than x86.

Component	Power consumption (Watt)	Price (\$)
Memory(4GB)	$\approx 3$	80
HDD Drive	$\approx 18 - 25$	150-200
Fan	3-5	30-40
PCI	$\approx 15$	$\approx 140$
Power Supply	$\approx 20 - 50$	100-300

TABLE VI  
POWER CONSUMPTION AND PRICE FOR SUPPORTING COMPONENTS.

server with 2 CPU sockets as the baseline in this part. To simplify the figure, we assume that the CPU frequency of x86 cores are the same as the x86 server in section III

Figure 6 illustrates ARM cycle cost behavior with increasing numbers of SoCs. The number of SoCs increasing is represented by the increasing  $\rho$  (ratio between the number of cores per server in ARM and x86 infrastructures, respectively). As expected, the ARM proposition becomes even more competitive with an increasing  $\rho$ . Moreover, the more cores x86 server has, the smaller  $\rho$  is needed for a ARM data center to get the same equivalent cycle cost. For example, comparing the magenta line and the blue line in Figure 6. The ARM server needs more than 2 times of the cores in one server to get the “equivalent cycles” as the 4 cores x86 server. However, for a baseline of 16 cores x86 server, only 1.15 times of the cores is needed.

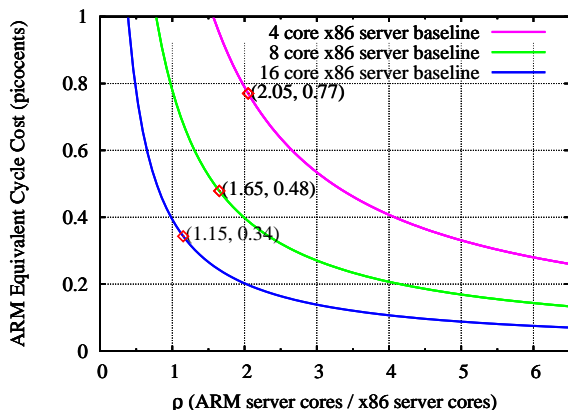


Fig. 6. ARM cycle cost improves with increasing  $\rho$  (ratio between the number of cores in ARM vs. x86 servers). The three curves correspond to different baselines for the number of cores in x86 servers (4,8,16 cores per x86 server) and increasing  $\rho$ . The red points (rhombus) in each line represent the  $(\rho, \text{ARM equivalent cycle cost})$  threshold at which costs of “equivalent cycles” for ARM become the same as in the corresponding x86 data center. An ARM data center with an average  $\rho$  beyond this point (greater than) provides cheaper “equivalent cycles” than its x86 counterpart baseline.

Finally, we would like to understand which of the main ARM/x86 parameters sub-ratios  $R_S$  (ratio of server price),  $R_E$  (ratio of energy consumption), and  $R_C$  (ratio of utilized CPU cycles) discussed in Section III, equation (7) are most significant in the  $\frac{\text{CycleCost}_{\text{ARM}}}{\text{CycleCost}_{\text{x86}}}$  ratio. Figure 7 shows that for multi-core scenarios with increasing  $\rho$ , the slope of  $R_C$  (ratio of maximum per-server CPU cycles) dominates.

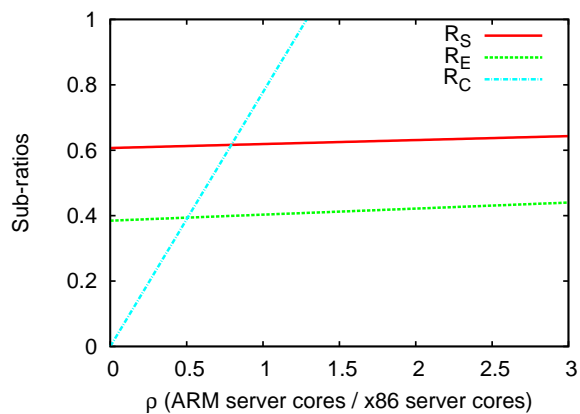


Fig. 7. ARM vs. x86:  $R_S$  (ratio of server price),  $R_E$  (ratio of energy consumption), and  $R_C$  (ratio of CPU cycles).  $R_C$  dominates.

