RC25462 (IRE1404-021) April 30, 2014
Computer Science

IBM Research Report

Clome: The Practical Implications of a Cloud-Based Smart Home

Zubair Nabi
IBM Research
Smarter Cities Technology Centre
Mulhuddart
Dublin 15, Ireland

Atif Alvi
Forman Christian College
Lahore, Pakistan

Clome: The Practical Implications of a Cloud-based Smart Home

Zubair Nabi
IBM Research, Dublin
zubairn@ie.ibm.com

Atif Alvi
Forman Christian College, Lahore, Pakistan
atifalvi@fccollege.edu.pk

Abstract

A rich body of work in recent years has advocated the use of cloud technologies within a home environment, but nothing has materialized so far in terms of real-world implementations. In this paper, we argue that this is due to the fact that none of these proposals have addressed some of the practical challenges of moving home applications to the cloud. Specifically, we discuss the pragmatic implications of moving to the cloud including, data and information security, increase in network traffic, and fault tolerance. To elicit discussion, we take a clean-slate approach and introduce a proof-of-concept smart home, dubbed Clome[1], that decouples non-trivial computation from home applications and transfers it to the cloud. We also discuss how a Clome-like smart home with decentralized processing and storage can be enabled through OpenFlow programmable switches, home-centric programming platforms, and thin-client computing.

1 Introduction

Today's home is a microcosm of personal computers, smart phones, embedded devices, networked appliances, and sensors. All of these rely on highspeed broadband and wireless connectivity. Highspeed Internet has also made rich multimedia services such as Apple TV [2] possible. These living spaces, which aim to improve quality of life are broadly categorized as *smart homes*. Smart homes invisibly perform tasks and functions that would otherwise need to be performed manually by the users.

With the widespread adoption of the vision of "Internet of Things" [17], the number of connected devices is increasing exponentially. We believe that the concept of "Internet of Things" blends well with that of a smart home – the home is the best place to start due to its device diversity. The growing use of HTTP in the house [29] both by PCs and other embedded and mobile devices [22] is a proof of this trend. All of this is in line with the vision of ubiquitous computing but a number of outstanding issues need to be resolved before smart homes can become truly pervasive. The existing home set-up is plagued by many problems such as inflexibility and poor manageability of home automation systems [18]. Additional problems include high energy consumption and heat dissipation, device heterogeneity, and poor manageability. Adding more devices will only exacerbate these problems.

Similarly, home networks have their own set of problems. Provisioning, complexity, troubleshooting, security, and composition problems can be attributed to the use of the end-to-end principle in these networks [19]. The absence of network administrators in the home can also lead to *misconfiguration* of the network [11]. Further, home networks are also susceptible to spam, *DoS* traffic, and/or scam or phishing attacks [23]. A number of solutions to these problems have been proposed [19, 11, 23] but each one of them requires dedicated resources and intensive computation in some cases.

Concurrently, researchers have started thinking of the home as a platform with its own Appstore to serve applications ranging from security systems to light controllers [21]. In the same vein, software defined networking (SDN) has also opened up a market for networking applications ranging from QoS to identity management [32]. But there is still no common platform and application store to house all of these efforts under one roof.

The emergence of cloud computing has revolutionized large-scale computation, storage, application development, and web services. Enterprises and service providers no longer need to spend directly on hardware, software, maintenance, and manpower. They can outsource them to cloud vendors and be functional in a matter of minutes with fine-grained control over scalability. The key advantages include on-demand scaling,

[1] Short for CLoud hOME.

pay-as-you-go price model, and high-speed network access [28, 36].

In view of the discussion above, it seems natural that a number researchers have advocated the use of the cloud to implement smart homes [41, 44, 45, 10]. But not surprisingly, none of these proposals have been able to garner much traction. We argue that while the advantages of migration to a cloud have been highlighted *ad infinitum*, the challenges have largely been ignored. We believe that these challenges and practical considerations need to be addressed before a cloud-driven smart home can become a reality. These challenges range from an increase in network traffic and opaque security to weak fault-tolerance and non-trivial cost model. Therefore, the major contribution of this paper is a discussion of these problems and where possible, a proposition of potential solutions. To drive the discussion, we also present the design and architecture of a cloud-aware smart home, dubbed *Clome*. The design of Clome is general enough to be applicable to any smart-home setup and realistic enough to be readily implemented and deployed.

