

The Personal Router Whitepaper

David D. Clark and John T. Wroclawski
MIT Laboratory For Computer Science
Version 2.0 – March, 2001

Abstract: We describe the Personal Router project, a program of research currently underway within the Advanced Network Architecture group at the MIT Laboratory for Computer Science.

Executive Summary

The purpose of this project is to demonstrate a new paradigm for wireless network services. The motivations for this project are the following:

- Today, wireless service is either local to an institution (as in wireless LANs) or is provided as part of a regional or national service (as in cellular). But there could be other models. Individual providers of wireless service could put up a base station and offer service locally. Users could move among these providers and select a service based on requirements and prices. Just as we have over 5000 wireline Internet service providers today, many only serving a small area, we could have wireless coverage built bottom up.
- Today, we see a restricted range of wireless devices - a laptop PC with a wireless card, a cell phone augmented with data capabilities, a highly integrated device such as a pager. There is not an open market for new wide-area wireless devices, because such devices today are tightly bound, both technically and through service contract, to a particular wireless service provider. But there could be a wide range of new consumer devices if the proper interfaces and modules were available.

For this to work, two things are necessary. First, each of the broad mix of competing wireless services must be accessible to a wide variety of devices. What is needed is an overall system architecture that allows cheap, small, low-power consumer-level objects to access a wide variety of technically incompatible wireless services with ease, using open interfaces. Second, selection of a service among competing alternatives must be easy. For example, a manual process of selection and entry of a credit card would be too

burdensome to succeed. What is needed is a model for automated dynamic negotiation, based on rules provided by the users and providers, together with a workable economic model and a simple micro-payment scheme.

The goal of this research is to demonstrate two related innovations. The first is a framework for automated negotiation for access to wireless services. The second is a small hardware device that the user can carry, a *personal router*, which contains the necessary wireless transceivers, implements the access negotiation protocol, and provides a network connection for the other devices and appliances that a person might choose to carry. By creation of an open interface between this personal router and other devices a user might carry, we can create an environment for the development of new devices and applications.

The intellectual merit of this project is embodied in: a) the automated service negotiation framework, b) experience with systems that use policy constraints to guide automatic configuration; c) the related protocols for security and billing; d) the demonstration of the new application user interface paradigms implied by the personal router; and e) identification and definition of the interfaces between this device and the other consumer devices that can be connected to the Internet using it. The broader impacts are the possibility of creating a new market model for wireless service, based on small business investment and local competition, rather than service provision (only) by large, national-level corporations, and the development of an increased understanding of user pricing preference for communication services. Research that explores technology supporting alternative economic models is not likely to be done in private industry, and is thus especially appropriate for academic inquiry and public-sector support.

Project Description

1: Objectives and significance

This project is motivated by two visions—first, that widespread wireless access to the Internet can be achieved by a different economic model than the one currently in place, and second, that an open market for a new generation of wireless devices and applications can be created by such a wireless infrastructure.

The first vision is one that links economics and technology. Wireless access requires both a transceiver in the mobile device, and a set of base stations to which to communicate. How can that infrastructure of base stations come into existence? The investment model today is that high-speed base stations (e.g. wireless LANs) may be installed by private organizations within their own facilities, but in the wide area, service is provided by a large-scale provider who blankets a region or a nation with towers. It need not be this way.

An alternative is that small businesses or individuals put up local base stations, and sell access to the Internet within small regions. With a suitable economic market, demand would trigger deployment. These small providers, together with today's large-scale providers and dedicated institutional networks, would create a rich, responsive, and competitive infrastructure for wireless Internet access. Our goal is to provide a technical and economic framework in which to explore this option. For example, a number of universities have provided wireless access on their campus for the students and staff, and a large number of people have equipped themselves with wireless cards for lap-tops. Given this investment, and the results of our research here, small businesses near such a university could install wireless LAN base stations, and offer service to this pool of users. Service could be offered for payment, or as an inducement to visit a merchant (a restaurant, coin laundry, coffee shop, etc). Service can be offered in different tiers with pricing structure if the market develops in this way. While the final form of such a market is unclear, facilitating the experiment is the only way to learn.

Successful realization of this vision depends on a number of technical capabilities. First, it must be

possible for the customer to move transparently and dynamically between different providers and service zones. This implies that transport layer connections, security associations, and the like survive the transition of the user from one provider to another. These capabilities are not supported gracefully in today's Internet, though a number of current research efforts seek to address them.

