

Dynamic Power Management using On Demand Paging for Networked Embedded Systems

Yuvraj Agarwal

Dept. of Computer Science & Engg.
University of California at San Diego
yuvraj@cs.ucsd.edu

Curt Schurgers

Dept. of Electrical & Computer Engg.
University of California at San Diego
curts@ece.ucsd.edu

Rajesh Gupta

Dept. of Computer Science & Engg.
University of California at San Diego
gupta@cs.ucsd.edu

Abstract— The power consumption of the network interface plays a major role in determining the total operating lifetime of wireless networked embedded systems. In case of on-demand paging, a low power secondary radio is used to wake up the higher power radio, allowing the latter to sleep for longer periods of time. In this paper we present use of Bluetooth radios to serve as a paging channel for the 802.11b wireless LAN. We have implemented an on-demand paging scheme on an infrastructure based WLAN consisting of iPAQ PDAs equipped with Bluetooth radios and Cisco Aironet wireless networking cards. Our results show power saving ranging from 23% to 48% over the present 802.11b standard operating modes with negligible impact on performance.

I. INTRODUCTION

Networked Embedded Systems (NES) are embedded system with networking interfaces. Our interest is in optimization of power/performance for an NES such as PDAs with wireless interfaces. The communication subsystem accounts for a major portion of total power consumption [1], and can dominate in these battery constrained wireless devices. This is illustrated in Figure 1, which is based on our experiments and iPAQ power figures from [2].

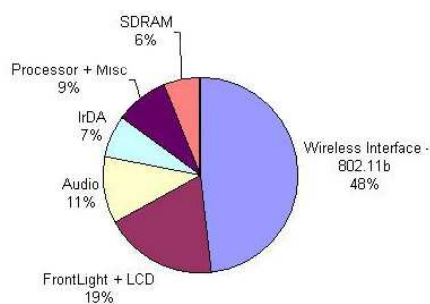


Fig. 1. Power Consumption of various Components of an iPAQ [2]

Since many of these systems come equipped with multiple radio interfaces, such as BT and 802.11, it is possible to use this diversity of radio interfaces to minimize power consumption. In this paper we show how we can leverage commodity radios to reduce total power consumption as well as show techniques to compensate for differences in radio ranges. We have implemented these techniques in a working platform using services in a power aware operating system that shows gains in

system level power management.

When no data transmission or reception is taking place, the radio interface is listening, and the power consumption in this idle state is the dominant component in the average radio power consumption. To remedy this, the IEEE 802.11 MAC protocol contains provisions to reduce the idle power, by transitioning the radio into a sleep state [3, 4]. Periodically, the radio wakes up to check for buffered data. An alternative approach is that the receivers are woken up on demand through a separate *paging* radio. Since the power consumption of BT radios is an order of magnitude less than that of 802.11 radios, as we will show, they are ideally suited to provide this paging functionality. The main 802.11 radio is only woken up on demand by the Access Point (AP) through this BT interface. We explore the issues involved with this scheme such as interference, provide detailed experimental data and investigate practical real-life traffic. We show significant savings above and beyond those of current low-power modes in WLAN, for realistic application scenarios.

II. RELATED WORK

There has been some research suggesting modification and optimization of existing protocols to make them power aware [5, 6, 7]. Regarding power consumption by the wireless interface, Stemm *et al.* [1] have quantified the effect on power consumption by techniques that increase radio sleep time through extensive simulations and application-based power optimizations. The parameters of 802.11b Power Save Mode (PSM) can also be adjusted to reduce energy consumption [8]. An asynchronous variant of PSM was proposed in [9]. In contrast, our approach does not require modifications to the existing standard and can be implemented on existing devices.

In the field of sensor networks, a passive wakeup scheme has been proposed [10]. A very low power receiver structure essentially fulfills the role of a separate wakeup channel. In addition, this approach requires custom built radio hardware, which is a viable option for sensor networks. Our solution targets existing commodity wireless access technologies for wireless data networks. In [11], Chiasserini *et al.* suggest an approach to use a secondary mechanism to wake up a group of nodes by means of small range ID tags. They however look at analytical results only, while we focus on actual implementation, associated practical problems and power measurements.