### C. Cost factors and their impact

According to equations (2) and (8), lower cycle cost for data center can be get in three way: lower server price, lower power consumption per server or higher computation power per server. However, it’s scarcely possible to find a server that optimizes for all these vectors. In practice this leads to trade-offs. Higher computation power can be achieved by either increasing the computation power of single processor or increasing the number of sockets per server. These two scenarios have already been discussed. Next, we discuss the trade-off between server price and server power consumption.

Naturally, price and power consumption are not independent. Figure 8 shows how data center cycle cost changes with different server price and power consumption values. The plane in 3D space illustrates a certain linear relationship between cycle cost, server price, and server power consumption. Corner A represents an optimal solution which is hard to achieve. Corners B and C seem to be near optimal solutions that are more realistic. Figure 9 illustrates the projection of the contours in Figure 8 in the XY plane. It’s worth noting that the contours are all parallel to each other regardless the change of computation power per server. The slopes are such that for every 100W additional power consumed, the server price should decrease by about 360 dollars to maintain the same data center cycle cost for the identical compute capacity.

## IV. INSIGHTS. LIMITATIONS.

**Fundamental Insight: Performance/Energy.** One of the main fundamental insights that drive the above results is the fact that performance/energy consumption ratios vary across architectures and may be stacked more favorably in the case of low-power designs in general and ARM cores in particular.

To see why that is the case, consider the DMIPS/mW ratio comparison for ARM vs. x86 architectures. DMIPS (Dhrystone MIPS) is a common representation of the Dhrystone [43] benchmark score. To get the DMIPS value, the score is normalized by the number of Dhrystones per second obtained on a 1 MIPS machine. As a result, DMIPS has almost the same meaning as MIPS (million instructions per second) but it

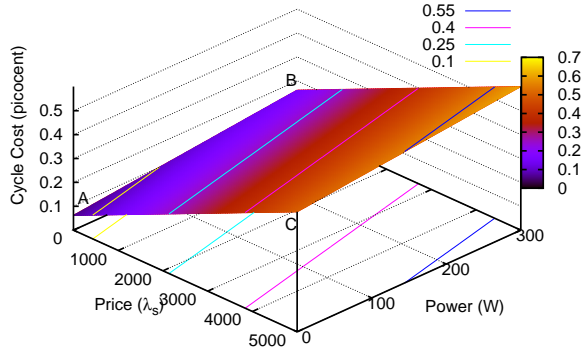


Fig. 8. This figure shows how the data center cycle cost is affected by server price ( $\lambda_s$ ) and power consumption under the assumption that all the hypothetical data centers built with any kinds of servers have the same computation power per server (equivalent of 14.4 GHz in cycles). Naturally, lower price and lower power consumption both bring down the data center cycle cost. The best case is a server with both parameters very small, which responses to the corner A in the figure. When A is not available, the next best cases with one parameter to be optimized are either corner B or C.

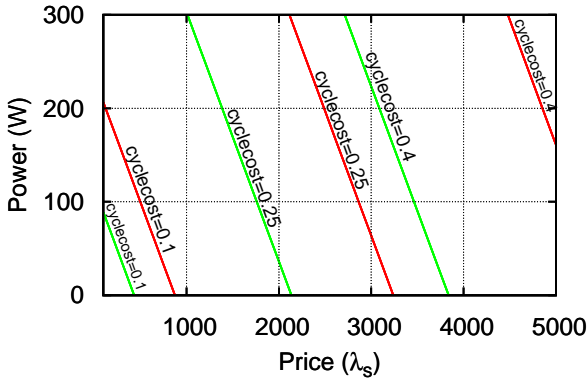


Fig. 9. This Figure represents the projection of Figure 8 on the XY plane. The green lines are the contours for a per-server equivalent computation power of 14.4GHz in cycles ( $v \cdot \eta \cdot \sigma = 14.4GHz$ ). The cycle cost for each of the contours are 0.1, 0.25, and 0.4 respectively. The red lines are the contours when the computation power increased to 20GHz per server. A server can be more expensive or more power consuming for the same cycle cost when the computation power increases. The two group of lines for servers with different computation power are parallel to each other, however the line for the same cycle cost shifts showing that the trade-off threshold between server price and server power is about  $\$3.6/W$  independently of server computation power.

represents the performance result more meaningfully because “the Dhrystone score counts only the number of program iteration completions per second, allowing individual machines to perform this calculation in a machine-specific way” [43]. This means that DMIPS becomes a platform-independent performance measurement.

Table VII lists the DMIPS/mW for both ARM and x86 processors across multiple application markets ranging from high-performance processing for feature rich OSes to deeply

embedded real-time applications [44–48].

