Radical Appropriation: The Configurations of Wireless Networking in a Community Group

David W. McDonald
The Information School University of Washington
dwmc@u.washington.edu

Abstract. Forms of appropriating activity range from adopting a technology largely as it is to radically transforming it to make it perform some other function. This field study describes how a wireless community group appropriated commodity wireless networking and transformed it to serve their goals. The study characterizes several configurations which were promoted by community leaders, the varying drawbacks which each led to a new, subsequent, configuration. The results identify a set of design considerations for hardware devices that seem to facilitate appropriation.

1 Introduction

A hack, in the traditional sense of the word, is an appropriation that illustrates some deeper understanding of a technical system. The reconfiguring of a device, making a device specifically designed to do one thing into a device that does something else, has slowly become more commonplace. At this time no one has made an automobile Anti-Lock Braking System play MP3s, but some MP3 players have been reconfigured as Linux computers (Leach, Carne et al. 2003).

Wireless networking, WiFi, 802.11a/b/g, has been fertile ground for a range of appropriation that illustrates creativity and in-depth understanding of a technical system. War driving (Byers and Kormann 2003), packet sniffing, breaking WEP (Wireless Encryption Protocol) (Fluhrer, Mantin et al. 2001; Rager 2001), represent a range of sophisticated hacking activities in which almost anyone can participate given access to the right software. The software lowers the barrier to participation in a technically sophisticated activity, but someone had to create that software first. Someone, some group, had to take their deep understanding of the technical system and embody it in a software artifact so that less sophisticated participants could benefit.

This study describes appropriation activity in a wireless community group, the Northwest Wireless Group (NWG). The focus is on the radical appropriation of commodity wireless equipment to create a wireless backbone and wireless community access. The fieldwork describes how individuals come together to appropriate wireless technology and solve difficult infrastructure construction and maintenance problems. The results focus on a set of design considerations that seem to facilitate appropriation of hardware devices.

Appropriation is commonly framed as a type of adoption of a piece of software or a complex system. The concept of adoption is described and studied in the computing and information technology literature (e.g. (Markus and Connolly 1990; Francik, Rudman et al. 1991; Orlikowski 1992; Levine and Rossmoore 1993; Kraut, Cool et al. 1994)) and the results can often be reexamined as appropriating activity. Studies of the organizational implementation (deployment) of complex systems like Group Decision Support Systems (GDSS) or Enterprise Resource Planning (ERP), have used 'appropriation' as a term to describe activities of users which are outside a normative model of system usage (Orlikowski and Robey 1991; Galegher and Kraut 1992; DeSanctis 1993; DeSanctis and Poole 1994; Olesen and Myers 1999). Often these are uses outside the normative model of work activity as understood by the system designers. Using appropriation in this way is powerful because the appropriation activities illustrate how users bridge the gap between their actual needs and the needs as implemented in the system.

This paper first describes the primary difference between the default wireless networking context and how the community group reshaped the context by defining a different networking model. This model frames three technical configurations that the community developed over several years. These configurations illustrate a number of difficulties that must be overcome by individuals who appropriate hardware devices.

2 Appropriating a Context: Reconceptualizing WiFi

The wireless industry has largely conceptualized WiFi as a localized service. Generic access points simplify the redistribution of Internet service with built-in software. However, the broader wireless community recognizes that WiFi technology has the potential to do more than provide simple hotspot service. By reconceptualizing how WiFi technology can be used, a larger network, not just hotspots, can be constructed. A wireless Metropolitan Area Network (MAN) can be developed by connecting nodes of a network through wireless connections (Flickenger 2001), creating a wireless backbone. The Northwest Wireless Group (NWG) is one group that led early efforts to develop infrastructure and software to achieve a wireless MAN.

The insight that a wireless MAN can be constructed from default WiFi components is an appropriation of the context of wireless networking. It illustrates a gap between what users want to be able to do with wireless equipment relative to the default design for wireless networking as promoted by the equipment manufacturers. In this section, we illustrate the difference between the default WiFi context and the context created by NWG members.