In their "Wake-on-Wireless" scheme E. Shih *et al.* [12] propose the use of a specially designed custom radio for paging. Their framework has various components including location servers, presence servers. They use proprietary paging radios which are low range and therefore need intermediate proxies. The authors show 17% to 40% increase in lifetimes for their PDA based phone and mention an added latency of 5 to 10 seconds. These energy saving estimates are based on a cell phone based application that has usage patterns with very large idle times. Compared to their work, we use commodity Bluetooth radios and present our solution that works without the need of intermediate proxies. We have also performed extensive power/energy measurements of our framework for various TCP/IP traffic including file transfer and remote sessions. Using this we show that the end to end latency of our scheme is less than 2s in one case to a maximum of 5s depending on the level of energy savings desired.

In STEM [13] the authors propose the use of a second duty-cycled radio to provide paging with their results based on analysis and simulation. Another similar duty-cycling based scheme is presented in [14]. These approaches could be utilized on top of our low power paging channel, as suggested in [13].

III. IEEE 802.11B POWER MODES

The 802.11 standard specifies Medium Access Layer (MAC) power management [3, 4] allowing mobile stations to conserve power by switching to low power modes. In case of an infrastructure network, the power management is centralized in the AP. The radio can be in an *active*, *idle* or *sleep* state. In the *active* state, the radio consumes the maximum power and is used for data transmission and reception. The *idle* state is when the radio is on but is not transmitting or receiving any data. The *sleep* state does not allow data reception or transmission and is a low power state consuming an order of magnitude less power than in the active state. The power figures for typical 802.11b cards are shown in Table I.

Using these states, there are two modes of operation as specified in the 802.11b standard, Awake Mode (AM) or Power Saving Mode (PSM). In case of AM the radio is continuously on and is in the *active* or the *idle* state. This is a high power mode as the radio consumes power equivalent to transmission, reception or idle power. In PSM the radio switches between *active* state and *sleep* state to save power.

TABLE I
TYPICAL POWER CONSUMPTION FOR 802.11B CARDS

| Vendor | Average Power (Watts) | | |
|---------------|-----------------------|------------|-------------|
| | Idle | Trans.(Tx) | Recept.(Rx) |
| Cisco PCM 350 | 1.31 | 1.80 | 1.52 |
| Orinico Gold | 1.02 | 1.50 | 1.22 |
| Linksys | 1.1 | 1.48 | 1.28 |

We use a Cisco Aironet PCM-350 for our experimentation. Power consumption of this card (with no data transmission or

TABLE II
PCM-350 POWER CONSUMPTION FOR VARIOUS POWER MODES (NO TX OR RX)

| Operating Mode | Average Power (Watts) |
|----------------|-----------------------|
| CAM | 1.31 |
| PSP | 0.39 |

reception taking place) for its two modes of operation: Constant Awake Mode (CAM) and Power Saving Polling (PSP), are shown in Table II.

IV. ON DEMAND PAGING USING BLUETOOTH RADIOS

We present an implementation of a low power scheme using Bluetooth (BT) as a paging mechanism to wake up the higher power 802.11b radio as and when required. As described in Section III, the idle power in 802.11b radios is substantial, even in PSM. The idea behind on demand paging is to shut off the 802.11b radio to save idle power whenever no data transmission or reception is taking place. It also reduces the power consumption of switching the interface on and off periodically to check for data buffered at the AP as is done in the PSM of 802.11b.