Architecture	Processors	Year	DIMPS/mW
x86	Intel Xeon E5-2690 v2	2013	3.46
	Intel Xeon E3-1230 v3	2013	1.24
	Intel Xeon L5530	2009	1.29
	Intel Core i7 3630QM	2012	2.513
	Intel Core i5-2500K	2011	0.874
	Intel Atom N270	2008	1.538
	AMD FX-8150	2011	0.871
	AMD Optero 4376HE	2012	1.909
ARM	ARM Cortex-R7	2011	> 46
	Samsung Exynos 5250 (Cortex-A15-like)	2011	3.5
	AppliedMicro X-Gene	2012	4.5
	ARM Cortex-A9	2009	16
	ARM Cortex-A5	2011	20

TABLE VII  
PERFORMANCE/ENERGY FOR DIFFERENT PROCESSORS.

Further, Figures 10 and 11 illustrate the performance/energy ratio for representative ARM and x86 implementations. It can be seen that a significant inter-platform difference persists. Even when comparing ARM cores with low-power Intel Atom CPUs, ARM processors feature a 3-5x higher performance for the same TDP (Figure 11). This is one of the main fundamental factors that drive the conclusions above.

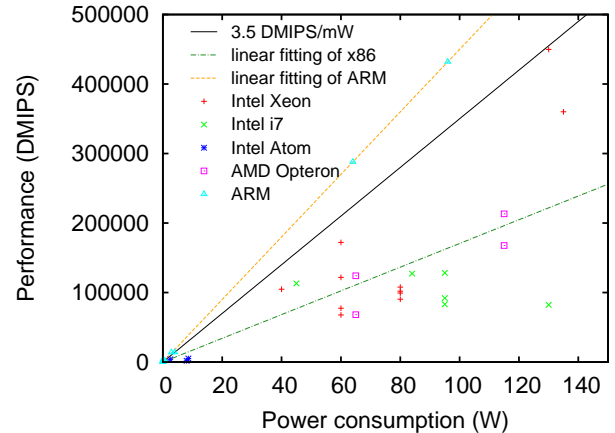


Fig. 10. The relationship between Performance (DMIPS) and power consumption for different modern processors. **(upper left is better than lower right)** The orange line and the green line are the linear fitting result for x86 processors and ARM processors respectively, which roughly describe the main development trend of each group. The significant gap between these two lines illustrates the advantages of ARM over x86 in the overall case (linear fitting occurs at about  $3.988 \pm 0.326 DMIPS/mW$  for ARM and  $0.467 \pm 0.203 DMIPS/mW$  for x86). The black solid line in the figure represents a performance/energy ratio of 3.5 DMIPS/mW. As can be seen, all existing x86 cores come in significantly below this line and all the ARM samples are in or above this line. It is also worth noting that the ARM processors located near this boundary are some relatively old processors while the x86 samples near the boundary are released in the recent 2 years. This suggests a certain potential convergence in the medium to long-term future.

**Architectural Details.** Understanding the details of why ARM design points result in better performance/power ratios is a subject of interest in the architecture community and

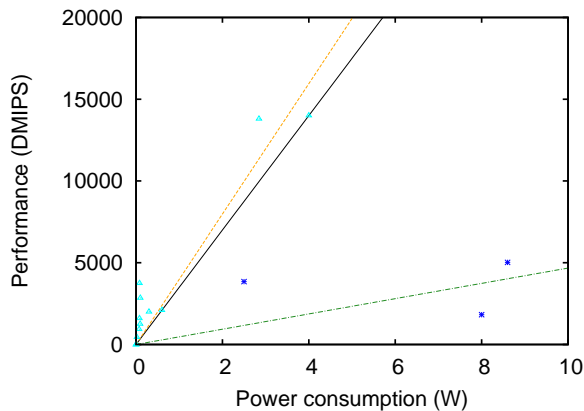


Fig. 11. Zoom out of Figure 10 for the low-power x86 (Intel Atom) and ARM processors. It can be seen that the ARM and low power x86 processors are basically following their fitting lines respectively and the significant difference between them is preserved.

somewhat out of scope here. However it may be worth outlining one of the major reasons: predictive branching and speculative execution. ARM simply does less of it. Much less. And every time an x86 chip speculatively executes a “losing” branch, it sacrifices additional energy on the altar of overall speed through coverage of the “winning” branches. Most ARM designs simply speculate *much* less aggressively. A second reason for the reduced power consumption is smaller on-chip caches and additional circuitry.

