Enabling Location Based Services in Data Centers

K. Kant, N. Udar, R. Viswanathan
krishna.kant@intel.com, neha.udar@intel.com, viswa@engr.siu.edu

Abstract—**In this paper, we explore services and capabilities that can be enabled by the localization of various "assets" in a data center or IT environment. We also describe the underlying location estimation method and the protocol to enable localization. Finally, we present a management framework for these services and present a few case studies to assess benefits of location based services in data centers.**

Key words: Asset Localization, Wireless USB (WUSB), Ultra Wideband (UWB), RFID, Management services.

I. INTRODUCTION

Major data centers routinely sport several tens of thousands of "assets" (servers, switches, storage bricks, etc.) that usually go into standard slots in a rack or a chassis that fits the rack. The racks are 78" high, 23-25" wide and 26-30" deep. The rows of racks are arranged in pairs so that the servers in successive odd-even row pairs face one another. Fig. 1 shows a typical row of a data center with the popular "rack mount" assets which come in 1U/2U/4U sizes (1U is about 1.8"). The other, increasingly common configuration involves "blade servers" that go vertically into chassis, and the chassis fits in the rack. A typical rack may take about 6 chassi, each with about 14 blade servers.

The ease with which assets can be inserted into and removed from their slots makes the assets quite "mobile". There are a variety of reasons for moving assets around in a data center, these include replacement of obsolete/faulty asset, OS and application software patching, physical space reorganization, logical group changes to handle evolving applications and services, etc. This makes asset tracking a substantial problem in large data centers and some tracking solutions are beginning to emerge [3].

In our previous work [7], [8], we have explored asset tracking by exploiting wireless USB radios embedded in servers. Wireless USB (WUSB) is an upcoming replacement for the wired USB and is expected to be ultimately ubiquitous. WUSB uses ultra-wide band (UWB) at its physical layer which can provide much better localization than other technologies such as WLAN [5] and much more cheaply than RFID [3]. In [8] we show that a combination of careful power control and exploitation of the geometry can localize individual servers with good accuracy.

In this paper, we exploit this localization capability of UWB to provide a variety of location based services (LBS) in the data centers. Unlike the traditional LBS, our focus here is not on arming humans with useful information, but to allow the middleware to do a better job of resource management. As a simple example, each rack in a data center has certain capacity for power circuits which cannot be exceeded. Therefore, a knowledge of rack membership of servers can allow abiding by this restriction. However, in a large data center, we need more than just locations – we need an efficient mechanism to exchange location and other attributes (e.g., server load), so that it is possible to make good provisioning/migration decisions. This is where LBS services come in. We envision the middleware to still be making the final selection of servers based on the appropriate policies; the function of LBS is merely to identify a "good" set of assets.

The rest of the paper is organized as follows. Section II describes asset localization technologies and discusses WUSB based approach briefly. Section III discusses how LBS fits in the management framework for the servers. Section III-D illustrates how LBS can be exploited for power and thermal balance among servers. Finally, section IV concludes the discussion.

II. LOCALIZATION IN DATA CENTERS

In this section we discuss localization technologies, WUSB localization protocol and some implementation issues.

Fig 1. Snapshot of Row of a Typical Data Center

A. Localization Technologies

In principle, the most straight forward way to track assets is to make the asset enclosures (chassis, racks, etc.) intelligent so that they can detect and identify the asset being inserted or removed from a slot. Unfortunately, most racks do not have this intelligence (chassis often do). Even so, the enclosures themselves would still need to be localized. Hence we need to look for other (perhaps wireless) solutions to the problem. Furthermore, any changes to existing infrastructure or significant external infrastructure for asset management is expensive and may itself require management. Therefore, low cost and low impact solutions are a must.

RFID based localization appears to be a natural solution for data centers but unfortunately it requires substantial infrastructure for acceptable accuracy. In particular, reference [3] describes such a system where each server has a RFID tag and a RFID reader per rack. The reader has a directional antenna mounted on a motorized track and each rack has a sensor controller aware of its position. The HP solution cannot be implemented due to prohibitive infrastructure cost. The achievable accuracy of RFID system implemented by LANDMARC is less than 2m [5]. Thus, RFID solution is neither cost effective nor can achieve the desired localization accuracy.

