Linux-based Ultra Mobile PCs

Rajeev Muralidhar, Hari Seshadri, Krishna Paul, Srividya Karumuri
Mobility Group, Intel Corporation
Rajeev.D.Muralidhar@intel.com, Harinarayanan.Seshadri@intel.com
Krishna.Paul@intel.com, Srividya.Karumuri@intel.com

Abstract

Ultra Mobile PCs present a new class of challenges for system designers since they are perceived to be versatile, low power devices and yet with the full functionality of larger handhelds/laptops. User experience of being ubiquitously connected and access to data anytime and anywhere is a fundamental requirement for this class of devices. However, access to data across wireless interfaces has a direct impact on battery life of such handheld devices. This paper presents detailed analysis of some of Linux file systems and data access across wireless. Based on our observations and analysis, we make some key design recommendations for file systems for Linux-based Ultra-Mobile PCs.

1 Introduction

The new generation of Ultra-Mobile PCs offer consumers significantly better capabilities than ever before to access the Internet, be productive and enjoy their favorite digital content while on the go. Consumers now want and expect access to their personal data and the Internet no matter where they are around the world, and take their laptops/ultra-portable PCs with them. Sharing data and content is an expected feature of any new technology—users want to be connected with the important people in their lives anytime, everywhere they go and want technology to make their lives easier. Some of the important technology features of such Ultra-Mobile PCs (UMPCs) are:

- **Full PC and Internet capabilities**: Full PC capability capable of running mainstream OSes like Linux and Windows, allowing consumers to run familiar applications.
- **Location Adaptibility**: Personalized information and services based on location, environment recognition, and adaptability and interaction with other devices at home, office, or in an automobile while driving.
- **Anytime connectivity**: Connectivity in several ways such as WLAN, WWAN, WPAN, WiMAX, etc. enabling "always reachability" via email, IM, or VoIP.
- **Ultra mobility**: Small, thin, light form factor with high battery life.

In addition to the more generic UMPCs which can essentially compute and communicate with rich productivity features and anytime, anywhere data access, there are other categories of ultra-mobile devices that are targetted to specific usages such as ruggedness (for application in environmental sciences, health/medicine, etc.), affordability (such as education devices), etc.

Regardless of the targeted usage or form factor of such devices, one of the fundamental visions driving pervasive computing research is access to personal and shared data anywhere and anytime. In many ways, this vision is close to being realized with multipe wireless connectivity options such as WLAN, WWAN, WPAN, and the upcoming WiMAX connectivity options to such small, mobile devices. Additionally, with several devices becoming the norm at home, sharing data between these devices in an efficient manner is an important aspect of usage, and it is important that the underlying platform and system support for such usages is done in an energy efficient manner. For example, it is quite likely that a home is equipped with multiple devices such as a desktop, media box (possibly with network connectivity options), home entertainment system with connectivity to a desktop/media storage, laptop, high-end cell phones, and also ultra-mobile PCs, all connected to the Internet through an external broadband connection and internally via high throughput wireless, like the IEEE 802.11n. This is becoming quite common in some of the economies of the world today. In such a scenario, a UMPC can be used to play locally stored media on the home entertainment device. Alternatively, it could be used to play remote digital media from set top box/desktop locally on the UMPC via wireless/streaming.

In such scenarios, we believe that substantial barriers remain to pervasive, energy efficient data access across wireless. Some of the key challenges are:

- **Low power platform design**: Although devices such as cell phones have matured to provide long talk time and standy battery lives, they lack the full featuredness expected in order to run producitivity applications, mainstream OSes, etc. We believe that the platform design of UMPCs pose significant challenges in order to meet the battery life and full PC capability expectations.
- **Low power communication / connectivity options**: Most connectivity options such as WLAN, WPAN, WWAN are power hungry. Usage scenarios such as media streaming over wireless connections will pose significant limitations on battery life since power-hungry network and storage devices tax the limited battery capacity of UMPCs.
- **Disconnected operation and access to data via local/network caches**: Disconnected operation is a way of life in wireless networks. Some research file systems have explored the use of local and network caches for faster access to data.
- **Energy efficient data sharing and file systems**: Power efficient file systems are critical since mobile data access performance can suffer due to variable storage access times caused by dynamic power management, mobility, and use of heterogeneous storage devices/connectivity options.

This paper focuses on the last aspect above, namely, energy efficiency in file systems, and makes the following contributions.