**Why Cycle Cost?** In this paper we chose an easy route – considering easily quantifiable items – and argued that ultimately the cost of cycles is what makes one architecture preferable to others in a data center. Note that this did not imply that only the CPU-related costs were taken into account. The model formulates the cycle cost as a function of all related costs including memory and network transmission energy and cost, as well as instructions-per-cycle architectural differences taken into account by using the “cycle count” paradigm. Nevertheless it is important to note that, to achieve a holistic end-to-end view, a large number of other factors should be considered before switching to a new architecture such as prefetch capabilities, bus speeds and others some of which are discussed below.

**Cache Sizes. RAM.** Processors cannot be considered in isolation. Caches, memory and storage are important as well, especially for e.g., niche workloads in data-intensive scenarios. A typical x86 processor can easily handle significant amounts of RAM (e.g., 1TB+ RAM sizes are not unusual in today’s servers). Nevertheless, while current ARM designs are not necessarily optimized for such large memory footprints, “server grade” 64 bit ARM processors are emerging, including designs such as the Cortex-A50 [49, 50]. And, as more and more low-power designs make their way into the modern data center, it will important to understand how the ability to handle and address significant DRAM capacity is impacting the performance/power consumption ratio, e.g., through increase cache sizes and prefetching logic.

**Workloads and I/O.** The cost model introduced here did not directly account for I/O and instead assumed that a server with a given CPU computation power can handle similar I/O throughput independent of the base CPU architecture. This is obviously subject to much debate, especially since such multi-power servers only seem to reach the market now [51]. Nevertheless, initial results seem to support this thesis. For example, the latest m400 HP server [51] features extremely high memory bandwidth and high I/O throughput, making it good at moving a lot of data into memory quickly. To this end, HP chose a modular design point in which it packages 64GB of DRAM, 10Gbps networking and FLASH memory together with a 8 core ARM CPU in a single cartridge. A chassis can package up to 45 cartridges.

**Everything Else.** Additionally, this study did not consider any of the significant potential costs associated with running a different architecture, management tools, and high level software tailored for ARM. These are all additional costs that would need to be absorbed by any ARM-based deployment.

## V. CONCLUSIONS

**Today.** The overarching conclusion of this work is that low-power, “wimpy” CPUs (for which we used ARM as a proxy) have the potential to be significantly more (cost) effective than x86 in cloud infrastructures at scale.

This conclusion is slightly dampened however by certain limitations of existing ARM/low-power chip designs, specifically their ability to handle massive data streams easily. This seems to be not an issue in reality however, as several emerging massive-I/O platforms seem to suggest [51].

And, as multi-core modular ARM server markets become a reality, this cost differential may be just enough to offset the additional costs of having to migrate to a different architecture, and system software.

**The Future.** Yet, as CISC/x86 cores become increasingly energy efficient, their cycle costs may become just cheap enough to tilt the CISC/x86-RISC/ARM trade-off balance. To see whether this is the case, consider the evolution of the computation speed to power draw ratio (DMIPS/mW) for x86, which has closely followed Moore’s law for the past two decades (Figure 12).

Figure 12 also illustrates actual transistor counts and ultimately shows that in the (mW/transistor) is in fact also subject to Moore, thus implying platform independence. As long as RISC/ARM chips are fabricated with a modern-enough technology, and unless x86 unilaterally breaks out of Moore, there is little hope for an ARM-x86 trade-off balance tilt to occur. Also,  $\rho$  is likely to continue to stay high in the future, since CISC cores consume and dissipate significantly more energy and their number cannot be arbitrarily increased without increasing form factors, cooling, energy conduits etc.

Further, Figure 13 illustrates the projected costs of “equivalent cycles” (actual cycle cost  $\times$  cycle count) in both ARM and x86 architectures until 2030. Moore’s law has been applied to the hardware cost and energy component of a cycle. As can be seen, ARM power advantages are likely to