Localization is a very well studied problem in wireless networks; however, our interest is in only those technologies that are accurate enough to locate individual racks/chassis and (preferably) individual servers. Note that the localization of 1U servers requires accuracies of the order of 1 inch. In the following we survey some localization technologies and address their applicability to data centers.

Wireless LAN (WLAN) based localization has been extensively explored in the literature [5] and can be implemented easily in software. Unfortunately, even with specialized techniques such as multipath decomposition method [5], the root mean square error (RSME) in the best line-of-sight (LoS) case is only 1.1 meters.

Ultrasonic or surface acoustic wave (SAW) systems perform localization based on time of flight (TOF) of sound waves. Because of very low speed of sound, SAW systems can measure distance with an accuracy of a few cm. Unfortunately, SAW systems require substantial infrastructure and uninterrupted sound channels between emitter and receivers.

In [7], [8], we have explored a wireless USB (WUSB) based localization solution that assumes that each server comes fitted with a WUSB radio (as a replacement for or in addition to the wired USB interface) that has requisite time of arrival (ToA) based measurement capabilities. This can provide an effective and inexpensive localization solution.

B. WUSB Standardization and Platform Issues

The IEEE standards group on personal area networks (PANs) is actively working on UWB based communications under Wi-Media alliance and 802.15.4 task group. WUSB is a middleware layer that runs atop Wimedia MAC. 802.15.4a focuses on low data rate (LDR) applications ($\leq$ 0.25 Mbps) which is set to serve the specific needs of industrial, residential and medical applications. The design of 802.15.4a specifically addresses localization capability and is ideally suited for LBS applications. Our suboptimal choice of WUSB/Wimedia is motivated by practical considerations: as stated above, we expect WUSB to be ubiquitous; therefore, using Wimedia does not require any additional expense or complexity for data center owners. Of course, everything about the proposed techniques (with the exception of timing) applies to 802.15.4a as well.

WUSB uses the MAC protocol based on Wimedia standard citedmac. It is a domain dependent MAC with a master-slave architecture involving a Piconet controller (PNC) and up to 255 terminals (slaves). The PNC maintains global timing using a super frame (SF) structure. The SF consists of 256 slots and each slot has duration of 256 microseconds. A SF consists of a beacon period, contention access period, and contention free period. The beacon period is used for PNC to terminal broadcasts, contention access period is used by the terminals to communicate with others or to ask PNC for reserved channel time, and contention free period is dedicated for individual transmissions over agreed upon time slots.

Server localization is often a crucial functionality when the server is inoperational (e.g., replacement, repair or bypass). Consequently, the localization driver is best implemented in the baseboard management controller (BMC) of the server rather than the OS of the main processor. BMC is the main controller that will stay operational so long as the server is plugged in and provides for intelligent platform management [11]. However, providing BMC control over WUSB in post-boot environment is a challenge that is not addressed here.

C. Location Estimation Methods

Localization involves determining the position of an unknown node in a 2 or 3 dimensional space using range estimates from few “reference” nodes, (i.e., nodes with known locations) to an unknown node. The range estimate can be obtained using received signal strength (RSSI), time of arrival (ToA), angle of arrival (AoA) technique or a hybrid method which is a combination of any of these methods. Here, we focus on the most widely used ToA method for UWB ranging. The ToA technique determines the distance by estimating the propagation delay between the transmitter and receiver. The position of an unknown node is then identified using the traditional methods such as the intersection of circles using TOA or intersection of hyperbolas using time difference of arrival between the two ToA’s [10]. However, due to errors in range measurements a statistical estimation technique such as Maximum Likelihood Estimation (MLE) is required. MLE estimates distributional parameters by maximizing the probability that the measurements came from the assumed distribution.

Since the server positions can only take a small number of discrete positions in a rack, the MLE problem can be transformed into a simpler maximum likelihood identification (MLI) problem [8]. MLI exploits the geometry of racks to accurately identify the position of the unknown server.