Structure of the Paper: We present the motivation behind cloud-based smart homes in §2.§3 presents the architecture of Clome and proposed implementation details. Applications that can benefit from a cloud based implementation are enumerated in §4. §5 highlights the challenges that arise due to a cloud environment. §6 gives an outline of related work. We conclude in §7 and also discuss future directions.

2 Motivation

To motivate the discussion, this section lists a number of advantages of moving home applications to the cloud.

Application Development: Development on the cloud will shorten development and prototyping time [10]. Also, the use of a common execution platform will lead to common programming practices and design standardization.

Device Heterogeneity: At present, homes contain a large number of diverse devices which complicates application development and interoperability [21, 18]. The use of thin-client computing where all the computation is performed on the cloud and only the output is displayed in the home, will lead to device homogeneity.

Flexibility and Scalability: Integrated smart home systems are inflexible, while on the other hand using multiple vendors can lead to interoperability issues [18]. Thus, changing or adding a new device requires a number of strict design choices. Using *dumb* client-side devices in the home makes the entire system flexible. Additionally, software upgrades can be applied to applications without a need to update the firmware in the thin-client. Further, the requirements of new applications can be fulfilled by scaling up the cloud resources.

Energy Efficiency: In 2005, households accounted for 42% of energy consumption in the European Union [40]. Electricity consumption can be reduced as data centers effectively use smart load management [34], efficient architecture [26], and energy-aware routing [39]. Moving computation to the cloud also moves heat dissipation to the cloud where data centers make use of techniques such as high-efficiency water-based cooling systems to act as heat sinks [27].

Pricing: Users can also take advantage of both temporal and geographic fluctuation in electricity prices to reduce their bill [38]. Studies have shown that government agencies that have moved to the cloud have been able to make savings between 25 to 50 percent [42]. Similarly, moving compute intensive applications from the home to the cloud will also lead to comparable gains.

Mobility: As more users adopt this model, standardized devices will become a part of every home. Users can move residence and start using their existing applications as their account/profile would be stored on the cloud.

Common User Interface: Current user interfaces in the home are too complicated for the average user. The use of a common user interface through which users can interact with the system and give their input to devices makes the environment accessible to everyone [19]. In addition, the current desktop-centric interfaces are incompatible with cloud-backed applications where people take center stage [35]. Decoupling the computation infrastructure (by moving it to the cloud) and the input system, enables multiple user interfaces to exist side by side allowing user-centric customization.

Network Management and Security: The use of a common processing platform allows users to take advantage of applications that can ensure proper network configuration [11], reduce network complexity [19], and provide security against attacks and spam [23]. The synergy of these applications will allow users to get the most out of their networks.

3 Clome

This section presents the architecture of Clome followed by a proposed implementation.

3.1 Architecture

We present one possible[2] architecture in Figure 1. All entities inside the house are connected to the outside world

[2]We do not claim that this is the best architecture for a cloud-enabled smart house but just a good starting point to elicit discussion.

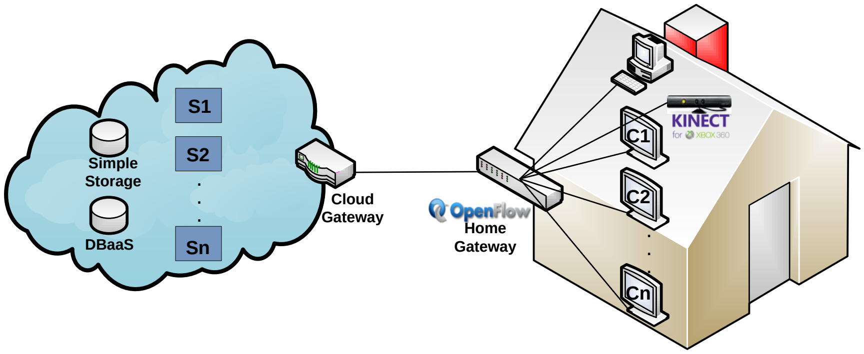

Figure 1: Architectural overview of Clome. S1 to Sn and C1 to Cn represent the server and client ends of thin-client applications, respectively.

through a programmable network switch. All applications and smart appliances have a CPU-heavy end staged in the cloud represented by *S* and a thin-client end represented by *C*. Applications in the cloud can make use of both simple storage and a transactional database. Users interact with and control the system and all applications through a natural user interface. Traditional devices such as PCs, smart phones, and tablets are also connected to the same network and can also be used to access applications staged on the cloud.