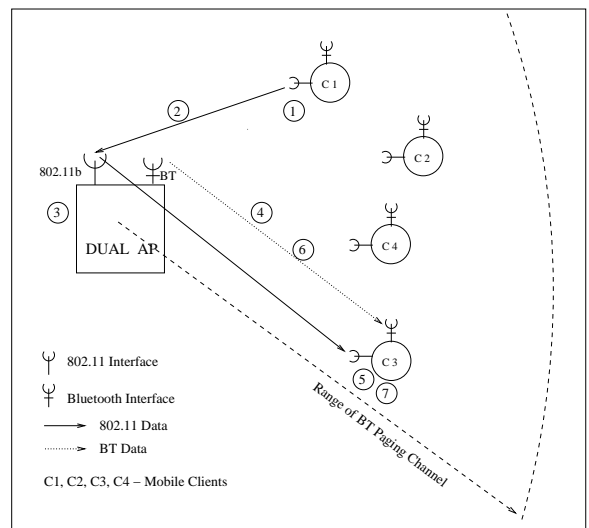


Fig. 2. On-Demand Paging Scheme

Our paging scheme works as illustrated in Figure 2 with the various steps shown as numbered circles. Assume a scenario with numerous wireless clients in an 802.11b based infrastructure network. The AP has both BT and 802.11b interfaces. A client has its 802.11b radios switched off when it is not involved in data communication. Each client monitors the BT paging channel continuously to sense paging signals. An application on client C1 wants to communicate with client C3. As soon as C1 detects this it switches its 802.11b radio on (Step 1), and sends data on the 802.11b channel (Step 2). The AP receives this data from C1 and knows the data is meant for C3 by looking at the destination IP. It then looks up and matches C3's IP address with its BT address (Step 3) and sends a *WAKE-UP* pag-

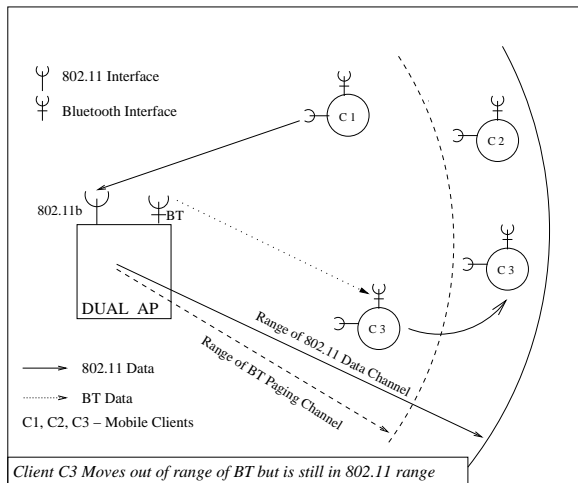


Fig. 3. Range inequality between BT & 802.11b Case : BT Range < 802.11b Range

ing signal to C3 via the BT interface (Step 4). C3 on receiving this paging signal from the AP switches on its 802.11b interface and is ready to receive data (Step 5). When C1 finishes sending the data it switches off its 802.11b radio. If the AP senses that all connections to and from C3 are closed it sends a *SLEEP* paging signal to C3 through its BT interface (Step 6). On receiving the *SLEEP* page from the AP, C3 shuts off its 802.11b interface (Step 7). The *SLEEP* page from the AP is important in case of range differences in the paging channel (BT) and the data channel (802.11b) and is explained in subsection B.

One advantage of our scheme is that since our implementation is done at the transport layer any application that uses TCP/UDP can use our framework without any modifications. Another advantage is that a mix of participating and non-participating clients are allowed. If a client (non-participating) requires minimum communication latency or does not have support for BT, it can still communicate with other clients that use the paging framework. The AP maintains a list of clients and whether they use the paging architecture or not.

One important aspect to consider is the difference in range of the two radios on the AP. Even though both 802.11b and BT radios are Class-0 devices (range > 100m), the ranges will rarely be identical. The range of the BT radio on the clients is not an issue as the client BT radios are used to only receive pages. The effective range for the whole framework is thus decided by the range of the BT radio on the AP which is over a 100m. The two possible cases for differences in range are :

A. BT range > 802.11b range

This case is not really an issue. An 802.11b client in an infrastructure network has to associate itself with an AP when it comes within range. Hence any client will know that it is out of 802.11b range if it is unable to associate with the AP.

B. BT range < 802.11b range

This scenario is illustrated in Figure 3. Client C1 is in BT range and also in 802.11b range, hence will not have any prob-

lems. Client C3, which was in BT range and actively communicating moves out of BT range, but is still within 802.11b range. To take care of this case the AP in addition to sending a wakeup page, also sends a sleep page to the client when all connections to and from the client are closed or timed out. Thus if client C3 was *actively communicating*, then went out of BT range, it will not receive a sleep page and will therefore have its 802.11b interface on. C3 can then switch to PSP mode to save power, but cannot utilize the paging architecture until it comes back within BT range.

Another scenario is when a *not communicating* client C2 moves out of range of BT, but is still in 802.11b range. To take care of this case the AP sends out periodic pages on the BT channel. As long as the BT client receives these pages it is in range of BT, and out of range otherwise. It then has the option to switch on its 802.11b interface in order to maintain connectivity in either CAM mode or PSP mode as desired. When C2 moves back in BT paging range, it can resume normal operation and use the paging scheme.