1. We analyze the platform power consumption of a Linux-based Ultra-mobile PC during different scenarios and quantify the impact of different platform components towards total platform power.
2. We then analyze the popular Network File System (NFS) for some of the common UMPC usage scenarios and identify the power bottlenecks.
3. Subsequently, we analyze one of the recent energy efficient file systems, BlueFS, and understand its impact on platform power.
4. Finally, based on our experiments and analysis, we make recommendations that we believe are critical to designing low power Linux-based ultra-mobile PCs.

This paper is organized as follows: Section 1 is this introduction. Section 2 reviews some of the related work. Section 3 describes the experimental analysis we performed with NFS and BlueFS. Subsequently, in Section 4, we make key observations and design recommendations. We finally conclude the paper with a summary and areas of future work.

2 Related Work

Energy efficiency in file systems is typically not a primary design consideration; this is true for distributed file systems as well. However, several recent projects target energy reduction in local file systems. In [5], the authors provided interfaces to applications to create energy-efficient device access patterns. Moving down the operating system stack, one of the primay components of a file system is its cache—file systems differ substantially in their choice and use of cache hierarchy. For example, NFS [2] uses only the kernel page cache and the file server. The Andrew File System (AFS) [4] adds a single local disk cache; however, it always tries to read data from the disk before going to the server. Coda [1], which is also designed for mobile computing, much like AFS, uses a single disk cache on the client; this cache is always accessed in preference to the server.

BlueFS [3] is different from other file systems in that it uses an adaptive cache hierarchy to reduce energy usage and efficiently integrate portable storage. BlueFS further reduces client energy usage by dynamically monitoring device power states and fetching data from the device that will use the least energy. BlueFS borrows several techniques from Coda including support for disconnected operation and asynchronous reintegration of modifications to the file server. BlueFS also can proactively trigger power mode transitions of storage devices that can lead to improved performance and energy savings. As reported in [3], mechanisms like this can greatly improve energy efficiency on the platform as a whole.

There has been a lot of analysis of power and performance of handhelds, but we believe this is the first analysis of platform power and power efficiency of full-featured ultra-mobile PCs.

3 Experimental Analysis

The experimental setup used consisted of a Linux-based Ultra-mobile PC that was instrumented to enable power measurements using a FLUKE Data acquisition system, 2680A/2686A Data Acquisition System. The following components were measured for power in different scenarios—processor, memory, chipset (GMCH), IO Hub (ICH), Hard drive (PATA), Buses (PCI-E, USB), and peripherals (Audio, WLAN card).

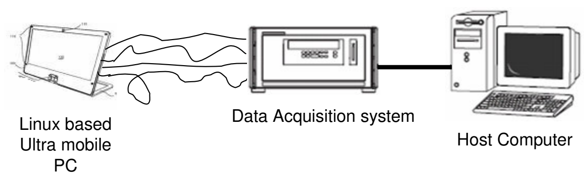

Figure 1: Experimental Setup

The experimental setup is shown in Figure 1 and the system configuration used for the UMPC is shown in Table 1.

The following experiments were performed:

- Idle state
- Audio playback over NFS
- Audio playback over BlueFS
- Rich Media (high-definition movie) playback over NFS
- Rich Media (high-definition movie) playback using BlueFS using local hard drive as cache
- Rich Media (high-definition movie) playback using BlueFS using USB drive as cache

4 Observations and File System Design Recommendations

Based on our analysis of NFS and BlueFS, here are a few observations:

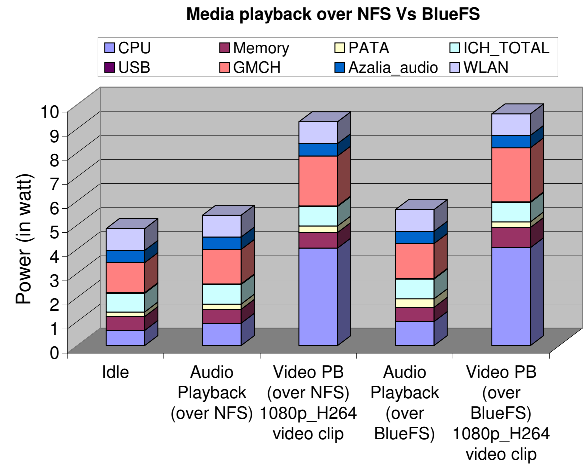

Figure 2: Performance of Media playback over NFS and BlueFS over the wireless link

1. Idle power on the platform was surprising higher, around 4.5W (*Figure 2*), as compared to other OSes which were around 3.2W; a closer look at the data reveals that the idle power for the WLAN component is as same as that of power it consumed when its exercising the work load, leading us to believe that we had an inefficient driver.