3.2 Proposed Implementation

The gateway is enabled by an OpenFlow compliant switch [31] connected to a high-speed Internet backbone. For thin-client computing, we advocate the use of THINC [16], a high-performance system that virtualizes the display at the device driver level. It uses a small set of low-level display commands that reduce both computation and bandwidth overheads. In addition, it implements a number of optimizations to ensure high interactivity even in low-latency networks. The natural user interface in this architecture is enabled by Microsoft's Xbox Kinect [5]. It has a built-in RGB camera, depth sensor and multi-array microphone. These enable Kinect to get user input through visual gestures and voice commands.

4 Applications

In this section we describe both new applications and existing ones from the literature which can benefit from a cloud based implementation. The applications can be made available to users through a *Home Store* [21]. These applications can be categorized into 2 broad categories: 1) User-centric, and 2) System-level. We discuss the former first followed by the latter.

4.1 User-centric Applications

4.1.1 Storage

Home users need multiple media such as compact discs and specialized storage drives to store their music collections. They also buy sophisticated set-top boxes and media centers to store their favourite movies and TV shows. This is both expensive in terms of hardware and extremely inflexible, i.e. there's an upper bound on the storage capability of a device. On the other hand, storing all this media in the cloud is both cost effective and scalable. Applications can directly stream favourite media from the cloud. Transactional databases can be used to implement applications that require structured data such as phonebooks and large music catalogues.

4.1.2 Gaming

According to a study [7], two-thirds of US households own a gaming console. Out of these, 65% of houses also have broadband connectivity. Additionally, it is expected that 260 million consoles will be connected to the Internet by 2015 [3]. These consoles have high processing speed, memory and local storage. For example, the Xbox 360 has 3 CPU cores which run at 3.2 Ghz, 512 MB DRAM, a 500 Mhz GPU and optional storage of up to 500 GB [12]. A cloud version of these consoles would perform its processing on the cloud and its output would be streamed directly to television sets. This *cloud console* would receive its input over the network from the Kinect controller. Multi-player capability is enabled through the use of Virtual LANs in the cloud. Such an approach would make the purchasing of new upgraded consoles redundant as the cloud resources can easily be scaled up with the emergence of newer games.

It is also important to highlight that a cloud-based gaming model greatly simplifies the architecture of Massively Multiplayer Online Games (MMOGs). Such cloud-based systems can gracefully service millions of concurrent users [30].

4.1.3 Automated Surveillance and Voice Recognition

Surveillance systems are one of the most popular components of a smart home. They can be integrated with the security system to provide new functions such as automatically opening the door when a home owner is at the door using face recognition. Additionally, these systems can also be voice activated and can discern the subtleties of natural human language to perform different tasks. Unfortunately, computer vision and natural language processing algorithms are compute intensive. The solution lies in the model of cloud enabled robotics which offload CPU-intensive tasks including speech recognition and 3D mapping to the cloud [14]. Such a system can also make use of distributed data intensive computing systems on the cloud for face recognition and other computer vision tasks [43].

4.1.4 Health Monitoring

Services such as Vignet [9] provide personalized health care solutions. Such services can be made a part of every home and their functionality can be enhanced by using the cloud. New health applications can use data from multiple sensors located in the home. Applications in the cloud can then mine the data to look for unusual patterns. These applications raise a flag when something abnormal happens to alert paramedic services. Privacy is ensured as the data is in the cloud and third-parties only get alerts.

