

Greening the Internet with Nano Data Centers

Vytautas Valancius
Georgia Tech
valas@gatech.edu

Nikolaos Laoutaris
Telefonica Research
nikos@tid.es

Laurent Massoulié
Thomson
laurent.massoulie@thomson.net

Christophe Diot
Thomson
christophe.diot@thomson.net

Pablo Rodriguez
Telefonica Research
pablorr@tid.es

ABSTRACT

Motivated by increased concern over energy consumption in modern data centers, we propose a new, distributed computing platform called Nano Data Centers (NaDa). NaDa uses ISP-controlled home gateways to provide computing and storage services and adopts a managed peer-to-peer model to form a distributed data center infrastructure. To evaluate the potential for energy savings in NaDa platform we pick Video-on-Demand (VoD) services. We develop an energy consumption model for VoD in traditional and in NaDa data centers and evaluate this model using a large set of empirical VoD access data. We find that even under the most pessimistic scenarios, NaDa saves at least 20% to 30% of the energy compared to traditional data centers. These savings stem from energy-preserving properties inherent to NaDa such as the *reuse of already committed baseline power* on underutilized gateways, the avoidance of *cooling costs*, and the reduction of *network energy consumption* as a result of demand and service co-localization in NaDa.

Categories and Subject Descriptors

C.2.4 [Computer Communication Networks]: Distributed Systems—Distributed Applications

General Terms

Design, Management, Measurement

Keywords

Energy Efficiency, Data Centers, Nano Data Centers, Video Streaming

1. INTRODUCTION

Most current Internet applications are served from a large number of collocated servers stacked together in one of multiple *data center* facilities around the world. This centralized hosting model is a classic example of the economies of scale: large numbers of similar servers yields relatively low manning requirements and eases

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CoNEXT'09, December 1–4, 2009, Rome, Italy.

Copyright 2009 ACM 978-1-60558-636-6/09/12 ...\$10.00.

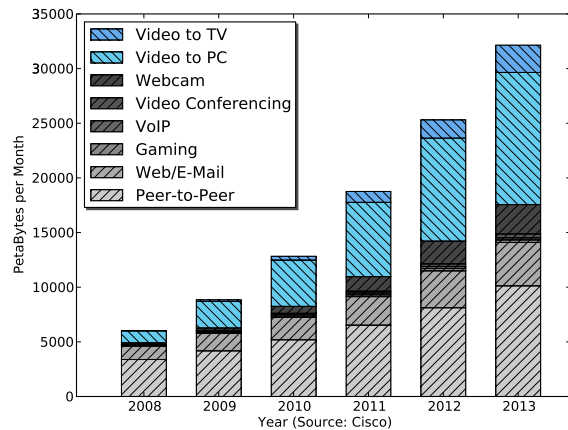


Figure 1: Growth of video content in the Internet.

procurement procedures. Homogeneous hosting environments allow for better resource optimization and management, while extensive use of virtualization technologies provide an abstraction of dedicated platforms to developers.

Centralization trend is not without its limitations. Data centers are prone to: 1) over-provisioning, 2) high cost of heat dissipation, and 3) increased distance to end-users. All of these issues, as we will see later, lead to ever increasing energy bills that the data center and network operators need to foot. The data centers are *over-provisioned* because they need to match the peak demand despite that the average load remains much lower throughout most of the day; in addition, redundancy requirements might push up the number of necessary servers significantly. Data centers also are *expensive to cool*. Despite significant efforts for improving the server power-efficiency, an average data center spends as much energy on cooling as it spends for powering its servers [23]. Even in the data centers using state-of-the-art cooling technologies heat dissipation accounts for at least 20% to 50% of the total power consumption [4]. Centralization trend also increases the data center *distance to the users*. Not every service can be hosted at well connected central interconnection points. Higher distance from end users increases bandwidth-mileage requirements and adds to the energy consumption of the networking equipment. It is not surprising therefore that data centers recently made headlines with reports that they consume 1.5% of total US electricity [3], or that their carbon-dioxide emissions are projected to surpass those of the airline industry by the year 2020 [6]. These concerns are indeed

expected to amplify in view of growth projections of data center-hosted applications like video distribution [13] (Figure 1).

While data centers are an example of centralization trend, another, opposite trend manifests through Peer-to-Peer (P2P) systems. The decentralized P2P systems free-ride on end-user computers and, being spatially distributed, incur little or no heat dissipation costs. In addition, some P2P systems can exploit locality and reduce the redundancy requirements, because each P2P user often is P2P server at the same time. Unfortunately, conventional P2P system performance typically suffers from free-riding, node churn, and lack of awareness of underlying network conditions.

Motivated by the problem of energy consumption, in this paper we propose a new way to deliver Internet services based on our *Nano Data Center* (NaDa) platform. The key idea behind NaDa is to create a distributed service platform based on tiny managed “servers” located at the edges of the network. In NaDa, both the nano servers and access bandwidth to those servers are controlled and managed by a single entity (typically an ISP) similarly to what is suggested in [20, 27]. Significant opportunities already exist for hosting such tiny servers on ISP owned devices like Triple-Play gateways and DSL/cable modems that sit behind standard broadband accesses. Such *gateways* form the core of the NaDa platform and, in theory, can host many of the Internet services currently hosted in the data centers. In this paper, however, we will focus on video streaming services, which, as shown in Figure 1, exhibited large growth in past few years. In addition, video streaming services allow for easier energy usage modeling and simulation. Figure 2 shows a high level architecture of such streaming architecture using home gateways as nano servers.

NaDa follows a P2P philosophy, but contrary to typical P2P, NaDa is *coordinated* and *managed* by an ISP that installs and runs the gateways that act as nano servers, as well as the network that interconnects them. Due to its managed nature, NaDa avoids most of the shortcomings of classic unmanaged P2P. ISPs can easily implement NaDa by providing new customers with slightly over dimensioned gateways, whose extra storage and bandwidth resources would be used by NaDa to implement services like video hosting, all of which will be totally isolated from the end-user via virtualization technologies. Thus, with a rather small investment in higher capacity devices, ISPs can go beyond their traditional role as communication carriers, and enter the potentially highly profitable service and content hosting market.

Obviously designing a system like NaDa spans a magnitude of issues that cannot all be covered here and thus in this paper we mainly focus on energy efficiency which, as discussed earlier, is probably the Achilles heel of monolithic data centers. As we will demonstrate in coming sections, NaDa can reduce the energy costs of offering Internet services, particularly content distribution. The savings come in three ways. First, through improved *heat dissipation* stemming from the distributed nature of NaDa which minimizes the need for cooling. Second, through *traffic localization* resulting from the co-location between NaDa and end users. Finally, NaDa performs *efficient energy use* by running on gateways that are already powered up and utilized for content access or other services. NaDa avoids wasting the high baseline power already paid for in an online gateway. We elaborate further on these energy saving aspects in Section 2.

Our main contributions are the following. We develop a model to evaluate the energy needed to provide services in both centralized data centers and distributed NaDa. This model relies on a large collection of empirical data from various sources, including power measurements of operational content distribution servers and end-user devices (Section 3). We then apply this model in the evalua-

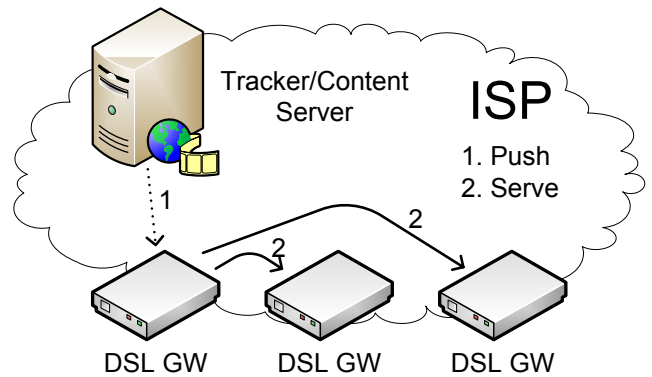


Figure 2: High level NaDa architecture. Content is served from home gateways whenever possible.

tion of energy consumption in the context of video-on-demand. We detail NaDa platform and placement mechanisms for video content (Section 4), and use trace-driven simulations to quantify energy savings under various interesting configurations. We find that even under the most pessimistic conditions NaDa, achieves at least 15% to 30% overall energy savings compared to traditional data centers, while more optimistic scenarios achieve up to 60% greater energy efficiency (Section 5). Related work and final remarks are discussed in Sections 6 and 7 respectively.