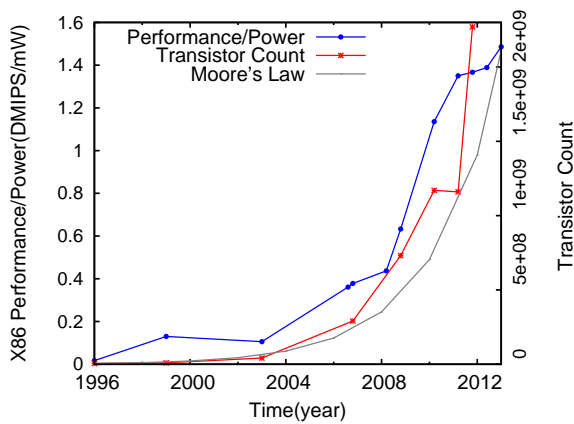


Fig. 12. Performance/Power (x86).

play an decreasingly important role over time and both types of infrastructures are likely to provide increasingly cheaper “equivalent cycles”. Yet, for any  $\rho \geq 2$  the ARM/low-power proposition is maintaining its cost advantage.

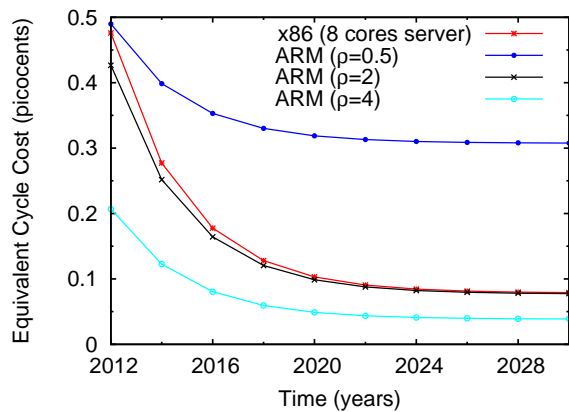


Fig. 13. Forecast. When compared with an 8 core x86 server baseline, ARM servers composed of a single smartphone SoCs (strawman case) have no cost advantage. Modular multi-SoC ARM servers featuring higher  $\rho$  values provide a clear advantage however.  $\rho$  is likely to stay high over time since x86/CISC cores consume and dissipate significantly more energy and cannot be arbitrarily increased in existing enclosures.

Finally, these results may constitute a first step towards understanding the scale-out vs. Scale-up trade-off for general purpose computing in clouds. With the advent of longer-distance high speed interconnects, RISC may just have made it back to town.

#### ACKNOWLEDGMENT

This research was supported in part by NSF grants NSF CNS-1161541, NSF CNS-1318572, NSF CNS-1223239, NSF CCF-0937833, by US ARMY award W911NF-13-1-0142, and by gifts from Northrop Grumman Corporation, Parc/Xerox, and Microsoft Research. The authors would like to thank Professor Mike Ferdman who helps with architecture problems in the paper.