4.2 System-level Applications

4.2.1 Security

Poor management and lack of technical know-how make home networks easy targets for a wide range of attacks. Also, systems in these networks can become compromised and become a source of spam and malicious traffic [23]. One suggested approach [23] to tackle this problem involves the use of OpenFlow programmable networking switches [31] and distributed inference algorithms in a closed feedback loop. The OpenFlow switch is used to both forward traffic to the inference engine and to implement new traffic filtering rules. These rules are computed by the distributed inference engine that uses various spam filtering and botnet/malware detection algorithms. In case of Clome, this third party inference engine – because of its compute-intensive nature – is deployed in the cloud from where it can control the OpenFlow gateway switch. Users of this application form region-wise hierarchical Virtual LANs to be a part of the distributed inference.

4.2.2 Misconfiguration

As mentioned earlier, the home network lacks a local administrator who can properly configure the large number of diverse applications and devices within it. *Misconfiguration* is a problem which arises due to the inexperience of the home user or the interaction of various components with each other [11]. Such misconfiguration can easily bring down the home network. NetPrints [11] is a tool which leverages the fact that there must be at least one user in the network with a correct configuration, and uses that information to properly configure misbehaving elements. It automatically indexes and retrieves this shared knowledge. Users label configurations as either "good" or "bad". NetPrints has a client/server model, in which the client runs on all elements in the network and collects configuration information and a network traffic trace. It uploads this information to a remote server which uses tree based learning and a *mutation algorithm* to construct a tree of both working and faulty configuration settings and gives suggested fixes to the clients. As the number of clients increases, the server can become a performance bottleneck. Therefore, Clome hosts the server on the cloud.

4.2.3 Traffic Monitoring

The popularity of cloud-managed routers from companies such as Meraki [4] has underlined the power of decentralized management and control. These routers enable features such as per-application QoS and traffic analysis. But they are designed for large enterprises with up to 5000 sites which makes them both unfeasible and unaffordable for home users. Interestingly, studies have shown that making per-user usage information available to all households members can help in better management of bandwidth [20]. Additionally, decentralized monitoring can automatically limit the bandwidth of both users and applications. Such a system can also be used to implement parental controls, in which undesirable content can be blocked. Therefore, Clome uses commodity OpenFlow switches [31] and monitoring algorithms running on the cloud to enable traffic monitoring and management.

4.2.4 Smarter Network

Problems that the previous three applications address can also be attributed to the inherent mismatch between the home environment and the architecture of the Internet [19]. The reliance of home networks on the end-to-end principle leads to end-host problems. Calvert et al. [19] propose a "smart middle" approach which turns the home network into an "edge network". This approach simplifies packet forwarding, device monitoring and brokering, and policy management. It makes use of a central *portal* as the control entity which provides connectivity, policy management, and mediation between devices. The portal controls both intra and inter home communication through a managed switch called the "interconnect". The portal also maintains a database of device location and behaviour. In Clome the portal is housed in the cloud and an OpenFlow switch [31] is used as the interconnect.

5 Practical Implications

In this section we highlight some challenges that arise as a result of moving home applications to the cloud. We also suggest possible solutions to these challenges.

5.1 Cost Model

Assuming a *Large Instance*[3] from Amazon EC2, a quick back-of-the-envelope calculation reveals that such an instance would cost approximately $245 per month at the current per hour rate[4]. Moving roughly 1 TB of data in and out of the cloud per month would incur an additional cost of $100. Further, 1 TB of permanent storage in the Elastic Block Store adds another $100. Factoring in I/O request costs and location variation, the total monthly cost comes out to be roughly $500. This is a reasonable price to pay for the large number of benefits highlighted in §2. Additionally, we believe that as cloud usage by home users increases, cloud vendors will introduce monthly package deals, in the same way as Internet charges have evolved from per hour pricing to a monthly flat-rate.

5.2 Choosing a Cloud Vendor

In a home environment, tenants have the luxury of choosing vendors for different services such as cable TV or broadband access. But in contrast, choosing a cloud vendor is a non-trivial task due to: 1) The different charging models of cloud services. For example, some charge by the hour while others by computation cycles, 2) The computation behaviour of different applications. For example, some applications might be compute intensive while others might be I/O bound. Fortunately, CloudCmp [28] is a framework which can assist users in directly choosing a platform. The same framework can also be used by application developers to make recommendations to users.