2. THE CASE FOR NADA

In this section we present a more detailed argument on why NaDa platform can be more energy efficient than conventional data centers and why it is certainly feasible to build it. We first detail the energy reduction principles of NaDa. We then quantify the availability of the end devices to be used as NaDa nano nodes.

2.1 Advantages of NaDa

Four main principles allow NaDa’s to surpass the energy efficiency of conventional data centers:

Heat Dissipation. Recent reports [4, 19] show that data centers consume large amounts of energy for energy transmission/conversion, and most importantly, cooling and heat dissipation. Such overheads are captured by the Power Usage Efficiency (PUE) metric, which is defined as the ratio between the total power consumed by a data center and the power actually delivered to its IT equipment (servers, networking equipment). The PUE factor ranges from as high as 2.0 in legacy data centers [23] to as low as 1.2 in recent state of the art facilities [4]. Operators are trying to improve their PUE by leveraging advanced scheduling [26], performing heat reuse [22], adopting free flow systems, and even exploiting colder geographical locations [12]. All these approaches, however, have limitations. Scheduling, heat reuse and advanced cooling systems save only a fraction of energy costs; placing data centers in remote locations sacrifices proximity to users and speed of service delivery. Finally, and most important, heat dissipation is hard to tackle due to the high density of co-located IT equipment, which is an inherent characteristic of monolithic data centers.

Church et al. [12] have recently proposed de-densifying IT equipment by spreading them among container sized data centers. NaDa pushes this idea to its extreme by distributing data center IT equipment to user premises where heat dissipation costs are negligible. We show that the additional energy consumed by a single gateway is negligible and thus easy to dissipate.

Service Proximity. Content providers build data centers in multiple locations to reduce service delays/communication costs and improve fault tolerance. Because of high construction costs, only a small number of data centers are built. They then have to be placed at major network interconnection points which are often subject to high real estate costs. Even at such locations, classic data centers are relatively far from end users in comparison to servers of content distribution networks, and *a fortiori* of NaDa, whose PoPs are inside user residences. Apart from reducing delays, service proximity reduces the distance that information has to travel and thus also the energy in powering and cooling the networking equipment that has to carry it.

Self-scalability. Conventional data centers and the networks supporting them are provisioned for peak load, which leads to rather low average utilization [23]. Utilization is worsen further by the fact that backup equipment needs to be powered and keep running so as to be able to receive immediately any load resulting from other failing equipment. Unlike conventional data centers, NaDa is largely self adaptive. As the user population grows, so does the number of available gateways to NaDa. Thus, NaDa reduces over-provisioning while providing built-in redundancy.

Energy efficiency. Today’s routers and servers spend most of their energy on the baseline activities such as running the fans, spinning the disks, powering the backplane, and powering the memory. Even in an idle state, modern systems can be consuming anything from 50% to 80% of the power consumed under maximum load [7, 10]. Therefore, operators constantly try to maximize the utilization of their servers, but this becomes increasingly complicated due to the daily variations of load, as illustrated in the following subsection. Even though operators have perfected service load balancing, idle servers are not necessarily switched off due to the need to keep spare capacity readily available. The problem is even more prominent in networks, where it is hard to shut off a router at off-peak times [24]. In contrast, no baseline powering is paid by NaDa, as it runs on gateways that are already on for servicing another purpose (basic connectivity).

2.2 Gateway availability

Re-using the already committed base-line power of DSL gateways is key to energy savings but depends largely on the number of the devices which can be found online at a point in time. In this subsection we quantify the above number through upper and lower bounds derived from real gateway deployments, and thus provide an idea regarding the expected capacity of NaDa as defined by the above availability of powered devices. The upper bound is computed from gateway response traces produced by Gummadi et al. [14], while the lower bound is derived from set-top-box activity traces, exploiting the fact that if a user is accessing content, his DSL gateway must be up.

Upper bound. While some users keep their gateways online at all times, there definitely exist energy-conscious ones that regularly switch them off when not in need, *e.g.*, during work hours or trips. The study of Gummadi et al. includes reports on the availability of the residential gateways in 12 major ISPs over one month. The data consists of gateway responses to active probing performed every 15 minutes. Based on these data, we computed the gateway uptimes by accumulating all the intervals during which we found a gateway to be active. To discount for possible network effects (such as firewalls) we removed the users that never replied to the probes.

Based on the above analysis, we found that on average, equipment at the end user premises is up 85% of the time¹, as seen in Figure 3. We also observed only insignificant, 5-8% uptime varia-

¹Note though that devices probed by Gummadi et. al. belong to

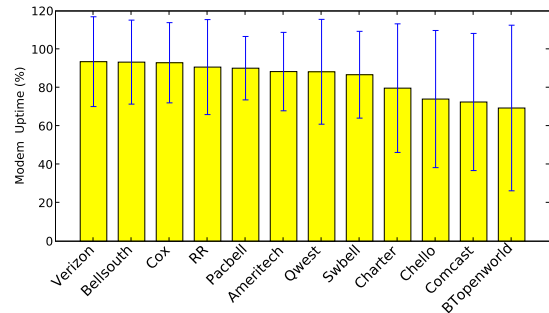


Figure 3: Average user uptime in different provider networks.

tion thought out a day. Therefore, 85% gives a rough upper bound on the time we can expect to find a gateway available for processing some workload sent to it by NaDa.

Lower bound. To derive a lower bound we made the observation that a customer’s gateway has to be up and running for at least as much as the time that we find a user actively accessing content on his set-top-box. To test the idea we resorted to the IPTV traces collected by Cha et al. [9]. The IPTV traces consist of 251,368 active users over 63 days. The traces record each operation the user is performing, including channel selection and switching. Figure 4 shows IPTV user activity throughout the day. We see a high variability in usage, depending on the time of the day. In the peak hour approximately 19% of customers are using their Internet connection for IPTV services; on average the usage is at around 7%.

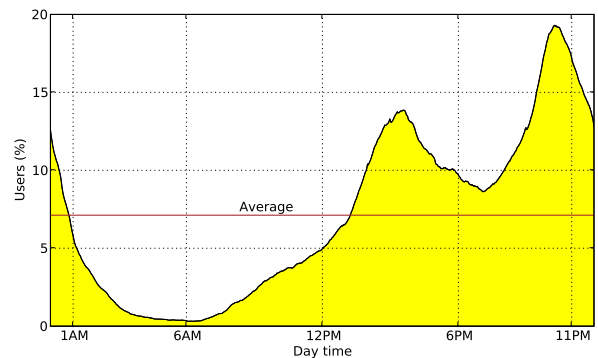


Figure 4: Usage of the IPTV service.

This bound is highly pessimistic since customers can use other services requiring Internet access. For instance, triple play users would want to keep their DSL gateways on because of the telephony service. In Section 5 we will examine both scenarios and show that even in such pessimistic case NaDa achieves considerable energy savings.

3. ENERGY SAVINGS MODEL

In this section we first quantify the energy consumption characteristics of different devices engaged in NaDa. Then we develop an energy consumption model that we use for performing back-of-the-envelope calculations of system-wide energy consumption. This BitTorrent users, thus biasing the sample.

model is further refined in Section 4 to capture additional content related details.

3.1 Basic Parameters

We distinguish between the servers, the gateways and the network in assessing the energy needed to deliver cloud services with NaDa and with monolithic data centers.

Servers. To illustrate typical server performance, we measure a video streaming server used in production in a Sapphire VOD platform [5], which provides both live IPTV and VoD services. More precisely, we monitored a server, with 5GB of RAM and 2 Intel Xeon DualCore 2GHz processors. We varied the load by changing the rates and numbers of video streams downloaded from the server, and tracked the resulting power consumption through a specialized energy measurement device that provides a precision of ± 0.01 Watt (HIOKI 3333 Power HiTester).