Fig. 2 shows the rack configuration and an associated coordinate system (x, y, z) where x is the row offset, y is the rack offset within a row, and z is the server height in a rack. Consider rack(0,0) with N plugged in servers. For determining the location of unknown server u MLI uses three reference nodes, of which first two are in rack (0,0) and third one in rack (0,1). Each reference node i (where $i \in 1, 2, 3$) measures the distance to an unknown node u as r_{iu} using ToA. We assume that a range estimate r_{iu} is distributed as Gaussian with zero bias (that is, expected value of the estimate equals true distance) and variance of $\sigma^2 = N_0/2$. The distance between each reference node and N-2 possible positions in the rack is known. Given the 3 range estimates and N-2 possible distances from each of the reference node, N-2 likelihood functions (LFs) are formed. Out of N-2 LF’s, the minimum valued LF identifies the position of an unknown server. In [8] it is shown that the performance of MLI method far exceeds the performance of the traditional methods.

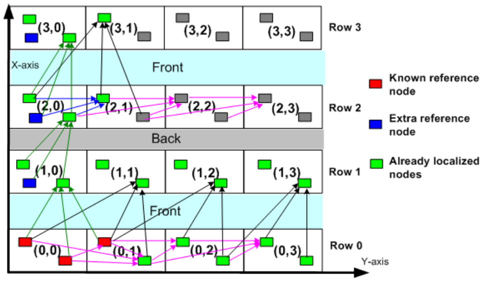

Fig 2. Localization in a Data Center During the Cold Start Phase

D. *Localization Protocol*

Asset localization in data centers involves two distinct phases: (a) *cold start* phase that localizes *all* servers starting with a few reference servers with known locations, and (b) *steady state* phase that tracks individual asset movements subsequently. The *steady state* phase is relatively easy to handle and is not descried here due to space constraints.

The *cold start* phase starts with one of the known server in servers hard coded as PNC and all others in the listening mode. The role of PNC is to form the Piconet with the servers from the current rack and few servers from adjacent and the opposite rack to enable rack to rack localization. One complication in *cold start* localization is the avoidance of servers in racks that we are currently not interested in localizing. This, in turn, requires "macro-localization", i.e., the determination of which rack the responding servers belong to, so that we can suppress the undesirable ones. This is handled by a combination of careful power control and by exploiting the geometry of the racks. Generally the localization proceeds row by row as explained below.

Row 0 Localization: We start with 3 known servers as shown in Fig. 2. During rack(0,0) localization all the unknown servers in rack(0,0) and at least one server in the adjacent rack(0,1) and *two* servers in the opposite rack(1,0) are localized to enable localization in the subsequent racks as shown by red and green/black arrows in Fig. 2. (To avoid clutter, not all arrows are shown.) Once the current rack localization is complete, the PNC in the current rack performs hand off to one of the localized servers(new PNC) in the rack(0,1). Thus, localization continues one rack at a time along with a few localizations in the adjacent and opposite rack until all servers in the last rack of row 0 are localized.

After the last rack localization, PNC in the last rack updates all the servers with the position of their neighbors and hands off to the selected PNC in the last but one rack in row 0. This hand off in the reverse direction continues until the rack(0,0) is reached. Now PNC in rack(0,0) is ready to hand off to the suitable known server in the rack(1,0) (odd numbered row).

Row 1 Localization: At the beginning of the Row 1 localization all the servers in row 0 are localized and the PNC in rack(0,0) selects a known server as a new PNC in rack(1,0). In the beginning of Row 1 localization, each rack in row 1 has at least 2 known servers. But, there are no known servers in row 2. Also, given the alternating rows of front and back facing servers, communication across the "back" aisles is very challenging due to heavily metallic nature of racks as shown in Fig. 2. Therefore, only the racks located at the edge of the one row can communicate with the racks located at the edges of next rows. During rack(1,0) localization all the servers in rack(1,0) and 3 servers in rack(2,0)(next even row) are localized. From rack(1,1) onwards only the servers in the current rack are localized until the last rack is row 1 is localized. The localization in reverse direction continues as in row 1 until the rack(1,0) is reached. The PNC in rack(1,0) hands off to the new PNC in rack(2,0). Location of unknown nodes in successive odd-even row pairs continues similarly and is not discussed here.