[3] 7.5 GB memory, 4 EC2 Compute Units, 850 GB instance storage, 64-bit platform.

[4] `http://aws.amazon.com/ec2/pricing/`

5.3 Cloud Vendor API Lock-in

A key selling point of cloud-enabled smart homes is mobility: tenants can change residence and their applications and profiles follow suit. This vision is in tension with cloud vendor API lock-in wherein applications are designed around specific APIs and any change in the API requires a complete application re-design [13]. To abstract away the API and low-level details of each cloud vendor, we propose the use of abstraction libraries such as Apache LibCloud [1], an opensource client library to standardize the interaction of applications with cloud services.

5.4 Copyright

The cloud has been touted as a convenient storage container for personal audio, image, and video collections. At the same time, the intellectual property rights side of the picture has not received the required attention. For instance, most image collections consist of pictures taken by the users themselves. In contrast, audio and video collections largely contain music, movies, and TV shows either purchased online or ripped from CDs, DVDs, etc. In some cases, some of this content is illegally downloaded from P2P file sharing networks. Services such as Apple's iCloud deal with illegal music by virtually legalising it for a small yearly premium [15]. We believe that the situation will be complicated by the addition of a diverse of content. Legal experts will need to be brought on-board before such storage containers see the light of day.

5.5 Inter-Application Communication

In some cases it might be necessary for two applications to communicate with each other. For example, a network management system might need regular traffic information from the security application. But the situation is complicated if these applications reside inside different Virtual Machines. CloudPolice [36], an end-host hypervisor access control system can be used to implement privacy and security. In addition, *feature interaction* between different applications might result due to inter-application communication. Feature interaction is beyond the scope of this paper but the approach described in [25] using bytecode profiles can easily be ported to a cloud environment.

5.6 Increase in Home Network Traffic

Communication between the client and server ends of applications in Clome imposes an additional network traffic overhead. But with reported broadband speeds of up to 17/1.2 Mbps (downlink/uplink) in some residences [29], we believe that current and obviously future networks will easily be able to support such traffic. In addition, different compression and encoding techniques can be used in tandem as well. Furthermore, a local caching server can also be deployed inside the home which can cache content which is popular across devices. Another challenge is network allocation and QoS for different applications. Standard network traffic is predominantly TCP and UDP based but streaming applications implement their own custom protocols such as RTP. Realistically, different applications will use different transport and application level protocols to best suit their requirements and the interaction of all them on the same network link is an open problem.

5.7 Security

Providing security is an overarching goal as a security breach in a home control system can be fatal. One can easily imagine an adversary gaining control over applications or accessing private data. This can happen both on the wire and at the cloud end. In case of the former, lightweight encryption at any level in the network stack (IPsec (L3), tcpcrypt (L4), or TLS/SSL (L7)) can be used to protect information. The situation is more complicated at the cloud end as data security in the cloud is a challenging problem but at the same time the cloud is uniquely positioned to enable confidentiality and auditability [13]. So much so that even healthcare companies with sensitive patient records have already moved to the cloud. In addition, CryptDB-like [37] systems enable queries to be run over encrypted data so that even cloud providers and system administrators cannot access it to ensure privacy. Finally, virtual machines can sandbox the user's data and computation within a single domain and make them inaccessible to malicious users.

5.8 Fault Tolerance

Datacenters which house the cloud are failure-prone due to a number of reasons, including component and power failure. Very recently Amazon Web Services (AWS) experienced problems leading to a blackout of several major websites and services [8]. This can be disastrous for a smart home. A number of solutions to this problem can be employed. First and foremost, cloud services are geo-replicated so that if one site fails, computation and storage can continue from another. Secondly, users have the option of subscribing to a secondary (backup) cloud vendor, which adds another line of defense against failure. Finally, there can be a dichotomy between critical and non-critical applications, with the former provided failure redundancy within the home environment itself (more details later).

Another problem that can plague a cloud-enabled smart home is Internet failure. In such cases, a backup network subscription can also be maintained. Moreover, in the common case, each home would be represented by a single VM staged in the cloud. As most applications are susceptible to software bugs, the entire VM can crash due to a single application. To avoid such a situation, applications can be housed within lightweight OS-level containers, to ensure failure and performance isolation.