The results, reported in Figure 5, show that power is a function of aggregate load, and does not depend on individual video rates or number of concurrent video streams. More importantly, the baseline power (325W) constitutes more than 80% of the overall power use at peak utilization, which corresponds to 1.8Gbps aggregate streaming rate. In addition, we observe that the load-dependent power consumption (obtained by discounting baseline power) depends linearly on aggregate load.

Thus for a service rate of x Mbps, the corresponding power consumption reads $b_s + a_s x$, where subscript s is for server, b_s is the baseline consumption, here equal to 325W, and a_s is the slope of the load-dependent consumption, here approximately equal to 30.6 Joule per Gigabit. In the present case, x varies between 0 and $MaxLoad$, with $MaxLoad$ equal to 1.8Gbps. Taking into account the baseline power, the energy per bit reads $a_s + b_s/x$, and is minimized when the server is fully used. In that case, it equals $a_s + b_s/MaxLoad$, which with the current parameters gives us a server cost of $\gamma_s := 211.1$ Joules per Gbit. Note that 211.1 Joules/Gbit represents the most efficient use of this VoD system. In fact this metric, which we nevertheless use in our evaluation and build a case for more efficient NaDa system, implies a data center without any stand by resources and with all servers running at 100% capacity.

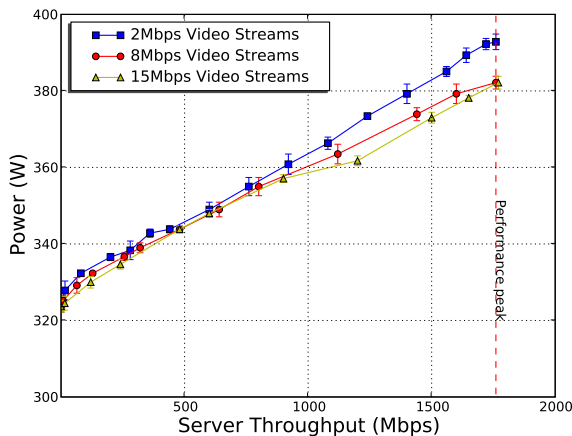


Figure 5: Power use of a video server.

Gateways We performed similar measurements on a commercial Thomson Triple Play VDSL2+ gateway used by France Telecom. We monitored TG787v model which has a Broadcom chipset, with

a 300 MHz MIPS32 processor. We uploaded video from the gateway, storage being provided either by an external hard disk, or a flash memory, connected by USB port.

Figure 6 shows the resulting power consumption for varying loads. Baseline consumption of the Gateway (in the absence of external storage) equals 13.5W. To power the USB port requires some additional power, which depends on the model of flash memory or HDD used. For example, the Flash 1GB stick increased the baseline by 0.5W. As we see from Figure 6, the additional, load-dependent power used is negligible compared to baseline. Also, as in the server case, it appears to depend linearly on load, and correlates with CPU activity.

Turning on such a gateway to serve load with Flash (254MB) storage incurs a cost of 14.5W/10Mbps, or 1450 Joules per Gigabit, considerably higher than the corresponding metric for servers. However, if we consider a gateway that is already on, its baseline consumption of 13.5W is already spent anyway. Thus the energy per bit becomes $1W/10Mbps = 100$ Joules per Gigabit, about half of the corresponding metric for servers. We shall denote this cost for serving one bit as γ_n (subscript n being for NaDa).

If we also consider next-generation Gateways with built-in flash memory, we might speculate that it is possible to eliminate the increase in baseline due to powering the USB port. Data in Figure 6 then yields an improved cost of only 18 Joules per Gigabit. Despite this opportunity, our evaluation in Section 5 is conducted with a more conservative 100 Joules/Gbit.

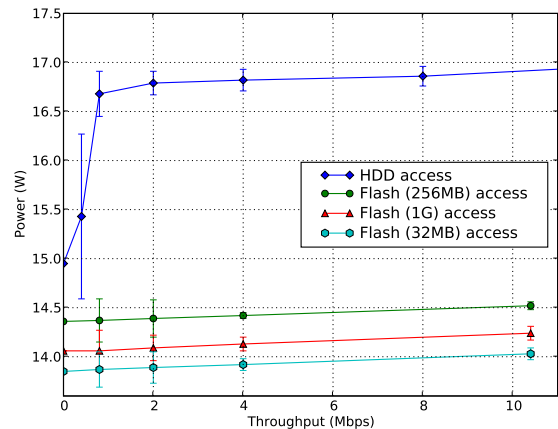


Figure 6: Power use of a DSL modem serving content.

Network To evaluate the network based power consumption, we pinpoint the cost in terms of Joules per bit incurred at each network hop, and then determine the number of hops that need to be traversed to deliver a service. We ignore non-routing equipment such as simple Ethernet switches or optical equipment. We argue that, for the same load, such equipment energy use is a fraction of routing equipment energy use, since it does not perform packet processing and route table lookups.

To evaluate energy cost per hop, we rely on the study of Chabarek et al. [10]. In this study the most efficient platform was found to be the Cisco GSR 12000 router chassis, whose power consumption was approximately 375W. A representative line card configuration used in that study achieves peak capacity of 5Gbps. We focus on this particular configuration for our evaluation. Assuming (optimistically) that routers are typically loaded at 50%, this gives a rate of 2.5Gbps. Combined with the power consumption, this gives

a ratio of $375\text{W}/2.5\text{Gbps}=150$ Joules per Gigabit. We shall denote this cost per bit as γ_r (subscript r being for router).

We now evaluate the typical number of hops involved in data transfers. We envision two distinct scenarios: 1) NaDa used across the wide-area Internet, and 2) NaDa used within an ISP. To find out distances to servers and between clients on the Internet we employed the DipZoom measurement [28]. DipZoom has about 300 measurement nodes; the clients provided by DipZoom are from North America, Europe and Asia. We grouped these clients based on the country, city or autonomous system they belong to. Table 1 shows the resulting data. The ‘‘Same DSLAM’’ line is not from DipZoom, and gives the hop count between two Gateways under the same DSLAM, assuming the DSLAM has routing capability.

The same Table 1 shows the typical distance between clients in the ISP in the same metropolitan area and distance from clients to VoD servers. We obtained these metrics from private communication with several large service providers. We recognize that it is hard to find both meaningful router efficiency metrics and perfect distance measurements that would satisfy all network scenarios. Therefore, in our evaluation section we provide NaDa multiple efficiency results for differing assumptions about the underlying network properties.

Measurement	Distance	Standard deviation
WAN: Popular servers	14.01	4.2
WAN: Clients within a country	14.09	4.01
WAN: Clients within a city	13.53	4.29
WAN: Clients within an AS	9.1	2.56
ISP: Accessing server	4	-
ISP: Accessing another client	2	-

Table 1: Number of router hops traversed between clients and servers.

3.2 Energy Consumption Model

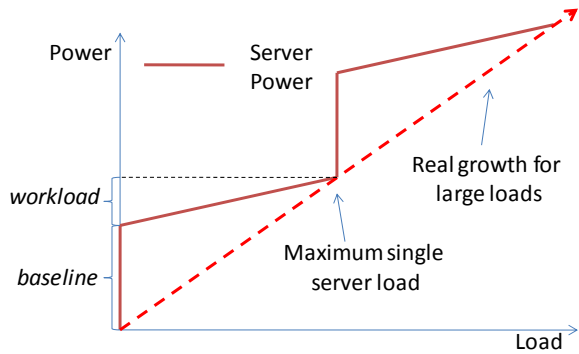


Figure 7: Energy usage slope.

Previous discussion led us to consider a linear model for the power consumption/load for all three components, namely servers, gateways and routers, characterized respectively by slopes γ_s , γ_n and γ_r . For gateways, such a linear model is motivated by the measurements in Figure 6 and the fact that in NaDa, only gateways that are already on will be used. Hence the baseline power should not be accounted, being already wasted no matter whether additional load is put on the gateway or not.