#### REFERENCES

- [1] W. Lang, J. M. Patel, and S. Shankar, “Wimpy node clusters: What about non-wimpy workloads?” in *Proceedings of the Sixth International Workshop on Data Management on New Hardware*. ACM, 2010, pp. 47–55.
- [2] D. G. Andersen, J. Franklin, M. Kaminsky, A. Phanishayee, L. Tan, and V. Vasudevan, “Fawn: a fast array of wimpy nodes,” in *SOSP*, J. N. Matthews and T. E. Anderson, Eds. ACM, 2009, pp. 1–14.
- [3] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, “The cost of a cloud: research problems in data center networks,” *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 1, pp. 68–73, 2008.
- [4] J. Hamilton, “Perspectives Blog,” Online at <http://mvdirona.com/jrh/work/>.
- [5] J. Glanz, “Google Details, and Defends, Its Use of Electricity,” *The New York Times*, online at <http://goo.gl/zXiDOQ>.
- [6] “Top Twelve Ways to Decrease the Energy Consumption of Your Data Center,” Online at [http://www.energystar.gov/index.cfm?c=power\\_mgt.datacenter\\_efficiency](http://www.energystar.gov/index.cfm?c=power_mgt.datacenter_efficiency).
- [7] Graeme Caldwell, “2014 Will Be The Year That ARM Breaks The x86 Monoculture,” Online at <http://goo.gl/FoZskL>.
- [8] Charles King, “Will ARM Get a Cold Shoulder from Data Center Owners?” Online at <http://www.ecommercetimes.com/story/79785.html>.
- [9] J. Koomey, K. Brill, P. Turner, J. Stanley, and B. Taylor, “A simple model for determining true total cost of ownership for data centers,” *Uptime Institute White Paper, Version*, vol. 2, 2007.
- [10] D. J. Cappuccio, “Use a tco model to estimate the costs of your data center,” 2012.
- [11] M. K. Patterson, D. Costello, P. Grimm, and M. Loeffler, “Data center tco; a comparison of high-density and low-density spaces,” *Thermal Challenges in Next Generation Electronic Systems (THERMES 2007)*, 2007.
- [12] B. Martens, M. Walterbusch, and F. Teuteberg, “Costing of cloud computing services: a total cost of ownership approach,” in *System Science (HICSS), 2012 45th Hawaii International Conference on*. IEEE, 2012, pp. 1563–1572.
- [13] R. A. Dines, “Build or buy? the economics of data center facilities,” 2011.
- [14] Y. Chen and R. Sion, “To cloud or not to cloud?: musings on costs and viability,” in *SoCC*, J. S. Chase and A. El Abbadi, Eds. ACM, 2011, p. 29.
- [15] Statista, “Global smartphone shipments from 2010 to 2017,” Online at <http://www.statista.com/statistics/263441/global-smartphone-shipments-forecast/>.
- [16] D. G. Andersen, J. Franklin, M. Kaminsky, A. Phanishayee, L. Tan, and V. Vasudevan, “Fawn: A fast array of wimpy nodes,” *Commun. ACM*, vol. 54, no. 7, pp. 101–109, Jul. 2011. [Online]. Available: <http://doi.acm.org/10.1145/1965724.1965747>
- [17] E. E. Marinelli, “HyraX: cloud computing on mobile devices using mapreduce,” DTIC Document, Tech. Rep., 2009.
- [18] A. Dou, V. Kalogeraki, D. Gunopulos, T. Mielikainen, and V. H. Tuulos, “Misco: A mapreduce framework for mobile systems,” in *Proceedings of the 3rd International Conference on Pervasive Technologies Related to Assistive Environments*, ser. PETRA ’10, 2010, pp. 32:1–32:8. [Online]. Available: <http://doi.acm.org/10.1145/1839294.1839332>
- [19] P. Jain, R. Kabra, S. Rustagi, T. Bansal, D. Patel, and V. Raychoudhury, “Mc2: On-the-fly mobile compute cloud for computational intensive task,” in *Proceedings of the 5th IBM Collaborative Academia Research Exchange Workshop*, ser. I-CARE ’13, 2013, pp. 7:1–7:4. [Online]. Available: <http://doi.acm.org/10.1145/2528228.2528236>
- [20] M. Y. Arslan, I. Singh, S. Singh, H. V. Madhyastha, K. Sundaresan, and S. V. Krishnamurthy, “Computing while charging: Building a distributed computing infrastructure using smartphones,” in *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies*, ser. CoNEXT ’12, 2012, pp. 193–204. [Online]. Available: <http://doi.acm.org/10.1145/2413176.2413199>
- [21] S. Harizopoulos and S. Papadimitriou, “A case for micro-cellstores: Energy-efficient data management on recycled smartphones,” in *Proceedings of the Seventh International Workshop on Data Management on New Hardware*, ser. DaMoN ’11, 2011, pp. 50–55. [Online]. Available: <http://doi.acm.org/10.1145/1995441.1995448>
- [22] E. Blem, J. Menon, and K. Sankaralingam, “Power struggles: Revisiting the risc vs. cisc debate on contemporary arm and x86 architectures,” in

*High Performance Computer Architecture (HPCA2013)*, 2013 IEEE 19th International Symposium on. IEEE, 2013, pp. 1–12.