V. IMPLEMENTATION

Our power management scheme is implemented as a part of the operating system. Our prototype client is an iPAQ device running Linux. To ensure transparent invocation, we modified the Linux kernel to raise a signal whenever an application initiates a network connection. The paging scheme is implemented in two modules called the Paging Module (PM) and the Wireless Power Manager (WPM).

The PM listens to traffic on the BT channel and if it receives a wake up or a sleep page from the AP, it sends a signal to the WPM module to turn the 802.11b radio on or off respectively. Another function of the PM is to listen to periodic BT pages from the AP as described in Section B, in order to determine if the client is out of BT range.

The WPM module is responsible for sensing the need for the 802.11b interface to be turned on as signalled by the kernel whenever an application wants to communicate (outbound connection). Another case is when the AP has some data intended for the client (incoming connection). The AP sends a wake up page through BT to the client. The PM receives this page and sends a software interrupt to the WPM module to turn on the 802.11b radio. When the PM receives a sleep page from the AP it sends a signal to the WPM to turn off the 802.11b radio.

There are two low power sleep states, SS1 and SS2 defined in our implementation. A wireless card has two power components - power consumed by the radio (transmitter, power amplifier) and power consumed by the electronics (MAC layer processing, supporting circuitry). In *Sleep State 1* (SS1) only the radio part is turned off and no transmission or reception is possible. The card in SS1 still consumes some power due to the electronics still being on. *Sleep State 2* (SS2) is the more aggressive sleep state, saving more power at the cost of added latency. It switches off the interface and thus puts the card in a complete sleep state. The latency and energy values to switch between awake state and SS1 or SS2 are given in Table III.

TABLE III
LATENCY AND ENERGY TO SWITCH STATES

| Transition | Latency | Energy |
|--------------|---------|--------|
| SS1 to Awake | 20ms | 20 mJ |
| Awake to SS1 | 20ms | 20mJ |
| SS2 to Awake | 0.81s | 490 mJ |
| Awake to SS2 | 1.71s | 700 mJ |

A. Platform and Test Setup

We chose Compaq iPAQ 3870 PDAs with integrated bluetooth, and Cisco PCM-350 wireless cards for our experimental setup. We use Familiar Linux as the operating system. The Access Point (AP) is a Linux laptop containing a Prism II chipset based card – Compaq WL-100, which can be configured in a “Master” mode to mimics an AP’s functionality. The Bluetooth card used was a Anycom CF-2001 with the BlueZ stack.

We use a PCI-6110E card from National Instruments for data aquisition to measure power. The total power was measured by sensing the current to the iPAQ by using a series resistor and measuring the supply voltage. To isolate the power consumption of the 802.11 card alone we use a PCMCIA extender card.

VI. RESULTS

A. Power & Latency Results

To quantify the benefits of our scheme, we compare the power consumption of our scheme to the operating modes specified in the IEEE 802.11b protocol i.e.CAM and PSP.

We first measured power for the various components of the system by turning them on one by one. Power measurements were taken with the screen backlight switched off. Table IV gives the power consumption figures of individual components like BT, 802.11b card in PSP, CAM, in Sleep State 1 (SS1), and in Sleep State 2 (SS2). All the figures given in Table IV are idle power figures, with no transmission or reception taking place.

As seen from Table IV, the idle power consumption of BT is 40mW compared to 1.31W (CAM Mode) or 0.39W (PSP Mode) of the 802.11b card. From Table IV the base cases for comparison are cases 2 and 3. Case 3 is the lowest power consuming mode of operation in the present power modes of the 802.11b protocol. Our paging scheme power consumptions are given as cases 5 and 6 (in addition to the BT power).

TABLE IV
POWER CONSUMPTION OF INDIVIDUAL COMPONENTS

| | Configuration | Power |
|---|---------------------|--------|
| 1 | iPAQ only | 1.30 W |
| 2 | 802.11b card in CAM | 1.31 W |
| 3 | 802.11b card in PSP | 0.39 W |
| 4 | BT only | 0.04 W |
| 5 | 802.11b card in SS1 | 0.23 W |
| 6 | 802.11b card in SS2 | 0.02 W |

The total power savings compared to CAM and PSP are shown in Figure 4 with the second columns showing the power savings for the wireless card alone.