For servers, assuming that one uses the minimal number of servers that can handle the load, the actual power/load relation-

ship is given by the ‘‘staircase’’ curve on Figure 3.2. However, if servers could be put to sleep (and hence not consume energy) when idle, and process load at maximal speed when awake, then the linear model would be appropriate. Hence the linear model for servers can either be thought of as an approximation of the staircase model, or as an accurate description of servers with efficient sleep mode.

Using the linear model for routers is harder to defend, as a router needs to remain powered, even when forwarding data at a low rate. An alternative assumption consists in assuming that router power consumption is load-independent, and thus savings can only come from shifting load from servers to gateways. We shall indeed consider energy savings under this assumption. However if network load is reduced, then network capacity upgrade can be delayed, and long-term energy consumption of network components can be reduced accordingly. We shall thus also evaluate energy savings with the linear model for routers.

Apart from power consumption of individual components, the overall system efficiency is also affected by power conversion, power loss, and cooling costs. Data centers, CDN networks and routers incur cooling and power conversion overhead; to account for it, we inflate the corresponding power numbers by the PUE factor defined earlier. Recent reports [4, 19] indicate that PUE of 2 is common, while PUE of 1.2 is the best to date, and unlikely to be improved much upon. To be fair to data center architecture, we assume that conventional equipment is subject to a PUE of 1.2.

Thus for a given load x expressed in terms of bandwidth, we propose the following expression for the corresponding power consumption in data centers, that we denote $E_{dc}(x)$:

$$E_{dc}(x) = x * PUE(\gamma_s + h_{dc} * \gamma_r) \quad (1)$$

In the above, h_{dc} is the average number of hops necessary to reach the client in a conventional, centralized system. In view of Table 1, we evaluate both $h_{dc} = 14$ and $h_{dc} = 4$. We may alternatively remove the term reflecting networking equipment, $h_{dc}\gamma_r$ if we want to account only for energy used in servers and gateways.

In the case of NaDa, gateways, unlike routers and servers, do not require cooling. Nevertheless they do incur overhead costs for energy transportation, distribution and conversion. Surveys [1, 2] indicate that such losses amount to 7.4% and 6.2% of the delivered energy respectively for the US and the UK. When evaluating energy consumption at gateways, we thus inflate our initial evaluation by a ‘‘loss’’ factor ℓ that we take equal to 1.07.

Thus for a given load x in bandwidth, the corresponding power consumption in NaDa, denoted $E_n(x)$, is:

$$E_n(x) = x * (\ell * \gamma_n + PUE * h_n * \gamma_r), \quad (2)$$

where h_n is the average number of hops necessary to reach the client inNaDa, which in view of Table 1, could be taken as $h_n = 9$ or $h_n = 2$ for a WAN or an ISP scenarios respectively.

3.3 Back-of-the-envelope calculations

The expressions (1,2) allow us to assess potential benefits from NaDa for content access services. Specifically, by using the proposed values $\gamma_s = 211.1\text{J/Gb}$, $\gamma_n = 100\text{J/Gb}$, $\gamma_r = 150\text{J/Gb}$, $PUE=1.2$, $\ell=1.07$, $h_{dc} = 14$, and $h_n = 2$, summarized in Table 4 we find:

$$\frac{E_n(\cdot)}{E_{dc}(\cdot)} = \frac{1.07 * 100 + 1.2 * 9 * 150}{1.2(211.1 + 14 * 150)} \sim 59\%. \quad (3)$$

Thus NaDa potentially reduces power costs by 41%. Similarly, we may compare NaDa efficiency to that of Content Delivery Networks (CDNs). For CDNs we would use the same formula (1) as

for data centers, except that we would set the hop count term to $h_{cdn} = 4$ in view of Table 1. The corresponding expression, that we may denote E_{cdn} , would yield the following evaluation:

$$\frac{E_n(\cdot)}{E_{cdn}(\cdot)} = \frac{1.07 * 100 + 1.2 * 2 * 150}{1.2(211.1 + 4 * 150)} \sim 38\%. \quad (4)$$

Thus we may expect energy savings of the order of 62% in NaDa compared to CDNs.

When we consider only gateway and server energy, by removing the network component in the above ratios, we arrive at a value of $1.07 * 100 / (1.2 * 211.1) \sim 42\%$, thus savings on data centers only (i.e. network excluded) of potentially 58%.

The above values are idealized values of course. To quantify more precisely a concrete application scenario, e.g., NaDa-based VoD, many more parameters need to be considered including the memory used at Gateways, the placement strategies, and the popularity profile of content. We perform an in-depth evaluation of these impacts in the next section.

4. VIDEO-ON-DEMAND SERVICE

NaDa platform is ambitious in the sense that it could eventually host various third party distributed applications (e.g. distributed games, file sharing, social networks, web hosting, etc). However, designing applications to run on such highly distributed environment is non-trivial (security concerns, data consistency and partitioning, etc) and thus it is outside of the scope of this paper. To highlight the benefits of NaDa in terms of energy savings, instead we focus on one particular application: a distributed VoD system. Video content distribution will likely be the main driver of Internet traffic growth in the next 5 years [13].

Results on the energy consumption of VoD delivered from NaDa, evaluated via event-driven simulations, are presented in Section 5. In this Section we set the stage by describing the system architecture, the data sets, the content placement method and the simulation setup used in the evaluation.

4.1 VoD Service Architecture in NaDa

The fundamental element in NaDa is the gateway. In addition, a *tracker* manages gateway resources. Finally, *content servers* take care of ingesting content in the system, and provide a fall-back option under shortage of gateway resources. We now detail further the roles of these three key components:

Gateways provide storage, and bandwidth resources. In the case of VoD, they store full or partial replicas of video content objects, and provide uplink bandwidth to deliver these objects to other gateways. NaDa gateways have two separate virtual circuits on their upstream line with different capacities allocated to each: one virtual circuit is dedicated to conventional Internet use while the other is allocated for NaDa use. Such setting is challenging to achieve on cable networks, but readily available with DSL and fiber-to-the-home technologies. In the case of DSL service providers can separate and contain the traffic from gateways using different VPI/VCI designators, while in the case of FTTH networks ISPs can use VLANs or GPON compatible technologies. In our evaluation we assume gateways with 1Mbps to 2Mbps upstream speeds.

Current home gateways today are (almost) always on, making efficient use of baseline energy an easy target. In the future however, we envision that gateways will have the ability to enter so called “sleep mode”. Such gateways operate as usual when they are in active use and switch to a sleep mode to preserve power when no user activity is present. While in practice the gateways can be active due to normal Internet use, in the most pessimistic case for NaDa

the gateways can serve the content only when the same gateway is used to retrieve some other content. As we shall see, even in such a pessimistic scenario we can achieve up to 15-30% energy savings.

It may be necessary to provide incentives for home users to host NaDa gateways, especially since the required power is paid for by the users. This could be achieved through “miles” or service discount schemes introduced to reward users. However, as we can see in the Figure 11, the typical daily load on a gateway is about 4000 megabits of traffic which translates to an additional 2.5 kW/hour a year. Given the price range of 10-20 cents per kW-hour, this additional energy cost to a single home user is not significant.

The tracker coordinates all VoD-related activities in NaDa. It monitors the availability and content possession of gateways, answers requests for content by lists of gateways holding the desired content. It is then up to the requesting gateway to download movie parts from other gateways. Such downloads are performed at a rate equal to the minimum of the spare uplink bandwidth and the streaming rate; if spare uplink bandwidth is less than streaming rate, other gateways are contacted in parallel. When the total download rate from gateways is less than the streaming rate, content servers provide the missing data at the desired speed.

The tracker also has the role to inform gateways of suitable content updates to perform. Such content updates are performed at low network utilization periods, possibly by using multicast. A cached content copy at the gateway can be retrieved numerous times thus rendering content update costs marginal.

Content servers provide the content from legacy data centers or caches. Content servers can belong to the entity that manages the NaDa platform, or to content providers. Their primary role is to pre-load gateways in offline mode with content that they can subsequently re-distribute. Content servers can also serve content requests online if no gateway can treat the request.