E. *Accuracy of Localization Protocol*

The accuracy of localization protocol depends on the variance and bias in range estimates. The variance comes from variations in channel parameters and the bias is generally a result of partial or full occlusion of the receiver relative to the transmitter. Our previous work [7] measured variance and bias in the range estimates by direct measurements in a commercial data center. In our localization protocol, lack of line of sight and hence substantial bias is expected only when we hop across the back aisle. The normal technique for handling bias is to simply estimate it and remove it [1]. Thus, the assumption of no bias is still reasonable. We expect to address the question of bias estimation in future works as it requires much more intensive measurements than what we have currently.

In [8] a maximum likelihood identification (MLI) method was proposed for localization and compared with the traditional method of hyperbolic positioning using Matlab simulation. It was shown that the performance of MLI method far exceeds the traditional method. The probability of error in identifying a location of a node increases with the increase in variance as expected and was found to be the order 10E-5 to 10E-2 for the variances between 0.15 to 0.5. It was further shown in [8] that by controlling the variance via multiple measurements, the rack to rack error propagation can be kept sufficiently small so that the protocol can handle large data centers.

III. LOCATION BASED SERVICES

Once the servers in a data center are localized, interesting LBS can be enabled in a data center. In subsection III-A the need or enabling location based services(LBS) is discussed. Next subsection lists variety of services that can exploit LBS. Subsection III-C explains the management framework for enabling LBS. The last subsection III-D illustrates the role of LBS in power and thermal balance in data centers.

A. *Need for LBS*

Data centers show perennially low average server utilization (5-10% range) and yet ever increasing server count, power consumption, and associated infrastructure and management costs. The low utilization is attributable not only to unpredictable demands but more importantly to the need for isolation among various applications and activities. Virtualization has recently gained acceptance as a way to increase resource utilization in data centers while still maintaining a level of isolation between various applications and activities. Aggressive virtualization leads to the notion of "utility computing" whereby the entire data center can be viewed simply as a pool of resources (computes, storage, special functions, etc.) which can be allocated dynamically to applications based on the current needs.

Virtualization can be viewed as a mechanism to make the physical location of resources irrelevant since any resource can be assigned to any application in this model. While this flexibility brings in several advantages, a location blind resource allocation can lead to anomalies, poor performance and ultimately suboptimal resource usage. In other words, a location aware resource management can retain all the advantages of virtualized data center while avoiding its pitfalls. We discuss these in the next few paragraphs.

The isolation requirement addressed above implies that each application executes on its own "virtual cluster", defined as a set of virtual machines (or virtual nodes) connected via QoS controlled virtual links[4]. However, the performance isolation between applications becomes increasingly difficult as more applications are mapped to common physical resources. Location awareness can be helpful in this regard. The increasing data center size and the utility computing approach make it an increasingly attractive targets of attacks via viruses, worms, focused traffic (distributed denial of service attacks), etc. Confining a virtual cluster to a physical region offers advantages in terms of easier containment of attacks. In this context, the relevant "physical region" is really "network region", e.g., set of servers served by one or more switches or routers; however, the two are strongly related. For example, all blade servers in a chassis share a switch, and all chassis switches in a rack connect to the rack level switch. Thus the location based provisioning and migration can be beneficial from security/isolation perspective. For essentially the same reasons, a location aware allocation can yield better performance for latency sensitive applications since the reduction in number of switches on the communication paths also reduces the communication latency.

The continued increase in processing power and reduction in physical size has increased power densities in data centers to such an extent that both the power-in (i.e., power drawn) and power-out (i.e., power dissipated as heat) have become serious problems. For example, most racks in today's data centers were designed for a maximum of 7 KWHr consumption, but the actual consumption of a fully loaded rack can easily exceed 21 KWHr. As a result, in older data centers, racks are often sparsely populated lest the power circuit capacity be exceeded resulting in a brownout. In addition, the power and cooling costs are becoming substantial percentage of overall costs. Consequently, an intelligent control over both power consumption and cooling becomes essential. Power/thermal issues are inherently tied to the location of the active assets. For example, cooling can be made more effective and cheaper if the servers with high thermal dissipation are not bunched up [6].