2. Another interesting aspect of the data was that the CPU power drawn when the system was idle was about 0.5W and a closer look at the processor's C3 residency state revealed that the CPU has C3 residency of only about 75%. Ideally this should be around 90–95% to get maximum platform idle power. We believe this is a critical piece to resolve to get better power for Linux based platform.

3. In addition to the impact of CPU residency, it is very important that all the devices in the platform support a low power idle state. As seen in our observations above, an inefficient device driver can spoil overall platform idle power and we need to make sure we have well optimized devices and drivers for a low ideal power. In the observation above, if we consider the ideal idle and transmit power for WLAN device, the idle power of the platform improves. This is shown in *Figure 3* which considers the projected ideal WLAN power and shows the performance of Media playback over NFS and BlueFS.

4. **NFS power bottlenecks:** The WLAN power consumption is among the top 3 in the list, next only to CPU and GMCH. This is as expected, all the

Component	Configuration
CPU	Intel® Core Duo 800 MHz
Memory	512 MB DDR2
ICH	ICH7
GMCH	Intel® 945GM Express Chipset Family
Audio	Intel® High Definition Audio
WLAN	Intel® 3945 ABG Network Connection

Table 1: Configuration of Linux-based Ultra-Mobile PC used for experimental analysis

NFS transactions are network centric. Assuming the network stack and the device are well optimized for the power, better platform power saving can be attained through file system specifically designed for low power.

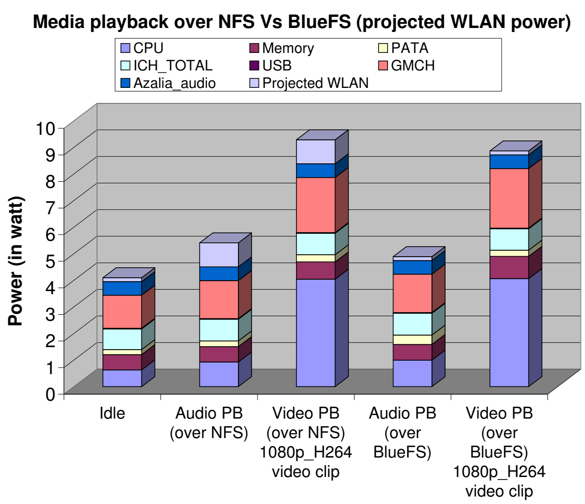

Figure 3: Performance of Media playback over NFS and BlueFS (with projected WLAN power)

4.1 BlueFS impact on Platform Power

1. BlueFS did show more effiency in terms of platform power for the given workload, as compared to NFS. The latency did come down drastically for BlueFS and this clearly brought down the net power consumed by the platform for a given workload to complete.

2. Caching mechanisms also helped in improving the user experience, and overall usage scenario. However, the choice of the caching device does have an impact on the system power. For example, from *Figure 4* BlueFS running over a local disk as cache consumed less power compared to USB drive as cache, suggesting that if we have an optimized local solid state drive like Robson flash drive [6] as cache for BlueFS will have significant impact on the platform power.

3. We did not see any significant impact on the write performance as compared to NFS though one possible optimization could be to re-evaluate the write to many strategy in terms of power efficiency.

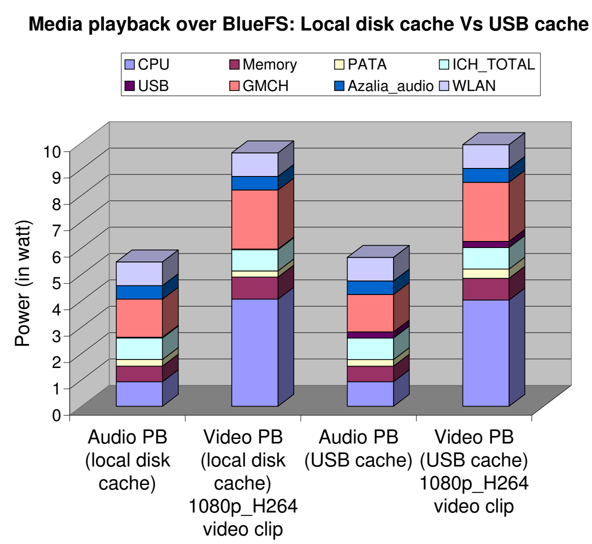

Figure 4: Performance of Media playback over BlueFS using local disk cache and USB cache

4. We also observed that while the Wolverine user daemon dynamically selects the device to read or write on run time, it does not do anything to the devices that are not being used. One possible power saving option would be to modify Wolverine to power down the devices that are not being selected for read or write operation.