We evaluate NaDa platform by performing content placement and access simulation in a large metropolitan area served by one PoP and a few thousands of users. As a consequence, each user is assumed to have identical network distance to every other user in a network (this is what would happen on a mid-sized metropolitan area network). Similarly every user has the same distance to the content servers. We describe the data sets we use to drive the simulations and then proceed to the VoD placement and serving algorithms.

4.2 Datasets

We use content access traces from three sources (see Table 2): (1) Netflix movie database, (2) Youtube traces [8], and (3) IPTV access statistics from a large ISP [9]. Netflix movie database contains the number of rentals for each movie - we interpret a rental as a content access. For each movie in Netflix database, we assume the same duration, namely 90 minutes. Youtube traces contain both view count and the length of each video object. The IPTV statistics represent a service with approximately 250,000 customers and contains anonymized 60 day-long trace. We take a random sample of 2,000 users and we treat each TV program as a separate object. While today IPTV programs are mostly streamed to users using multicast, service evolutions such as *catch-up TV* will make IPTV look more and more as another VoD service. Finally we use a *mixed* trace which is an equal mix of all the content in the three traces mentioned above.

Figures 8 and 9 show length and popularity distributions for the traces. Netflix is not plotted in the length figure as all movies are assumed to have the same length. Length of IPTV content is almost constant, while Youtube content duration is more variable. The second dip in the Youtube curve reflects the 10 minute limit imposed

Trace	Objects	Avg. len (s)	Avg. views
IPTV	65,486	2246.2	2127.9
Youtube	1,686,264	166.5	2191.1
Netflix	17,770	5400	6555.5

Table 2: Content trace properties.

on uploaders. Some privileged users, however, can upload longer movies.

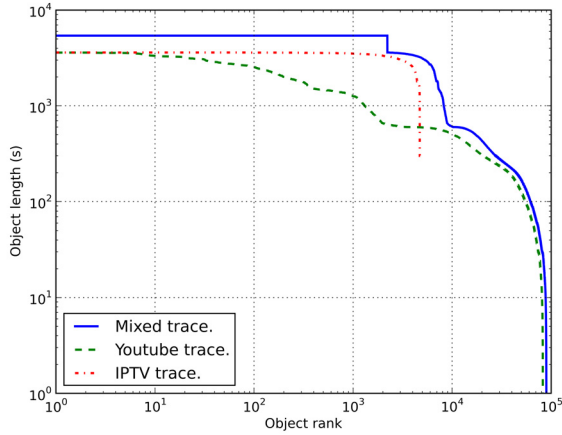


Figure 8: IPTV and Youtube content length.

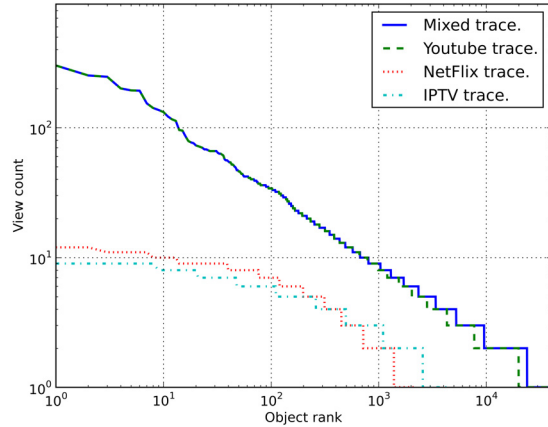


Figure 9: Content view counts.

Figure 9 represents content popularity, measured in view counts. The heaviest tail is observed in the Youtube content, while IPTV and Netflix offer slightly more uniform, but still highly skewed popularity distributions.

We perform two transformations to our content traces before we feed them into the simulator. First, we normalize the content to match the number of users in the simulated network. We use the data collected by Cha et al. [9] to determine the average amount of time users spend daily watching IPTV, which we find to equal to approximately 100 minutes. We then multiply this average by the number of users in our population to get the aggregate number

of content-hours. We then use this aggregate number of hours as a target to scale down our original traces. Once the number of accesses per item in the trace is normalized, we spread the accesses throughout the span of a single day to reproduce the daily use pattern observed in Figure 4.

4.3 Content placement strategy

The content must to be split into smaller chunks called *data windows* to make it more manageable. We use the following convention for content formatting in NaDa system. Each movie is split into data windows, whose length is a parameter w that we let vary from 10 to 120 seconds. The number of windows constituting a movie is then the ratio of movie duration by parameter w .

We perform placement by first determining, for each movie f , the number of movie replicas n_f we target to store overall on gateways. This in turn gives us the number of data windows we store in the system: if n_f is an integer, each window is replicated exactly n_f times. For fractional n_f , a subset of windows is replicated $\lfloor n_f \rfloor$ times, while other windows are replicated $\lfloor n_f \rfloor + 1$ times, in order to have approximately n_f times the full movie size replicated overall. Given the number of replicas of each window of content, we then place these at random on gateways, under the constraint imposed by memory resources at each gateway.

Optimization formulation. We now explain the rationale we used to determine the replication number n_f of each movie f . We assume that information is available on movie popularity, captured by the average number ρ_f of simultaneous viewings of movie f during the peak hour. The latter could be defined by the 95-th percentile, say, of the daily distribution of viewings. Based on the diurnal activity patterns in Figure 4, we would take the peak load equal to three times the average load over a day.

Let n_f denote the number of copies of movie f stored in total. We phrase the problem of selecting such parameters n_f as an optimization problem, whose objective is to minimize the number of requests that must be served by the infrastructure. To this end, we list some constraints that are natural to the problem. First, if n_f is allowed to be less than 1 (i.e. only a subset of the original movie is stored within NaDa), then the number of simultaneous requests, x_f , that can be served from gateways, must satisfy

$$x_f \leq \rho_f n_f. \quad (5)$$

Indeed, each one of the ρ_f requests can't receive from gateways more than the amount n_f stored there in total.

Second, denote by L_f the length of movie f in seconds. Then for an upstream bandwidth of u and a streaming rate of r , the number x_f of simultaneous requests that can be served from gateways must satisfy

$$x_f \leq (u/r) * (n_f L_f / w). \quad (6)$$

Indeed, the first term u/r reflects the fact that a window of content stored on a given box can be streamed at rate r simultaneously to at most u/r downloaders. Now, the number of boxes holding such a window of content is precisely n_f times the number of windows per replica of the movie, that is L_f/w .

Another effect we account for is the following. A fraction n_f/N of the content needed for the ρ_f viewings is available on the gateway of the user requesting the content, where N is the total number of gateways in the system. The residual number of requests that need to be served from either other gateways, or infrastructure servers, is given by

$$\rho'_f = \rho_f (1 - n_f/N). \quad (7)$$

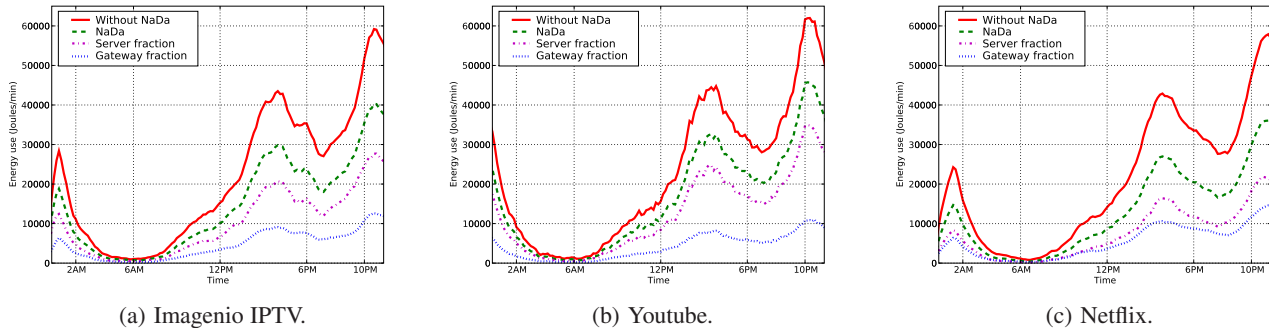


Figure 10: Representative diurnal patterns of energy use. NaDa energy use is further split into server and gateway components. Energy is accounted in 1 minute intervals.