The high velocity fans required for effective cooling in increasingly dense environments is making noise also an important issue in data centers. Besides, fans are usually 3rd or 4th largest consumers of power in a platform and may waste a significant fraction of that power as heat. Therefore, an intelligent control of speed of adjacent fans can not only reduce noise but can also make the cooling more effective.

B. Application of LBS

Since the feasible services depend on achievable localization accuracy, we introduce LBS at two levels of localization granularity:

1) Coarse grain localization (CGL), defined as the ability to identify (with, say, 95% or better accuracy), data center racks, pods or cubicals containing small clumps of IT equipment, storage towers, and mobile devices in the vicinity (e.g., people carrying laptops). The desired precision here is ± 0.5 meters.
2) Medium grain localization (MGL), defined as the ability to identify (again, say, with 95% or better accuracy), individual plugged-in assets within a chassis (and by implication, the chassis itself), and individual mobile devices (e.g., laptops, blackberries). The desired precision here is $\approx \pm 5$ cm.

In the following we list a variety of service that can exploit CGL and MGL. The list is not intended to be exhaustive, but merely attempts to indicate the usefulness of LBS within a data center. Also, a real implementation of such services may include some environment and usage model specific elements.

1) Application allocation to minimize IPC (inter-process communication) or storage access delays among the virtual nodes.
2) Temporary inclusion of a mobile device in a logical group within its physical proximity (it is assumed that the device can communicate over a much wider physical range, so this service may use attributes beyond just the ability to communicate).
3) In an IT environment, direct a print job to the nearest working but idle printer.
4) Dynamic migration of VM's among adjacent servers to balance per-server power-in (and especially the power-out).
5) Roving query distribution to maximize power savings and balance out heat dissipation. This technique is opposite of load balancing in that it allows idle servers to go into deep low power states while keeping the active servers very active.
6) Logical grouping of assets based on their location in order to simplify inventory management, allocation, deallocation, migration, etc.

7) Trouble ticket management, i.e., identify the asset that needs replacement, fixing, SW patching, etc.
8) Physically segregated allocation of applications based on their trustworthiness, reliability, sensitivity, or other attributes.
9) Quick quarantine of all servers belonging to the same enclosure as the server that detects a DoS or virus attack.
10) Automated adjustment of air-flow direction flaps from shared fans in order to maximize cooling of hot spots and perhaps control fan noise. This situation is generally applicable to blade chassis which have shared fans. (Racks usually don't).

C. A Management Framework for LBS

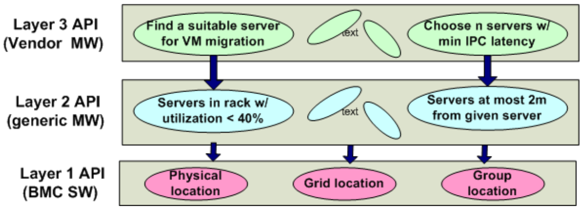

Fig 3. Illustration of LBS Application Layers

Flexible management of virtualized data centers and creation of utility computing environments is currently being driven by initiatives from major SW vendors such Dynamic System Initiative (DSI) from Microsoft, adaptive enterprise from HP, on-demand computing from IBM, and Sun Microsystem's N1. These initiatives are geared towards providing middleware solutions to the dynamic data center management problem based on the information available from the OS and low level management SW running on the BMC [11].

Although the management SW can implement LBS arbitrarily based on the physical locations reported by the localization layer running in the BMC, a more structured approach is highly desirable. We envision the following 3 layers:

1) Layer1: API's to obtain asset location in various formats. At a minimum, three formats seem necessary: (a) Physical 3-D location relative to the chosen origin, (b) Grid based location (rack_row_no, rack_no, asset_no_in_rack), and (c) group level location such as location of the entire rack or chassis.
2) Layer2: API's to identify a group of assets satisfying constraints that relate to their location, static attributes (e.g., installed memory) and perhaps even the current utilization levels. For flexibility, the constraints may be expressed in a declarative fashion (see below).
3) Layer3: LBS themselves, implemented as a part of the middleware. It is envisioned that the LBS will invoke layer2 API's to select promising candidates and then do further selection based on its needs.