2.1 The WiFi Backbone Model

The challenge in reconceptualizing wireless networking is how to connect a large number of standard WiFi nodes without using wires. This requires moving away from the idea that a node in the network is composed of a single piece of equipment with a single WiFi compliant radio. Nodes in an NWG network must be more complex.

A node for the NWG network is composed of at least four pieces of equipment; a computer, two access points and a directional antenna. A small computer serves to route data and monitor the node. These computers often run a version of Linux. In a basic node, one AP is used to provide local (omni-directional) connectivity for wireless devices in the physical vicinity. A second AP provides a directional connection to another node in the wireless network. The directional connection is provided by directional antennas at each end of a link.

A minimal node provides a limited type of connectivity. That is, only supporting a single directional backbone link creates network design problems. Thus the reconceptualized network model supports nodes of differing complexity, with different levels of connectivity to the network. Figure 1 is a diagram of how wireless nodes are connected through directional RF links. This figure illustrates the different levels of connectivity and the general NWG network model.

While the WiFi backbone model is conceptually possible, a fundamental challenge in implementing this network architecture was to develop a model for the basic NWG node. The WiFi standard included and ‘ad-hoc’ or ‘peer-to-peer’ mode which could provide the needed directional link, but in practice different manufacturers implemented this feature slightly differently. As a result, a node model, or node configuration, could help galvanize participation around equipment that would interoperate and simplify how nodes were connected.

3 Radical Appropriation: Finding the Right Configuration

The NWG community explored a number of possible node configurations over the past few years. The four configurations that will be discussed generally overlapped with each other. The community never focused solely on a single configuration. Often configurations were explored and fielded as experiments to test both reliability and to provide service. Each possible configuration was promoted to the general population of participants. In most cases NWG members purchased equipment, explored the capabilities and contributed some way to making a working solution. In two of the cases, severe hardware or software constraints resulted in the configurations being abandoned.

3.1 The Airport/Orinoco RG-1000 Configuration (c. 2001-2002)

In 2000 Apple Computer introduced the Airport wireless access point to the public and initiated the low-cost public acceptance of the 802.11b “WiFi” wireless standard. Wireless networking had been available in various forms prior to the introduction of the Airport, but equipment was expensive and not well supported by the most prevalent computer operating systems. The Airport included a 56K modem that could be configured with NAT (Network Address Translation) and bridging so that several computers on the wireless connection could also use Internet service through the modem. With the introduction of Airport almost anyone could quickly set up a Local Area Network (LAN) in the home or office.

Configuring the Airport[1] to boot a different operating system is not trivial.[2] In particular, the Airport needed support from a second computer running Linux or similar Unix variant. This host machine needed to have a Network File System (NFS) partition created that the Airport would use for a remote boot and for a dedicated file system. On this NFS partition a special ‘.nbi’ (Net Boot Image) would be placed where it would be available for the Airport to read. Lastly, the firmware in the Airport needed to be “flashed” (semi-permanently modified) with new firmware so that the device would boot from the appropriately named .nbi file off the NFS partition of the host machine. A Linux/Unix system administrator with two or more years of experience would find this type of configuration relatively simple. An average user would find this configuration difficult.

---

[1] In the following discussion “Airport” is used in the general sense of any access point that relied on the same internal hardware.

[2] The initial insight about the Airport and the first effort to get Linux to run on the device is credited to Till Straumann. See http://www.slac.stanford.edu/~strauman/pers/airport/airport.html for more details on how to install and configure Linux for an Airport.

The Airport/Orinoco RG-1000 configuration was problematic because it required an additional machine to host the remote boot. But as well, the device had very limited memory. With only 512K of flash memory and 4MB RAM the device was barely able to run Linux. Routing tables, simple network monitoring code and other configuration data all take space in the limited memory. Despite efforts to strip Linux to the bare minimum, memory problems were common, often causing directional links to fail.

3.2 Pebble/Soekris v.1 Configuration (c. 2002-2003)

Frustrated with the installation problems and physical limitations of the Airport/RG-1000 the broader wireless community began exploring solutions using Linux and embedded computers. The Pebble[1] Linux distribution is conceptually similar to that of the Airport Linux; a minimalist distribution designed to be able to run on a range of small computers with limited memory, disk space, and with at least one wireless card.