Our optimization formulation is then the following:

$$\text{Minimize } \sum_{f \in \mathcal{F}} \max(0, \rho'_f - x_f) \quad (8)$$

where ρ'_f is given by (7). Minimization is over the non-negative variables n_f , x_f , for all f in the whole catalogue \mathcal{F} , under constraints (5), (6), and the following:

$$\sum_{f \in \mathcal{F}} n_f L_f \leq sN \quad (9)$$

where s denotes the storage of each gateway, to which we add a constraint on total usage of uplink bandwidth:

$$\sum_{f \in \mathcal{F}} x_f \leq N(u/r). \quad (10)$$

The “hot-warm-cold” placement method. The optimization problem (5-10) is a standard linear program. It can be shown using elementary properties of linear programming that an optimal solution has the following structure.

Assume that movies f in the catalogue \mathcal{F} are sorted by decreasing popularity normalized by duration, i.e. $\rho_1/L_1 \geq \rho_2/L_2 \dots$. Then movies should be partitioned into three groups: the most popular f_1 movies, constituting “hot” content, should be replicated on all gateways. The subsequent f_2 most popular movies, constituting “warm” content, are replicated minimally so that all their requests can be served from gateways, that is:

$$n_f = \max \left(1, \frac{\rho_f}{N^{-1}\rho_f + (u/r) * (L_f/w)} \right). \quad (11)$$

Finally, the less popular movies, constituting “cold” content, are not stored within NaDa. The sizes f_1 and f_2 of the “hot” and “warm” groups are determined so that constraints (9-10) are met with equality, where x_f is set equal to ρ'_f .

We now comment on general properties of this placement strategy. When storage is scarce, the “hot” group can vanish; at the other extreme, with massive storage most content ends up being replicated everywhere.

Concerning the “warm” content, for small window size w relative to movie length L_f , constraint (6) becomes irrelevant, and as a result the optimal number of replicas n_f in (11) ends up equal to 1. That is, a single copy is stored. While this may seem counter-intuitive, numerical evaluations to be presented next will confirm that this suffices to serve most requests.

Parameter	Values	Notation
Network size	1,000-30,000 users	N
Upstream bandwidth	0.1-2Mbps	u
Streaming rate	0.1-8Mbps	r
Gateway storage	100-128,000MB	s
Window length	10-120sec	w
Simulation duration	86400sec	d
Peak to average ratio	3	p

Table 3: Simulation parameters.

Note finally that in the case of gateways with efficient sleep mode, the above placement strategy should be adjusted as follows. If at peak hour a fraction ϵ of gateways is expected to be on, then in Equations (5–7) variable n_f should be replaced by ϵn_f . The corresponding optimal placement strategy has the same structure as before, the expression for n_f in (11) being scaled up by the factor $1/\epsilon$.

4.4 Simulation environment

We implement event-driven simulator to investigate NaDa performance and dimensioning for VoD services. The simulations consist of two phases: (1) object placement which is performed offline, according to the mechanisms described in the previous subsection; and (2) trace-driven simulation using our IPTV, Netflix, YouTube and mixed traces, transformed to reflect user video watching time and diurnal variations as we previously explained.

Each simulation represents 24 hours of user activity. We vary NaDa system parameters within the range given in Table 3. We limit the number of users to 30,000 in in our largest simulations order to make the simulations scalable. All other values have been chosen to match realistic operational settings. The output of the simulation is then used to calculate energy consumption, using the models of Section 3 and the parameters listed in Table 4. We set hop count to 2 for gateway-to-gateway communication, and 4 for gateway-to-server communication in ISP scenario; and we set counts to 9 and 14 for WAN scenario. These parameters are summarized in Table 4.

5. RESULTS

Unless otherwise noted, in this section we run simulations using video streaming rate of 2Mbps, a network of 6000 Gateways, each equipped with 8GB storage, 1Mbps upstream and hosting content with window size of 60 seconds. The default workload, if none

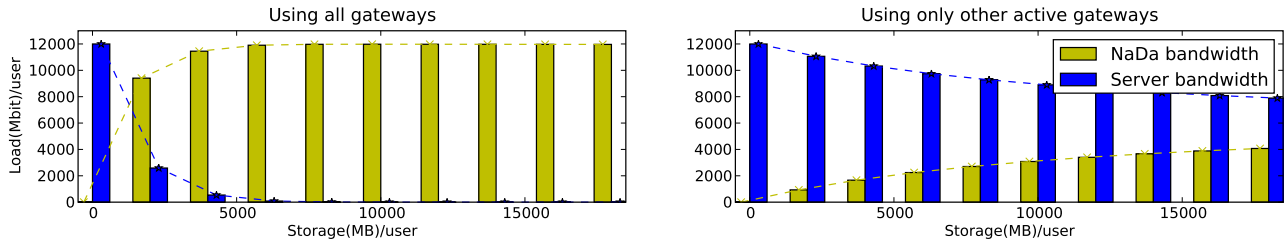


Figure 11: Bandwidth to the servers and to the NaDa system as we increase storage in each Gateway. Left graph represents bandwidth using legacy Gateways, right graph represents a pessimistic scenario for Gateways with sleeping support.

Parameter	Value	Notation
Hops to servers	ISP:4, WAN:14	h_{cdn}
Hops between clients	ISP:2, WAN:9	h_n
Hop(router) energy/bit	$150 * 10^{-9} J$	γ_r
Server energy/bit	$211 * 10^{-9} J$	γ_s
NaDa energy/bit	$100 * 10^{-9} J$	γ_n
Data center PUE	1.2	PUE
Home energy PUE	1.1	ℓ

Table 4: Parameters for energy computation.

specified, as achieved by using a mixed trace from our three data sources. We also use router distances from an ISP network as described in Table 1. All the traces in the simulations are normalized to represent a daily population as described in Section 4.2. We provide results both for legacy Gateways (Figures 10, 11 and 12), expected to be always on, and for Gateways with efficient sleep mode (Figures 11, 13 and 14).

5.1 Effects of daily usage patterns

Figure 10 illustrates energy consumption pattern throughout the day for three different content traces. The top curve in Figure 10 represents the energy used when relying on a traditional Data Center. Going down, the second line shows the energy consumed with NaDa. The bottom two plots show the fraction of NaDa energy due to the requests directed to infrastructure and gateways respectively.

We first observe that benefits of NaDa are larger at high activity periods. At peak time, using default settings described above, NaDa saves around 30% of the legacy Data Center energy. Figure 10 also shows how NaDa can smoothen the load on the network core by reducing its peak use. While hard to quantify, such smoothing enables to save on investments by delaying upgrades of network capacity and equipment.

A second observation is that the benefits of NaDa change with the characteristics of the content. Savings are the smallest for the Youtube trace. This may sound counter-intuitive, as Youtube has the most skewed popularity distribution of the three traces (i.e. relative popularity of popular content is higher than for IPTV or Netflix). Skewness is advantageous to NaDa, as it increases the predictability of which content is more likely to be accessed. However, Youtube also has the largest catalogue, and requests for “cold” content in Youtube more than offset requests for “hot” content.

5.2 Proportion of “on” Gateways

NaDa makes use of storage and uplink bandwidth of active Gateways. As discussed in Section 2, legacy Gateways are (almost) always on, and thus always available as NaDa resources. If we transition to Gateways with efficient sleep mode (we shall refer to

these as “sleeping Gateways”), fewer resources will be available to NaDa. At the very least, Gateways of customers requesting content must be on, and thus NaDa capacity scales with demand.

Figure 11 shows side to side the efficiency of NaDa system with legacy and with sleeping Gateways in terms of load served, as we vary the storage per Gateway. In a legacy scenario, NaDa handles all demand with as little as 6GB of storage per Gateway. NaDa with sleeping Gateways has lower bandwidth efficiency and takes 18GB of storage per Gateway to attract approximately 1/3 of the total load. Additional storage can in principle lead to all demand being served by NaDa (e.g. when all content ends up replicated everywhere), but in practice efficiency increases very slowly with additional storage above 18GB.