The Fig. 3 shows the illustration of these layers and their interactions.

There is a strong trend in management SW to use a standardized representation of the underlying management data and access it using web services. In particular, the distributed management task force (DMTF) has developed a common information model (CIM) for describing computing and business entities that has been adopted widely (www.wbemsolutions.com/tutorials/CIM/cim-specification.html). For example, a CIM model of a NIC will have all relevant attributes of the NIC (e.g. speed, buffer size, TSO and whether it is enabled, etc.). CIM supports hierarchical models (nested classes, instances, inheritance, etc.) for describing complex systems in terms of its components. CIM models can accessed via a web services management (WSMAN) interface for querying and updating the attributes. The location can be part of CIM model and can be accessed via WSMAN services.

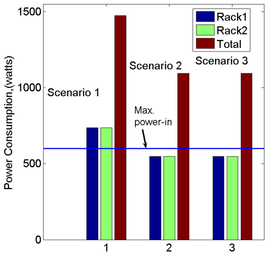

Fig 4. Power consumption for various localization scenarios

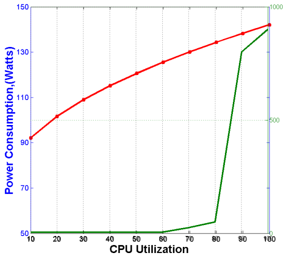

Fig 5. CPU utilization vs power & response time

Although CIM can adequately describe configuration of servers and applications, a more powerful language such as the services modeling language (SML) (www.microsoft.com/business/dsi/serviceml.mspx) is required to specify service related constraints. Being XML based, SML can describe schemas using XML DTD's (document type definition). Furthermore, SML documents can refer to elements in other SML documents and thereby specify complex relationships via schematron (www.schematron.com). For example, it is possible to say something like "allocate the application to a server only if the server utilization is less than 40%". Thus SML can allow for resource management based on declared constraints as opposed to those buried in the middleware code.

D. Exploiting LBS for Power/Thermal Balancing

In this section we show that LBS can be used effectively to handle the issues of power and thermal balance in a data center. Consider a data center having a single row with 2 racks. Each rack has 12 slots and is partially filled with 8 identical servers. Suppose that each rack has maximum power draw capacity of 650 W. Let us consider running an application that demands 320% CPU utilization. In the following subsections, we analyze allocating this application in three different ways:

- Scenario 1: No Localization, the server locations are unknown.
- Scenario 2: CGL, it is known that server belongs to a particular rack but the exact location in the rack is not known.
- Scenario 3: MGL, the exact location of the server in the rack is known.

E. Power-Load Balance

It is well known that the power consumption P relates to the CPU utilization U by a non linear relationship. In [2] the authors performed detailed measurements on streaming media servers with several configurations to study the relation between CPU utilization and the power consumption, and found that the power consumption can be expressed approximately as:

$$P = P_I + (P_F - P_I)U^{0.5} \tag{1}$$

where $P_I =$ is the idle power, P_F is the power when CPU is fully loaded and U is the CPU utilization.

Such a dependence is very much a function of the machine and workload characteristics and there is no suggestion here that this is a general equation. However, it suffices to illustrate a few interesting points about power/thermal balance.

We also make use of the power numbers reported in [2]: an idle power of $P_I = 69$ watts and $P_F = 145$ watts at full load. The authors also specify a low power mode consumption of $P_L = 35$ watts. This mode generally puts the CPU, memory and disk in low-power modes.

Given the power consumption in the idle mode and the low power mode, it is power efficient to distribute higher load on fewer servers and force more servers in the low power mode. The distribution of higher load on fewer servers is limited by response time of the server. As shown in Fig. 5, the response time takes off sharply beyond 60% CPU utilization.