More interesting is the need for this competitive, dynamic service environment to be made accessible to customers in comprehensible terms. It is not enough that the user have a choice between wireless service providers. To truly benefit, the user must be able to make this choice simply and intuitively, and to re-evaluate the choice frequently, as service offerings change and the user moves about. Mechanisms that involve manual intervention, detailed understanding of application service requirements or network QoS offerings, and similar complexities, are too burdensome to succeed. What is needed is an automated service selection mechanism, driven by a high-level intuitive capture of the user's current requirements. This mechanism, by considering the user's requirements as well as rules and service descriptions made available by providers, transparently and dynamically selects the most appropriate provider and offered service at any given time. The development of this selection framework, together with its supporting interfaces, mechanisms, and economic models, is a central objective of our research.

The second vision relates to the form and modularity of wireless user devices. Today, wireless LAN cards are inserted into laptops. Cellular wireless devices are either (like CDPD modems) packaged as laptop PCMCIA cards, or are semi-flexible but closed consumer devices, such as WAP-enabled mobile phones, or serve fixed applications such as pagers or traditional cell phones. Some of these devices are open at the Internet level, in that they can make a connection to any suitable server on the network. But they are not open in a way that permits a device manufacturer to produce a new device, and independently connect it into the wireless infrastructure. In contrast to a wireline network like Ethernet, where anyone can put an Ethernet interface on a device, the only way to put a cellular data radio into a device is to work with

the cellular service provider to get them to market, or at least support, the device.¹

An alternative outcome would be that anyone could design and manufacture a personal consumer device that has wireless access to the Internet, and bring it to market independently of the service provider, just as anyone can manufacture a telephone. We believe that if the market for wireless devices were open, that all sorts of devices with different functions and form-factors might appear—wireless tablets, wrist-watches with network connections, audio input and output devices, novel displays, etc. Our goal is to create an open network model for wide area wireless access, and this requires us to examine both technical and economic issues.

One possible strategy is to try to create an “open WAN market” in which any device could contain an easily certified, license-free wide area radio interface. However, our approach to facilitating this market is not to put a wide area radio into each device. Doing so has a number of undesirable implications. A wide area radio is bulky, produces radio transmissions that may cause health concerns, and requires substantial batteries to support the transmission. Further, in our desired competitive service environment, there may be several wide area communications standards in use in the same region, and our user may also have access to higher-performance microcells or picocells at some times. In this world each fully functional device would need several radios, not just one.

We suggest that it is much better for the user to carry only one wide area device, which we call the *personal router*, and for all his other devices to use this one unit as a relay to reach the rest of the Internet. In this way, the other devices only need connect to the personal router, through either a short wire, or a very short range wireless link. These paths require much lower power levels, smaller batteries, more flexible form factors, and so on. The personal router need not have any user interface elements at all (e.g. it need not look like a cell phone, a PDA, or a laptop). It can be carried

¹ This discussion ignores some wireless infrastructures, such as the pager networks, which permit the creation of new devices that connect to the service. But the active co-operation of the service operator is still required.

in the purse, briefcase, on the belt, etc, and perhaps can be a little more bulky than if it has to be held up to the head. It provides a point of interconnection between a range of very local interconnection methods to other consumer devices on the one hand, and to a range of wide area wireless services on the other hand. Further, this device represents the point of negotiation between the consumer and the set of wireless services with which he chooses to interact – and is thus the natural place to implement the selection framework described above. The device’s place in the architecture of network components is exactly that of a router, and thus the name.

Given this device, other devices and the applications that run on them can be designed without knowledge of which wireless service is being used. This independence mirrors the basic modular design of the Internet—applications do not know what exact network technology and what provider gives the network access. They only need to know that they are sending Internet packets, and a rough idea of the level of service they can expect. The second central objective of our research is to demonstrate that the personal router, together with its dynamic service selection framework, provides effective, powerful support for a new generation of mobile devices and applications.

2: Components of the research

The research program required to meet the objectives described in Section 1 contains several components. We outline the research space briefly here, and then present further discussion of key components in Sections 3 and 4.

Successful completion of the project also depends on the use of a number of techniques and results developed by others. This section points out crucial places where a technology or capability is required, but is not itself part of our proposed research. Much of the work supporting these results is discussed in Section 7.