- [23] A. Carroll and G. Heiser, “The systems hacker’s guide to the galaxy energy usage in a modern smartphone,” in *Proceedings of the 4th Asia-Pacific Workshop on Systems*, ser. APSys ’13, 2013, pp. 5:1–5:7. [Online]. Available: <http://doi.acm.org/10.1145/2500727.2500734>
- [24] 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 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (Ccgird 2012)*, ser. CCGRID ’12, 2012, pp. 115–123. [Online]. Available: <http://dx.doi.org/10.1109/CCGrid.2012.84>
- [25] P. L. Inc, “Primate Labs’ Android benchmark chart,” Online at <http://browser.primatelabs.com/android-benchmarks>.
- [26] “Meet the Beagles: Open Source Computing,” Online at <http://beagleboard.org>.
- [27] “Raspberry Pi — An ARM GNU/Linux box for \$25. Take a byte!” Online at <http://raspberrypi.org>.
- [28] C. D. Patel and A. J. Shah, “Cost model for planning, development and operation of a data center,” 2005.
- [29] APC, “Determining total cost of ownership for data center and network room infrastructure,” Online at [http://www.apcmedia.com/salestools/CMRP-5T9PQG\\_R3\\_EN.pdf](http://www.apcmedia.com/salestools/CMRP-5T9PQG_R3_EN.pdf).
- [30] “Amazon Elastic Compute Cloud,” Online at <http://aws.amazon.com/ec2>.
- [31] Google Inc., “Google App Engine,” Online at <http://code.google.com/appengine/>.
- [32] W. P. Turner IV and K. G. Brill, “Cost model: Dollars per kw plus dollars per square foot of computer floor,” *White paper*, 2008.
- [33] Wikipedia, “Xeon,” Online at <http://goo.gl/5hv3Je>.
- [34] HP, “HP Power Advisor,” Online at <http://h18000.www1.hp.com/products/solutions/power/advisor-online/HPPowerAdvisor.html>.
- [35] —, “Hp proliant dl rack servers,” Online at <http://goo.gl/mTDXa>.
- [36] U. Hoelzle and L. A. Barroso, *The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines*, 2nd ed. Morgan and Claypool Publishers, 2013.
- [37] M. Ferdman, A. Adileh, O. Kocberber, S. Volos, M. Alisafae, D. Jevdjic, C. Kaynak, A. D. Popescu, A. Ailamaki, and B. Falsafi, “Clearing the clouds: a study of emerging scale-out workloads on modern hardware,” in *Proceedings of the seventeenth international conference on Architectural Support for Programming Languages and Operating Systems*, ser. ASPLOS ’12, 2012, pp. 37–48. [Online]. Available: <http://doi.acm.org/10.1145/2150976.2150982>
- [38] M. Anderson, “Europe Wants a Supercomputer Made From Smartphones,” Online at <http://spectrum.ieee.org/computing/hardware/europe-wants-a-supercomputer-made-from-smartphones>.
- [39] J. hamilton, “AMD Announces Server Targeted ARM Part,” Online at <http://perspectives.mvdirona.com/2012/10/29/AMDAnnouncesServerTargetedARMPart.aspx>.
- [40] Dell, “Dell Datacenter Capacity Planner,” Online at [http://www.dell.com/html/us/products/rack\\_advisor\\_new/index.html](http://www.dell.com/html/us/products/rack_advisor_new/index.html).
- [41] HP, “HP Power Advisor,” Online at <http://h18000.www1.hp.com/products/solutions/power/advisor-online/HPPowerAdvisor.html>.
- [42] eXtreme Outer Vision, “eXtreme Power Supply Calculator Lite,” Online at <http://www.extreme.outervision.com/psucalculatorlite.jsp>.
- [43] Wikipedia, “Dhrystone,” Online at <http://en.wikipedia.org/wiki/Dhrystone#Results>.
- [44] —, “Instructions per second,” Online at [http://en.wikipedia.org/wiki/Instructions\\_per\\_second#cite\\_note-94](http://en.wikipedia.org/wiki/Instructions_per_second#cite_note-94).
- [45] P. Kennedy, “Intel Xeon E5-2600 V3 Review,” Online at <http://www.tomshardware.com/reviews/intel-xeon-e5-2600-v3-haswell-ep,3932-7.html>.
- [46] A. Ltd, “ARM Processors,” Online at <http://www.arm.com/products/processors/index.php>.
- [47] P. Kennedy, “Intel Xeon E5-2697 v2 12 Core Ivy Bridge-EP Benchmarks,” Online at <http://www.servethehome.com/intel-xeon-e5-2697-v2-12-core-ivy-bridge-ep-benchmarks/>.
- [48] P. H. Katie Roberts-Hoffman, “ARM Cortex-A8 vs. Intel Atom: Architectural and Benchmark Comparisons,” Online at <http://caxapa.ru/thumbs/229665/armcortexa8vsintelatomarchitecturalandbe.pdf/>.
- [49] A. Holdings, “Cortex-A53 Processor,” Online at <http://www.arm.com/products/processors/cortex-a/cortex-a53-processor.php>.
- [50] —, “Cortex-A57 Processor,” Online at <http://www.arm.com/products/processors/cortex-a/cortex-a57-processor.php>.
- [51] Y. SVERDLIK, “HP Starts Shipping 64-bit ARM Servers,” Online at <http://www.datacenterknowledge.com/archives/2014/09/29/hp-starts-shipping-64-bit-arm-servers/>.