As NWG became frustrated with the limitations of the Airport/RG-1000 configuration, they began developing nodes around embedded computers. Soekris Engineering developed two computers, the net4511 and the net4521, which were well engineered and highly capable. A single net4521 could potentially support up to four wireless connections as well as an Ethernet link to the Internet. Conceptually, Pebble on a Soekris computer provided an ideal NWG node; potentially a "Class A" node in a single box (see Figure 1).

The Pebble/Soekris configuration was fielded by several participants and problems started to emerge. In general, the Pebble/Soekris configuration was more stable and reliable than the older Airport/RG-1000 configuration. With this reliability, people were more willing to deploy NWG nodes in locations with limited access like roofs and other outdoor locations. NWG participants began exploring how to inexpensively weatherize node equipment. Attempts that used Tupperware, silicone glue, shrink wrap, and shrink tubing all met with varying degrees of success. While the net4521 can support up to three wireless cards, this is not practical because the amount of interference caused by RF bleed from the cards and the pigtail connectors seriously decreases overall throughput.

While the cost of a complete Pebble/Soekris node was less than commercial grade equipment, the combination of cost and complexity of set-up limited the number of NWG participants who would select this equipment. As well, a third alternative, based on an inexpensive consumer grade access point, diverted attention and effort from the Pebble/Soekris configuration.

---

[1] Terry Schmidt, a founder of NYCWireless in New York, initiated and led the Pebble project through several early and critical distributions. The current Pebble distribution is available at http://www.nycwireless.net/pebble/

3.3 WRT54g Configuration (c. 2003-2004)

In 2003 the Linksys Corporation began distribution of a new wireless access point called the WRT54g. Linksys had been an active manufacturer of consumer grade 802.11b equipment that was relatively simple to install and quite popular with the average consumer. The WRT54g was Lynksys' first attempt at an 802.11g access point. As a consumer grade AP the WRT54g was relatively cheap and provided backward compatibility with the 802.11b equipment that was already widely deployed. Thus the WRT54g represented a natural upgrade path from one generation of wireless standard to the next.[1]

Like many APs and routers, the WRTs use a built-in web server and a set of web pages to facilitate set-up and monitoring. Some of the WRT pages collect data from web form fields and pass the data directly to a Linux command line with little or no error checking. Because most command line shells support multiple commands per line, this oversight allowed a user to enter a valid parameter followed by a command separator and a subsequent set of commands. One of the web pages allowed the user to pass parameters to the 'ping' command and provided a large text field area to view the response. With this simple exploit and a way to see the results of a general command response, the community systematically explored the version of Linux and the available command tool set distributed in the WRT. The community quickly realized that the WRTs had some specific limitations, but it had flash memory and that could be changed.

The broader wireless community began efforts to create a custom version of the WRT firmware that would include the tools that were need to remake the WRT into more than a simple AP. Through systematic exploration and some trial and error the community developed wrt54g_tools, a set of software tools specifically for creating a valid firmware image. Despite this success, two problems remained. First, Linksys had used a GPL code base and was reluctant to release the code they had developed. Second, the WRT used a Broadcom wireless card which was designed for the OEM (Original Equipment Manufactuer) marked and Broadcom has never released the drivers for the wireless card.

Linksys eventually released their code to the community, but the WRT54g configuration did not take off. The use of a Broadcom OEM wireless card with proprietary drivers meant that the community was prevented from understanding the underlying hardware well enough to appropriate its latent functionality. Also, equipment manufacturers are constantly looking for ways to simplify their design and produce a product more cheaply. In the case of the WRT, each product revision was like dealing with a new unique piece of equipment. Changes to the hardware, introduced by a manufacturer can be handled in software, such that to an outsider, the product looks the same, the default functionality is the same, the

---

[1] The relatively common availability of 802.11b and 802.11g equipment is largely a function of the fact that they operate in the same frequency range that simplifies the engineering of the hardware (transmitters, receivers, antennae). The other WiFi standard, 802.11a, has never achieved broad public acceptance.

user interface is the same, while the device hardware could be completely different. In the case of the WRT changes happened frequently and were dramatic enough that small revisions in the underlying hardware made the community based firmware incompatible from one board revision to the next. For the average user, these differences are difficult to explain and an incomplete understanding can result in a modified device that is completely useless.