This feature stems from the fact that with our default settings, video streaming rate (2Mbps) is larger than upstream rate (1Mbps). When the upstream to video streaming rate ratio is larger than one, far less storage is needed for all load to be diverted from Servers to Gateways. This can be seen on Figure 14, which shows overall energy savings with sleeping Gateways and various choices of upstream and video streaming rates.

5.3 Storage and window size

While Figure 11 shows bandwidth use in NaDa, Figure 12 translates such bandwidth to overall energy consumed per user. Zero storage corresponds to traditional system without NaDa support. As expected, additional storage decreases energy use, down to the point where the benefit of additional storage becomes negligible. This happens at around 4GB for all three traces, in the considered setup of legacy always-on Gateways.

Further increases in efficiency are driven by availability of local copies of content. Note that for Youtube content, energy consumption decreases more slowly than for the other traces because of the relatively huge size of the Youtube catalogue. Nevertheless, 4GB of memory is small and affordable, and achieves a 55% savings with legacy Gateways.

We now discuss the impact of window size variation (introduced in Section 4.3) on the efficiency of NaDa. Figure 12 displays a “perfect” split curve for the mixed dataset. Perfect split here means that the content is divided into infinitely small chunks. Such splitting results in an idealized system where the collection of Gateways behaves as a single server with aggregate upstream capacity of $u * N$, equal to the sum of Gateway uplink bandwidths. Indeed, contention at the individual Gateway level decreases with window size, since the smaller the window, the less likely it is that distinct requests compete for the same Gateway’s uplink bandwidth.

As can be seen in Figure 12, the energy consumption for a window size of 60 seconds follows closely the energy consumption of the idealized system. From the viewpoint of bandwidth utilization, window sizes of 60 seconds are already small enough for an effi-

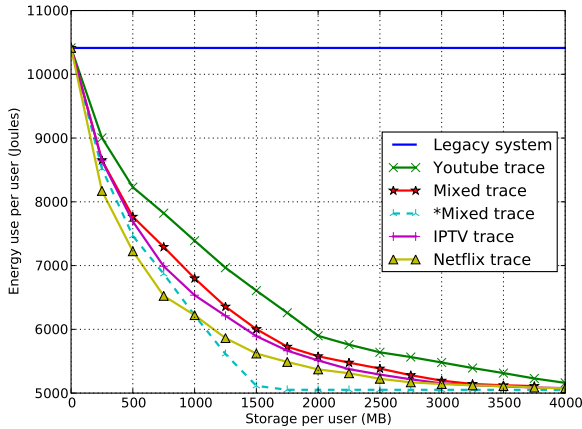


Figure 12: Energy usage with legacy Gateways as a function of storage. *Mixed trace represents a “perfect splitting” with infinitely small window size.

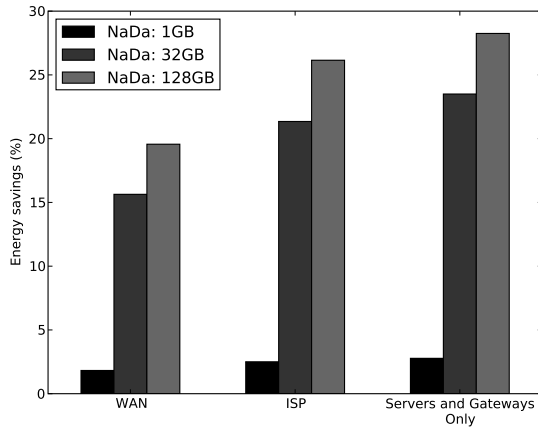


Figure 13: Energy savings for NaDa with sleeping Gateway support using different assumptions about the underlying network.

cient operation. While smaller windows may have other advantages – for instance reduced start-up delay – they introduce management overhead at the tracker and at the Gateways.

5.4 Network and population size

We now discuss the impact of network hop count and user population size on savings. Figure 13 shows NaDa efficiency with different assumptions about the underlying network.

WAN scenario assumes 14 router hops to a server and 9 router hops between two clients. ISP scenario assumes 4 router hops to the server and 2 router hops between clients. The last scenario in Figure 13 measures NaDa efficiency only as it applies to servers and Gateways. We observe that the introduction of the network into the equation results in reduced overall relative savings. However, even in the most adverse scenario, we achieve more than 20% energy savings.

Energy savings also occur as we increase user population participating in NaDa. Figure 14 shows how energy efficiency grows as

we increase the number of users. We observe that with 9000 users, video rate of 0.7Mbps and upstream rate of 1Mbps NaDa achieves 39% energy savings. As we increase the network size to 27000 users the energy savings grow to 44%. As more Gateways participate as clients, more Gateways can act as servers, thus increasing the likelihood of finding needed content.

In the legacy system the savings with larger population size would grow faster. With limited Gateway storage, NaDa performance is *memory-bound* rather than *bandwidth-bound* (at least for small populations). As population grows, the total memory available to the system increases, and the corresponding memory limitation is relaxed.

5.5 Streaming and uplink rates

Each set of three groups of bars in Figure 14 shows energy efficiency as a function of video streaming rate and upstream bandwidth available to Gateway. Energy savings increase steeply until we hit the point where the upstream rate is equal to the streaming rate. For example, for 18,000 users switching from 1Mbps to 2Mbps service at video rate of 2Mbps increases savings from 15% to 25%. On the other hand switching from 1Mbps to 2Mbps at a video rate of 8Mbps improves savings only insignificantly. With upstream rate less than streaming rate, individual Gateways serve windows of content at a rate below the streaming rate. One thus needs to download content from several Gateways simultaneously if one wants to obtain it fast enough from NaDa. This causes more contention for Gateway uplink bandwidth, and hence reduces the efficiency at which Gateway uplink bandwidth is used.

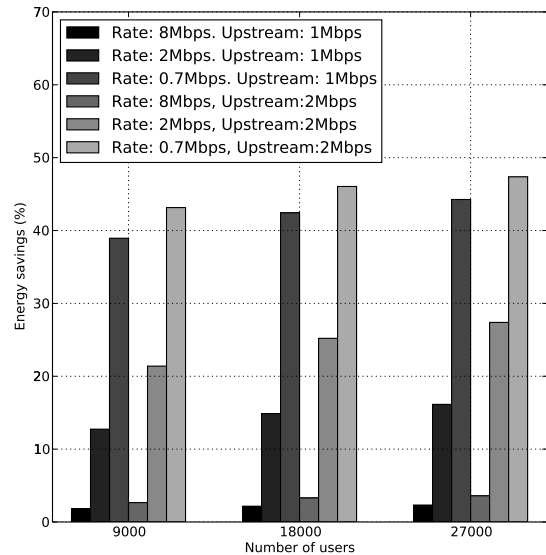


Figure 14: Energy savings for NaDa with sleeping Gateway support using different population sizes with varying video and upstream rates.

6. RELATED WORK

Component level energy savings in the IT equipment is an active research area, with ideas ranging from dynamic voltage Scaling [29] and memory power management [15], to operating system scheduling [21]. A holy grail of efficient energy use is a *energy*

proportional computing [7], where the overall energy consumption of a system is linearly proportional to the load. Achieving this goal would eliminate the waste of baseline power in servers that are not 100% utilized. This objective is strongly related to our goal of leveraging baseline power wasted today in Gateways. Energy proportionality is a challenging objective and it remains to be seen how quickly it will move forward. Until then, new energy-aware distributed systems techniques like the one put forth by NaDa will be needed for keeping energy consumption under control.

Still at the scale of individual components, the authors of [17] address a networking issue, namely energy consumption in LAN Ethernet cards. They make interesting proposals, suggesting to use proxying and splitting to put LAN cards to sleep without losing existing TCP connections.

At a larger scale, one of the early works to discuss aggregate network consumption was a 2004 position paper on “Greening of the Internet” [18] that coined the idea of energy efficiency of Internet components and protocols. More concrete subsequent works include [10] and [25]. The first one looks at energy gains by reconfiguring Internet topologies either dynamically through routing or earlier at the stage of network design. The second proposes delaying batches of traffic at buffers at the network edges so as to allow time for core network components to sleep and save energy. The above works all approach energy savings essentially at the network layer and below, and thus differ substantially from our application-level approach in NaDa.