5.9 Opaque Cloud Setup

While subscribing to a secondary cloud vendor is one way to ensure fault-tolerance, this too is not foolproof as cloud vendors at times depend on services provided by the same third party [24]. For instance, two cloud vendors might be dependent on storage provided by the same storage provider. Therefore, their failure will be correlated. Unfortunately, there is no mechanism for the application to have such insight as these details are abstracted away by the cloud stack. One possible solution for this problem is for each vendor to expose an explicit dependency graph which can be used by the rich set of smart-home applications to reason about the underlying structure [24], thus negating this failure correlation.

5.10 Everything in the Cloud?

So far we have assumed that all applications can be moved to the cloud with potential benefits. At the same time, it is also important to acknowledge that moving some trivial applications to the cloud might be overkill. For instance, moving the logic of a light dimmer switch seems highly unnecessary. Therefore, we are working on defining a generic metric which users and developers can employ to find out if an application can benefit from a cloud-based implementation. For critical applications and those with no visible benefits from a migration to the cloud, we envision a centralized HomeOS [21] server[5] running within the house. Additionally, this server can also be used to access and configure the cloud instance. Furthermore, to ensure complete privacy, users can store their confidential data locally and grant access to remote applications on a need to know basis. Finally, the same local server can receive input from data-intensive sensors and only forward a periodic snapshot to the cloud for any non-trivial processing.

6 Related Work

In this section we present relevant related work which inspired us to write this paper.

[5]This server need not be a full-fledged machine. For most simplistic applications, a Raspberry Pi [6] will suffice.

HomeOS [21] is a home-centric operating system designed to negate device heterogeneity and to provide a common application development platform. Further, these applications can be downloaded from a *HomeStore* on the same lines as the Apple *App Store*. In the same vein, Clome applications can easily be downloaded from a store to the user's cloud resources. In addition, as mentioned in §5.10, a local HomeOS can be used to run applications for which a cloud migration would be overkill. Niedermayer et al. [33] have tried to devise strategies for the *Future Internet*. Out of these, the strategies most relevant to our work are, a) defining a well known boundary for a home network and all the devices and users within it and, b) providing centralized network resources that make use of cloud computing and P2P networks for data storage and context-dependent access. Clome is complementary to their work as it deals with the *Future Home* by making use of the same design principles.

A cloud based smart home is also described by [41, 44, 45]. Unfortunately none of them discuss the benefits of such a set for a home environment or any security implications. Similarly, Reference [10] envisions a *smarter* home that makes use of the cloud for faster service development and deployment cycle but does not address any practical deployment issues.

7 Conclusion and Future Work

Using a proof-of-concept cloud-enabled smart home, dubbed Clome, we argued that a number of practical issues need to be addressed before such architectures can find real-world deployment. These issues include cloud vendor API lock-in, security, fault-tolerance, and network traffic. We took the position that some of these issues can be easily tackled using existing solutions but others require attention from the research community. We also highlighted some of the advantages of moving applications to the cloud, such as minimizing the computation and energy footprint. Furthermore, our design of Clome builds upon existing technologies, thus enabling it to be readily implemented and deployed.

As Clome is a work-in-progress, our future work is extensive. It includes the testing and evaluation of all applications described in this paper. We also intend on deploying these applications in actual homes to study the effect of the bandwidth overhead. Further, we are also interested in designing thin-client components that can seamlessly be embedded in existing appliances.

References

[1] Apache LibCloud. `http://incubator.apache.org/libcloud/`.

[2] Apple TV. `http://www.apple.com/appletv/`.

[3] Examiner- 65% of broadband households have a game console. `http://exm.nr/gzkoLj`.

[4] Meraki MX Cloud Managed Routers. `http://meraki.com/products_services/mx/`.

[5] Microsoft Kinect 3D motion controller. `http://www.xbox.com/en-US/kinect`.

[6] Raspberry Pi – An ARM GNU/Linux box. `http://www.raspberrypi.org/`.

[7] Softpedia - Two Thirds of United States Households Have a Gaming Console. `http://bit.ly/5LRQT3`.

[8] Summary of the AWS Service Event in the US East Region. `http://aws.amazon.com/message/67457/`.