4 Facilitating Appropriation

The story of NWG is fundamentally a story of collaborative appropriation. It is collaborative in the way the technology is systematically explored and exploited. The collaboration spans the larger wireless community, the activities of the NWG group itself, and the small scale collaboration of individual members as they attempt to build and install network nodes. This study has focused on the technical trajectory of appropriation and the systematic exploration and reconfiguration of WiFi network equipment.

This story illustrates a number of key aspects for the design of technology that facilitates appropriation. In particular, the story highlights how appropriation is possible when devices have latent functionality that is identified and exploited by users. This latent functionality is easier to identify and exploit when users have access to patterns that illustrate aspects of the device design. Lastly, appropriation is possible when there is some configuration stability in the device. This stability allows users to communicate the appropriation to others and know that the modifications will most likely work on another's device.

5 Acknowledgments

This research would not have been possible without the NWG community and the participants' willingness to talk about the project. This research was supported by the University of Washington, Royalty Research Fund Grant 65-2805 and by contributions from Intel Corporation, Intel Research, Seattle.

6 References

Byers, S. and D. Kormann (2003). "802.11b Access Point Mapping." Communications of the ACM 46(5): 41-46.

DeSanctis, G. (1993). Shifting Foundations in Group Support Systems Research. Group Support Systems: New Perspectives.

L. M. Jessup and J. S. Valacich. New York, NY, Macmillan. DeSanctis, G. and M. S. Poole (1994). "Capturing the Complexity in Advanced Technology Use: Adaptive Structuration Theory." Organization Science 5(2): 121-147.

Flickenger, R. (2001). Building Wireless Community Networks, O'Reilly & Associates.

Fluhrer, S., I. Mantin, et al. (2001). Weaknesses in the Key Scheduling Algorithm of RC4. Lecture Notes in Computer Science, Springer-Verlag. 2259: 1-24.

Francik, E., S. E. Rudman, et al. (1991). "Putting Innovation to Work: Adoption Strategies for Multimedia Communication Systems." Communications of the ACM 34(12): 53 - 63.

Galegher, J. and R. E. Kraut (1992). Computer-Mediated Communication and Collaborative Writing: Media Influence and Adaptation to Communication Constraints. Proceedings of the 1992 ACM Conference on Computer-Supported Cooperative Work (CSCW '92), ACM Press.

Kraut, R. E., C. Cool, et al. (1994). Life and Death of New Technology: Task, Utility and Social Influences on the Use of a Communication Medium. The 1994 ACM Conference on Computer-Supported Cooperative Work (CSCW '94), Chapel Hill, NC, ACM Press.

Leach, B., D. Carne, et al. (2003). iPodLinux, iPodLinux Project.

Levine, H. G. and D. Rossmoore (1993). "Diagnosing the Human Threats to Information Technology Implementation: A Missing Factor in Systems Analysis Illustrated in a Case Study." Journal of Management Information Systems 10(2): 55-73.

Markus, M. L. and T. Connolly (1990). Why CSCW Applications Fail: Problems in the Adoption of Interdependent Work Tools. The 1990 ACM Conference on Computer-Supported Cooperative Work (CSCW '90).

Olesen, K. and M. D. Myers (1999). "Trying to Improve Communication and Collaboration with Information Technology: An Action Research Project which Failed." Information, Technology and People 12(4): 317-332.

Orlikowski, W. J. (1992). Learning from Notes: Organizational Issues in Groupware Implementation. The 1994 ACM Conference on Computer-Supported Cooperative Work (CSCW '92).

Orlikowski, W. J. and D. Robey (1991). "Information Technology and the Structuring of Organizations." Information Systems Research 2(2): 143-169.

Rager, A. (2001). WEPCrack.