One of the early works on energy-efficient Data Center designs is [11] whereas more recent results and an overview of the current state of research efforts can be found in [16]. These works take the monolithic paradigm of Data Centers for granted and look at how to improve its energy efficiency. Our work has suggested that there are fundamental limitations inherent to the paradigm itself and has proposed the NaDa approach as an alternative or a partial substitute to monolithic Data Centers.

To the best of our knowledge the most directly related works to ours are two 2008 position papers [20] and [12]. Both of these works coin the idea of breaking and distributing monolithic Data Centers. The first one goes all the way to suggesting using edge devices like in NaDa, whereas the second one stays at the coarser granularity of container-sized mini Data Centers. Our work is aligned with these approaches and extends them by providing a thorough quantification of gains based on concrete application scenarios and with the use of extensive measurement data.

7. DISCUSSION AND CONCLUSION

We have introduced NaDa, a new communication architecture for Internet content and service delivery. In NaDa, content and services are stored on home gateways instead of data centers. The access to these services on gateways is provided by using a managed P2P infrastructure. NaDa greatly reduces the need for content servers and network resources. We use real-life service load traces and extensive simulations to show that NaDa can save at least 20-30% of the energy spent by legacy data centers. Our models show that each gateway requires just modest memory upgrades.

A number of questions need to be answered before NaDa can be effectively deployed. First, users may be concerned that the ISP takes advantage of their own electricity to offer services to others, and may need to be “incentivized” to accept NaDa. Simple rewards schemes could be used to this end, motivating users through awareness programs or rewarding them with service credit equivalent to their energy use increase.

A second issue is that of uplink bandwidth and Gateway capacity. NaDa critically relies on these resources, and it requires substan-

tial investment from an ISP to secure the necessary uplink bandwidth, and to deploy powerful enough Gateways. However, from discussions with ISPs, these investments could indeed be made: the Gateway is strategically located as the hub to the Internet, and ISPs may want to leverage this strong position to offer home networking services. ISPs currently providing triple-play services are in fact considering deployment of powerful Gateways with memory and multicore processors in order to provide more services. In addition, FTTH deployment is making progress and removes the uplink bandwidth limitation, as well as simplifies the network architecture (while DSLAMs can theoretically perform routing tasks, most of the gateway-to-gateway communication as of today still has to go up to the router to which the DSLAM is connected).

We therefore believe that NaDa not only can, but also should happen, given its potential. We are currently working on prototypes of advanced Gateways and on fine-tuning the tracker and placement mechanisms to deliver VoD services.

8. REFERENCES

- [1] <http://climatetechnology.gov/library/2003/tech-options/tech-options-1-3-2.pdf>.
- [2] <http://www.powerwatch.org.uk/energy/graham.asp>.
- [3] Data Center Report to Congress. http://www.energystar.gov/ia/partners/prod_development/downloads/EPA_Datacenter_Report_Congress_Final1.pdf.
- [4] Google Data Center Power Usage Efficiency. <http://www.google.com/corporate/datacenters/measuring.html>.
- [5] Thomson’s sapphire server data sheet. [http://www.thomsongrassvalley.com/docs/DataSheets/tbm/sapphire\\$_\\$_broadcast/CDT-3039D-2.pdf](http://www.thomsongrassvalley.com/docs/DataSheets/tbm/sapphire$_$_broadcast/CDT-3039D-2.pdf).
- [6] Uptime Institute. <http://www.uptimeinstitute.org>.
- [7] L. A. Barroso and U. Hölzle. The case for energy-proportional computing. *Computer*, 40(12):33–37, 2007.
- [8] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon. I tube, you tube, everybody tubes: analyzing the world’s largest user generated content video system. In *IMC ’07: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 1–14, New York, NY, USA, 2007. ACM.
- [9] M. Cha, P. Rodriguez, J. Crowcroft, S. Moon, and X. Amatriain. Watching television over an ip network. In *IMC ’08: Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*, pages 71–84, New York, NY, USA, 2008. ACM.
- [10] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsang, and S. Wright. Power awareness in network design and routing. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 457–465, 2008.
- [11] J. S. Chase, D. C. Anderson, P. N. Thakar, A. M. Vahdat, and R. P. Doyle. Managing energy and server resources in hosting centers. In *Proc. of ACM SOSP ’01*, pages 103–116, 2001.
- [12] K. Church, A. Greenberg, and J. Hamilton. On delivering embarrassingly distributed cloud services. In *Proc. of HotNets-VII*, 2008.
- [13] Cisco Visual Networking Index. <http://newsroom.cisco.com/visualnetworkingindex/>.
- [14] M. Dischinger, A. Haebleren, K. P. Gummadri, and S. Saroiu. Characterizing residential broadband networks. In *IMC ’07: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 43–56, New York, NY, USA, 2007. ACM.
- [15] X. Fan, C. Ellis, and A. Lebeck. Memory controller policies for dram power management. In *ISLPED ’01: Proceedings*

- of the 2001 international symposium on Low power electronics and design, pages 129–134, New York, NY, USA, 2001. ACM.
- [16] X. Fan, W.-D. Weber, and L. A. Barroso. Power provisioning for a warehouse-sized computer. In *ISCA '07: Proceedings of the 34th annual international symposium on Computer architecture*, pages 13–23, New York, NY, USA, 2007. ACM.
- [17] C. Gunaratne, K. Christensen, and B. Nordman. Managing energy consumption costs in desktop pcs and lan switches with proxying, split tcp connections, and scaling of link speed. *Int. J. Netw. Manag.*, 15(5):297–310, 2005.
- [18] M. Gupta and S. Singh. Greening of the internet. In *SIGCOMM '03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 19–26, New York, NY, USA, 2003. ACM.
- [19] J. Hamilton. Where does power go in high-scale data centers? *Tutorial, ACM Sigmetrics*, 2009.
- [20] N. Laoutaris, P. Rodriguez, and L. Massoulié. Echos: edge capacity hosting overlays of nano data centers. *ACM SIGCOMM Comput. Commun. Rev.*, 38(1):51–54, 2008.
- [21] J. R. Lorch. *Operating systems techniques for reducing processor energy consumption*. PhD thesis, 2001. Chair-Smith, Alan Jay.
- [22] S. McAllister, V. P. Carey, A. Shah, C. Bash, and C. Patel. Strategies for effective use of exergy-based modeling of data center thermal management systems. *Microelectron. J.*, 39(7):1023–1029, 2008.
- [23] J. C. Mogul. Improving Energy Efficiency for Networked Applications. Dec. 2007. Keynote Address at ACM/IEEE ANCS 2007.
- [24] S. Nedeveschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall. Reducing network energy consumption via sleeping and rate-adaptation. In *NSDI'08: Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*, pages 323–336, Berkeley, CA, USA, 2008. USENIX Association.
- [25] S. Nedeveschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall. Reducing network energy consumption via sleeping and rate-adaptation. In *NSDI'08: Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*, pages 323–336, Berkeley, CA, USA, 2008. USENIX Association.
- [26] A. Sinha and A. P. Chandrakasan. Energy efficient real-time scheduling. In *ICCAD '01: Proceedings of the 2001 IEEE/ACM international conference on Computer-aided design*, pages 458–463, Piscataway, NJ, USA, 2001. IEEE Press.
- [27] K. Suh, J. F. Kurose, L. Massoulié, D. Towsley, M. Varvello, C. Diot, and C. Neumann. Push-to-peer video-on-demand system: design and evaluation. Technical report, In UMass Computer Science Technical Report 2006-59, 2006.
- [28] Z. Wen, S. Triukose, and M. Rabinovich. Facilitating focused internet measurements. In *SIGMETRICS '07: Proceedings of the 2007 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pages 49–60, New York, NY, USA, 2007. ACM.
- [29] B. Zhai, D. Blaauw, D. Sylvester, and K. Flautner. Theoretical and practical limits of dynamic voltage scaling. In *DAC '04: Proceedings of the 41st annual conference on Design automation*, pages 868–873, New York, NY, USA, 2004. ACM.