5. Caching being a critical piece of BlueFS efficiency, proper focus needs to be made on the cache hierarchy selection. Other power efficient caching technologies like co-operative caching needs to be explored for more optimized power saving.

5 Summary and Future Work

Ultra-Mobile PCs present a new class of challenges for system designers since they are perceived to be high performance, low power devices and yet with the full functionality of larger handhelds/laptops. User experience of being ubiquitously connected and access to data anytime and anywhere is a fundamental requirement for this class of devices. However, access to data across wireless interfaces has a direct impact on battery life of such handheld devices.

This paper presented detailed analysis of the Network File System (NFS) and Blue File System (BlueFS) for specific usage scenarios targetting network data access across wireless interfaces and makes. Based on the power and performance analysis of these file systems, we made some key design recommendations to designers of file systems for Linux-based Ultra-Mobile PCs.

We plan to extend this work further in the following areas:

1. Evaluate BlueFS for multiple storage devices—USB, newer storage mechanisms like Robson Flash memory recently introduced on Intel's newest platforms. We believe that although BlueFS has an excellent mechanism for caching data, the choice of the device will be important in total platform power consumption.

2. Evaluate other wireless interfaces and usage models, specifically mobile digital TV, IP TV, since such scenarios would be critical for ultra-mobile PCs.

6 Acknowledgements

We would like to thank Bernie Keany for his valuable discussions and inputs, Adarsh Jain, Arvind Singh and Vanitha Raju for all their help with power measurements and analysis on the Ultra-mobile PC platforms. We would also like to thank Ananth Narayan for the tools he provided for our analysis.

References

[1] J.J. Kistler and M. Satyanarayanan, *Disconnected operation in the Coda File system*, ACM Transactions on Computer Systems, 10(1), February, 1992.

[2] Network Working Group, *NFS: Network File System*, Protocol specification, March 1989. RFC 1094.

[3] Edmund B. Nightingale and Jason Flinn, *Energy-Efficiency and Storage Flexibility in the Blue File System*, Proceedings of the 6th USENIX Symposium on Operating Systems Design and Implementation (OSDI), San Francisco, CA, December, 2004.

[4] J.H. Howard, M.L. Kazar, S.G. Menees, D.A. Nichols, M. Satyanarayanan, R.N. Sidebotham, and M.J. West, *Scale and performance in a distributed file system*, ACM Transactions on Computer Systems, 6(1), February, 1988.

[5] T. Heath, E. Pinheiro, and R. Bianchini, *Application-supported device management for energy and performance*, In Proceedings of the 2002 Workshop on Power-Aware Computer Systems, pages 114–123, February, 2002.

[6] Michael Trainor, *Overcoming Disk Drive Access Bottlenecks with Intel Robson Technology*, Technology at Intel Magazine, Volume 4, Issue 9, December, 2006.

Proceedings of the Linux Symposium

Volume Two

June 27th–30th, 2007
Ottawa, Ontario
Canada

Conference Organizers

Andrew J. Hutton, *Steamballoon, Inc., Linux Symposium,*
Thin Lines Mountaineering

C. Craig Ross, *Linux Symposium*

Review Committee

Andrew J. Hutton, *Steamballoon, Inc., Linux Symposium,*
Thin Lines Mountaineering

Dirk Hohndel, *Intel*
Martin Bligh, *Google*
Gerrit Huizenga, *IBM*
Dave Jones, *Red Hat, Inc.*
C. Craig Ross, *Linux Symposium*

Proceedings Formatting Team

John W. Lockhart, *Red Hat, Inc.*
Gurhan Ozen, *Red Hat, Inc.*
John Feeney, *Red Hat, Inc.*
Len DiMaggio, *Red Hat, Inc.*
John Poelstra, *Red Hat, Inc.*