In Scenario1, given that the server locations are unknown, a simple strategy is to distribute the load equally on all the available servers. Each of the 16 servers in this case carries a load of 20% to meet the total load demand of 320%. With equal load sharing, each rack exceeds the maximum power-in for a rack as shown in Fig. 4. In Scenario2 using CGL, it is known which servers belong to either of the 2 racks. Therefore, the total load is divided equally between the two racks. Further, within each rack, 4 out of 8 servers share the 40% load and the remaining servers are put in low power mode. The non-uniform distribution of load among the available servers leads to power saving as shown in Fig. 4 and also meets the maximum power-in requirement of a rack. Scenario 3 is identical to Scenario 2 in terms of power since knowing the precise location of server does not provide any additional advantage. Further power saving can be achieved if 2 servers in each rack carry a load of 60%, one server carries 40% load and the remaining 5 servers in each rack are in the low power mode.

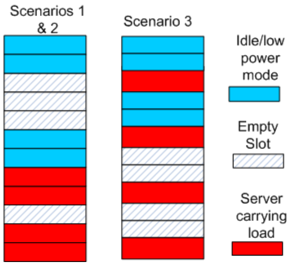

Fig 6. Power/thermal balance scenarios

F. Thermal-Load Balance

Thermal Power dissipated from the CPU is proportional to the power consumed and a non-uniform distribution of thermal power places more demand on cooling the data center [6]. To illustrate the point, let us reconsider the Situation of Scenarios 2 and 3 above, i.e., 8 servers sharing the entire load while the other 8 are put in low power mode. In scenario 2 the lack of precise server location can result in loaded servers being all placed in physical proximities but Scenario 3 can achieve better thermal balance by spreading out the loaded servers as shown in Fig. 6.

IV. CONCLUSIONS

In this paper we introduced an important topic of asset localization in data centers and discussed wireless USB based techniques for the same that does not require any external infrastructure. Further, a localization protocol for systematically localizing assets in a data center was described briefly. We also introduced the notion of location based services and illustrated that localization can be used to obtain power/thermal balance in a data center.

REFERENCES

[1] B. Alavi and K. Pahlavan, "Modeling of ToA Based Distance Measurement Error Using UWB Indoor Radio Measurements", *IEEE Comm. Letters*, vol.10, pp. 275-277, April 2006.

[2] L. Chia-Hung, B. Ying-Wen and L. Ming-Bo, "Estimation by Software for the Power Consumption of Streaming-Media Servers", *IEEE Trans. on Instr. and Meas.*, vol. 56, Issue 5, pp. 1859-1870, Oct. 2007.

[3] HP Press Realse, "HP Creates RFID Technology for Tracking Data Center Assets", Calif., Oct. 17, 2006.

[4] K. Kant, "Virtual Link: An Enabler of Enterprise Utility Computing" , *Proc. of ISPA*, pp. 104-114, Nov. 2006.

[5] H. Liu, H. Darabi and P. Banerjee, "Survey of Wireless Indoor Positioning Techniques and Systems", *IEEE Trans. on Sys., Man and Cybernetics*, Part-C, vol. 37, No.6, pp. 1067-1080, Nov. 2007.

[6] R.K. Sharma, C.E. Bash and et.al, "Balance of Power Dynamic Thermal Management for Internet Data Centers", *IEEE Internet Computing*, pp. 42-29, Feb. 2005.

[7] N. Udar, K. Kant, R. Viswanathan and D. Cheung, "Ultra Wideband Channel Characterization and Ranging in Data Centers", *ICUWB*, pp. 322-327, Sept. 2007.

[8] N. Udar, K. Kant and R. Viswanathan, "Localization of Servers in a Data Center Using Ultra Wideband Radios", *IFIP Networking*, pp. 756-767, May 2008.

[9] WiMedia Draft MAC Standard 0.98, "Distributed Medium Access Control (MAC) for Wireless Networks", *WiMedia*, August 2005.

[10] Y. T. Chan, H. Y. C. Hang, and P. C. Ching, "Exact and approximate Maximum Likelihood Localization Algorithms", *IEEE Trans. on Vehicular Tech.*, vol. 55, Issue 1, pp. 10-16, Jan. 2006.

[11] H. Zhuo, J. Yin and et.al, "Remote Management with the Baseboard Management Controller in Eighth-Generation Dell PowerEdge Servers", *Dell Retrieved*, June 2005.