[9] Vignet. `http://vignetcorp.com/index.php`.

[10] The IBM vision of a smarter home enabled by cloud technology. White paper, IBM Sales & Distribution, January 2010.

[11] AGGARWAL, B., BHAGWAN, R., DAS, T., ESWARAN, S., PADMANABHAN, V. N., AND VOELKER, G. M. NetPrints: diagnosing home network misconfigurations using shared knowledge. In *Proceedings of the 6th USENIX symposium on Networked systems design and implementation* (2009).

[12] ANDREWS, J., AND BAKER, N. Xbox 360 system architecture. *IEEE Micro 26* (March 2006).

[13] ARMBRUST, M., FOX, A., GRIFFITH, R., JOSEPH, A. D., KATZ, R., KONWINSKI, A., LEE, G., PATTERSON, D., RABKIN, A., STOICA, I., AND ZAHARIA, M. A view of cloud computing. *Commun. ACM 53*, 4 (Apr. 2010).

[14] ARUMUGAM, R., ENTI, V. R., LIU, B., WU, X., BASKARAN, K., FOONG, F. K., KUMAR, A. S., KANG, D. M., AND GOH, W. K. DAvinCi: A cloud computing framework for service robots. In *Robotics and Automation (ICRA), IEEE International Conference on* (2010).

[15] BAILEY, J. What iCloud and Music Match Mean for the Piracy Fight, June 2011. `http://www.plagiarismtoday.com/2011/06/09/what-icloud-and-music-match-mean-for-the-piracy-fight/`.

[16] BARATTO, R. A., KIM, L. N., AND NIEH, J. THINC: a virtual display architecture for thin-client computing. In *Proceedings of the twentieth ACM symposium on Operating systems principles* (2005), SOSP '05.

[17] BOHLI, J.-M., SORGE, C., AND WESTHOFF, D. Initial observations on economics, pricing, and penetration of the internet of things market. *SIGCOMM Comput. Commun. Rev. 39* (March 2009).

[18] BRUSH, A., LEE, B., MAHAJAN, R., AGARWAL, S., SAROIU, S., AND DIXON, C. Home Automation in the Wild: Challenges and Opportunities. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)* (2011).

[19] CALVERT, K. L., KEITH, W., REBECCA, E., AND GRINTER, E. Moving Toward the Middle: The Case Against the End-to-End Argument. In *Sixth Workshop on Hot Topics in Networks* (2007).

[20] CHETTY, M., BANKS, R., HARPER, R., REGAN, T., SELLEN, A., GKANTSIDIS, C., KARAGIANNIS, T., AND KEY, P. Who's hogging the bandwidth: the consequences of revealing the invisible in the home. In *Proceedings of the 28th international conference on Human factors in computing systems* (2010), CHI '10.

[21] DIXON, C., MAHAJAN, R., AGARWAL, S., BRUSH, A. J., LEE, B., SAROIU, S., AND BAHL, V. The home needs an operating system (and an app store). In *Proceedings of the Ninth ACM SIGCOMM Workshop on Hot Topics in Networks* (2010), Hotnets '10.

[22] ERMAN, J., GERBER, A., AND SEN, S. HTTP in the home: it is not just about PCs. *SIGCOMM Comput. Commun. Rev. 41*.

[23] FEAMSTER, N. Outsourcing home network security. In *Proceedings of the 2010 ACM SIGCOMM workshop on Home networks* (2010), HomeNets '10.

[24] FORD, B. Icebergs in the clouds: the other risks of cloud computing. In *Proceedings of the 4th USENIX conference on Hot Topics in Cloud Computing* (Berkeley, CA, USA, 2012), HotCloud'12, USENIX Association.

[25] GREAVES, D. J., GORDON, D., ALVI, A., AND OMITOLA, T. Using a .NET Checkability Profile to Limit Interactions between Embedded Controllers. In *Proceedings of the 2008 Second International Conference on Sensor Technologies and Applications* (2008).

[26] GYARMATI, L., AND TRINH, T. A. How can architecture help to reduce energy consumption in data center networking? In *Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking* (2010), e-Energy '10.