In case of networks, energy savings usually have an associated tradeoff - added latency. In our paging based scheme, the two sleep states SS1 and SS2 have different latency penalties, as given in Table III earlier. In addition to this switching latency, there is also an added latency to send the BT page which we measure to be 20ms. The total latency to switch to SS1 is thus 20ms (802.11b switching) + 20ms (BT paging) = 40ms. In case of PSM of 802.11b, power saving is achieved by turning off the radio and turning it on periodically, where the period is called *beacon period*. Therefore, the average latency for PSM is 50ms and a maximum of 100ms, for a common *beacon period* of 100ms. If the radio is woken up even less frequently then the power saving increases, but at the cost of further increased latency. Our scheme thus saves power as well as reduces latency compared to PSM. The total latency to switch to SS2 will be 0.81s (802.11b switching) + 20ms (BT paging) =0.83s.

B. Energy savings for actual network load

To quantify the energy savings in case of actual wireless network traffic, we performed energy measurements for synthetic workloads. Traffic patterns were generated to depict a typical FTP and an SSH session to an isolated server. The FTP traffic, ftp1, was generated such that the client logged on to this server, downloaded and uploaded a few files of varying sizes. There were various such download and upload combinations in the script including varying periods of inactivity, in order to simulate actual network traffic. The SSH traffic pattern, ssh1, was also generated similarly. It executed various commands including directory listing, copying, renaming, and editing files intermixed with periods of inactivity. These traffic pattern scripts were then executed for the various power modes and energy consumption was measured. Figure 5 shows the normalized energy consumption for these traffic patterns - ftp1 and ssh1.

These are small scripts that execute for about 5-8 minutes. The energy consumption for our scheme for traffic pattern – ftp1 in SS1 is 23% less than CAM and 6% less than PSP. The

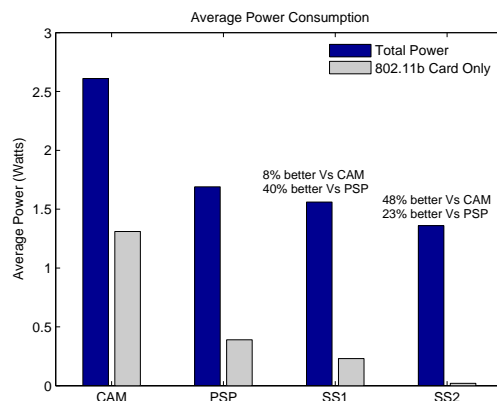


Fig. 4. Power Savings Vs CAM and PSP

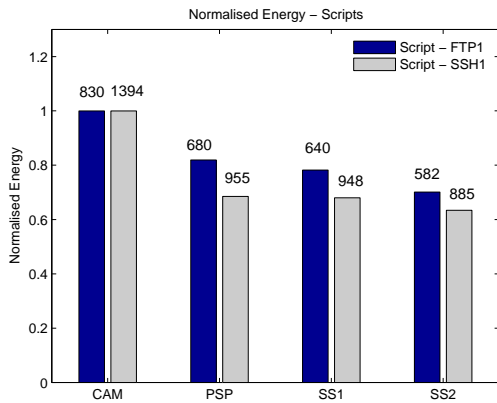


Fig. 5. Energy Savings for an FTP & SSH session - ftp1,ssh1. Actual Energy values (Joules) depicted above columns

energy consumption in the case of SS2 is 30% less than CAM and 14% less than PSP. As expected, longer periods of inactivity enable the wireless interface to be switched off for longer periods of time and hence greater energy savings. The total added end-to-end latency compared to PSP for switching to SS1 was in fact measured to be 0.5s less, while for SS2 it was greater by 4s. Compared to CAM the added latency of switching to SS1 and SS2 was 2s and 5s respectively.

We believe that this latency, especially in the case of switching back from Sleep State 2 (SS2), is due to the wireless cards taking a long time to switch modes. Bahl *et al.* have mentioned in [15] that 802.11b Next Generation cards are especially designed to support quick mode changes with latencies of the order of 15-20ms. Also note that, when 802.11b cards are switched off and then switched back on in Linux, the kernel raises a media connect notification causing additional latency before the card can actually start communicating. This latency is an artifact of implementation and we are working on ways to eliminate it. The end-to-end latency of our framework thus will reduce considerably using next generation 802.11b cards.

Another important performance parameter in case of networks is throughput. The throughput offered by our scheme is the same as offered in the Constant Awake Mode (CAM) mode of operation. The additional latency as explained in Section B above is a setup latency and occurs only at the beginning of a session. The WPM module after sensing that an application wants to use the wireless interface switches the 802.11b radio on and sets it to CAM mode. The maximum throughput equivalent to CAM mode is therefore available.

BT and 802.11b both operate in the same unlicensed 2.4 Ghz band. In addition to the fact that both BT and 802.11b were designed to be robust against interference, our BT paging channel generates limited interference since the amount of data being sent on it is low and is only sent by the AP. A BT paging packet is of the order of a few kilobits and takes less than 10ms to transmit. We validate this claim of negligible interference through real life experiments. Our measurements for an FTP session while simultaneously sending BT messages to the client do not

show any perceptible packet loss.

VII. CONCLUSION

BT radios are low power, making them ideal for use in paging. Manufacturers are already proposing integrated AP containing both BT and 802.11b interfaces. Our implementation has advantages such as application independence, high throughput and low latency. Our scheme leads to considerable amount of power savings ranging from 23% to 48% over the present 802.11b standard operating modes, which is especially attractive in case of larger networks. We have shown the energy saving of our implementation for various TCP/IP applications such as file transfer and remote sessions.

In future, this scheme may also be applicable to large sensor based networks by supporting the network using separate control and data channels, thus allowing for greater adaptation to application needs in energy efficient ways.

REFERENCES

- [1] M. Stemm and R. H. Katz. Measuring and reducing energy consumption of network interfaces in hand-held devices. *IEICE Transactions on Communications*, E80-B(8):1125–31, 1997.
- [2] S. Cho, Power Management of iPAQ, USC ISI. <http://pads.east.isi.edu/presentations/misc/sjcho-pm-report.pdf>.
- [3] IEEE 802.11b/D3.0 Wireless LAN Medium Access Control(MAC) and Physical(PHY) Layer Specification. High Speed Physical Layer extension in the 2.4Ghz band, 1999.
- [4] IEEE 802.1x-2001. IEEE standards for Local and Metropolitan Area Networks, 1999.
- [5] D. Bertozzi, A. Raghunathan, L. Benini, and S. Ravi. Transport protocol optimizations for energy efficient wireless systems. In *Design Automation and Test in Europe*, 2003.
- [6] S.A. Akella, R.K. Balan, and N. Bansal. Protocols for low power. Technical report, Carnegie Mellon University, 2001.
- [7] C.E. Jones, K.M. Sivalingam, P. Agrawal, and J. Chen. A survey of energy efficient network protocols for wireless networks. *Wireless Networks*, 7(4):343–358, 2001.
- [8] E.S. Jung and N. Vaidya. An energy efficient MAC protocol for Wireless LANs. In *INFOCOM*, pages 1756–1764, 2002.
- [9] Y.C. Tseng, C.S. Hsu, and T.Y. Hsieh. Power saving protocols for IEEE 802.11 based multi-hop ad hoc networks. In *INFOCOM*, 2002.
- [10] C. Guo, L. Zhong, and J. Rabaey. Low-power distributed MAC for ad-hoc sensor radio networks. In *Globecom*, pages 2944–2948, 2001.
- [11] C. F. Chiasserini and R.R. Rao. Combining paging with dynamic power management. In *INFOCOM*, pages 996–1004, 2001.
- [12] E. Shih, P. Bahl, and M.J. Sinclair. Wake on Wireless: an event driven energy saving strategy for battery operated devices. In *Mobile Computing and Networking*, 2002.
- [13] C. Schurgers, V. Tsatsis, S. Ganeriwal, and M. Srivastava. Optimizing Sensor Networks in the Energy-Latency-Density Space. In *IEEE Transactions on Mobile Computing*, pages 70–80, 2002.
- [14] A. Salkintzis and C. Chamzas. An outband paging protocol for energy-efficient mobile communications. In *IEEE Transactions on Broadcasting*, pages 246–256, 2002.
- [15] P. Bahl, P. Bahl, and R. Chandra. MultiNet: Connecting to Multiple IEEE 802.11 Networks Using a Single Wireless Card. Technical report, Microsoft, 2003.