[27] KATZ, R. Tech Titans Building Boom. *IEEE Spectrum 46* (February 2009).

[28] LI, A., YANG, X., KANDULA, S., AND ZHANG, M. CloudCmp: shopping for a cloud made easy. In *Proceedings of the 2nd USENIX conference on Hot topics in cloud computing* (2010), HotCloud '10.

[29] MAIER, G., FELDMANN, A., PAXSON, V., AND ALLMAN, M. On dominant characteristics of residential broadband internet traffic. In *Proceedings of the 9th ACM SIGCOMM Internet measurement conference* (2009), IMC '09.

[30] MARZOLLA, M., FERRETTI, S., AND D'ANGELO, G. Dynamic Scalability for Next Generation Gaming Infrastructures. In *DIstributed SImulation & Online gaming workshop (DISIO 2011) - ICST Conference on Simulation Tools and Techniques (SIMUTools 2011)* (2011).

[31] MCKEOWN, N., ANDERSON, T., BALAKRISHNAN, H., PARULKAR, G., PETERSON, L., REXFORD, J., SHENKER, S., AND TURNER, J. OpenFlow: enabling innovation in campus networks. *SIGCOMM Comput. Commun. Rev. 38* (March 2008).

[32] MERCHANT, S. Looking at SDN Beyond the Data Center. Open Networking Summit, 2012.

[33] NIEDERMAYER, H., HOLZ, R., PAHL, M.-O., AND CARLE, G. On using home networks and cloud computing for a future internet of things. In *Proceedings of the Second Future internet conference on Future internet* (2010), FIS '2009.

[34] PAROLINI, L., SINOPOLI, B., AND KROGH, B. H. Reducing data center energy consumption via coordinated cooling and load management. In *Proceedings of the 2008 conference on Power aware computing and systems* (2008), HotPower '08.

[35] PHAM, H., PALUSKA, J. M., AND WARD, S. User interface models for the cloud. In *ACM CHI Workshop on Designing Interaction for the Cloud*, CHI '11.

[36] POPA, L., YU, M., KO, S. Y., RATNASAMY, S., AND STOICA, I. CloudPolice: taking access control out of the network. In *Proceedings of the Ninth ACM SIGCOMM Workshop on Hot Topics in Networks* (2010), Hotnets '10.

[37] POPA, R. A., REDFIELD, C. M. S., ZELDOVICH, N., AND BALAKRISHNAN, H. CryptDB: protecting confidentiality with encrypted query processing. In *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles* (2011), SOSP '11.

[38] QURESHI, A., WEBER, R., BALAKRISHNAN, H., GUTTAG, J., AND MAGGS, B. Cutting the electric bill for internet-scale systems. In *Proceedings of the ACM SIGCOMM 2009 conference on Data communication* (2009), SIGCOMM '09.

[39] SHANG, Y., LI, D., AND XU, M. Energy-aware routing in data center network. In *Proceedings of the first ACM SIGCOMM workshop on Green networking* (2010), Green Networking '10.

[40] TOMPROS, S., MOURATIDIS, N., CARAGIOZIDIS, M., HRASNICA, H., AND GAVRAS, A. A pervasive network architecture featuring intelligent energy management of households. In *Proceedings of the 1st international conference on PErvasive Technologies Related to Assistive Environments* (2008), PETRA '08.

[41] WEI, Z., QIN, S., JIA, D., AND YANG, Y. Research and Design of Cloud Architecture for Smart Home. *IEEE International Conference on Software Engineering and Service Sciences (ICSESS)* (2010).

[42] WEST, D. Saving Money Through Cloud Computing. Brookings Institution, 2010.

[43] WHITE, B., YEH, T., LIN, J., AND DAVIS, L. Web-scale computer vision using MapReduce for multimedia data mining. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining* (2010), MDMKDD '10.

[44] YANG, Y., WEI, Z., JIA, D., CONG, Y., AND SHAN, R. A cloud architecture based on smart home. In *Second International Workshop on Education Technology and Computer Science (ETCS)* (2010).

[45] YE, X., AND HUANG, J. A framework for cloud-based smart home. In *Computer Science and Network Technology (ICCSNT), 2011 International Conference on* (2011).