2.1: Software and systems

A central question of our research concerns the issue of how a user (or a service provider) can express high level preferences about service and pricing and then have software make reasonable

low-level decisions guided by these preferences. The proliferation of “Wizards” in popular software attest to the desirability of letting users express their needs in an abstract way, and letting the system propose specific approaches to meeting those needs. This sort of translation between high level rules and low level outcomes is often described using the term *intelligent agent*.

The vocabulary of “intelligent agent” is one way to capture the idea that software will translate between high level requirements or preferences and low level implementation decisions in a sophisticated way. This idea is sometimes stated in a different way—the need for system self-organization guided by high level policy or administrative constraints. The Internet has automatic routing algorithms (which are sophisticated, if not intelligent, which are distributed, but which have never been burdened with the term “agent”); however, much manual effort is required to control routing today because of the organizational constraints that network operators wish to impose on it. This level of manual control is unworkable in a more broadly consumer-oriented or more dynamic environment. What is needed is a way to express high level constraints, and then let the automatic algorithms do what they are designed to do, subject to these constraints.

We believe that automated selection of network services given a set of service levels, application requirements and price structure is a problem that is important in its own right, but that will also yield insights about the general problem of automated network management. Automated or agent-based network management is an approach that will have increasing application as networks becomes more complex, and the user base broadens to include more people with less technical sophistication.

A discussion of our ideas and concepts for the personal router’s service selection system, which form a principal focus of this research, is given in Section 3.

2.2: Economics and business models

As the Internet has matured, there has been a great deal of discussion about suitable pricing models [McKnight, Shenker96] and some

experimentation with pricing alternatives². Wireless access potentially provides a context for experimentation that is useful in a number of respects, but that potential has not been met to date. There are two main reasons for this, both of which are addressed by our project.

First, the current industry structure has grown up in a very constrained way, based on the assumption that the only way to get substantial wide-area coverage is top-down integrated deployment over a large scale by a major provider. The technology has not permitted an experiment to see if another model might work—a bottom up grass-roots deployment of wireless by small business.

Second, since bandwidth in wireless access is actually scarce (as opposed, for example, to fiber capacity), there is a real incentive to find service models that match user needs, to price wireless in an efficient manner, and to explore user preference by real experiment. Wireless services today, which are almost always based on long-term subscription, do not explore the full space of service and pricing options for a scarce resource.

The technical work proposed here will enable a wireless service environment that is substantially richer and more dynamic than those that exist today, while still being resource-limited in many circumstances. This environment provides both opportunity and motivation to explore new economic models for service provision. In particular, both the technical and procedural limitations that narrow the range of wireless service pricing models will be eliminated. This will allow providers and consumers of wireless services, as well as third-party brokers and intermediaries, to experiment with alternative pricing models. While we do not make research on alternative pricing models a core part of this NSF proposal, we intend to use the technology developed by this project to support this type of research in the context of our Internet and Telephony Convergence Consortium. The ITCC³ is a research consortium assembled to conduct

² For example, see the Web page for the Index project at Berkeley at <http://www.INDEX.Berkeley.EDU/public/index.phtml>

³ See <http://itel.mit.edu/>

cross-disciplinary research on technical, economic and business aspects of the emerging networking industry, and is ideally suited to carry out this related research in an effective manner.

2.3: Hardware and software modularity for mobile applications

While it is possible to explore issues in service and price specification and automated selection using wireless network cards in laptop PCs, this will not demonstrate the full potential of open interfaces to create a new market. We argue that the future portable user “device” is not a PC or any other single object, but a wide range of “post PC” devices, PDA’s and palm-tops, writing tablets, note-takers, heads-up displays, scanners, digital jewelry, health and medical monitors, cameras, and other devices not yet conceived. How are all of these to be attached to the global network? Our view is that the user should carry a single device, which we call a *personal router*, to act as his agent in connecting all his other devices. We propose to prototype and deploy a device that implements this concept, and invite our students (and others) to acquire or invent other user devices that can connect to it, as a way to validate this vision and drive the development of new standards, and thus a new market.

Our broad technical goals are to explore issues in device and application modularity, how applications can organize themselves to take advantage of the resources they discover as they are activated, and how applications can negotiate for network service, and respond to changes in this service over time. The work proposed here focuses on the personal router as a) the basic connectivity path from the user’s personal environment to the Internet, and b) the focus of a service selection framework that will support the dynamic service acquisition and new market models described above.

Our design for the personal router includes significant processing and memory capacity, as well as wide area wireless access, so parts of application code, as well as infrastructure functions such as packet forwarding, routing, QoS support and service selection, may run on the router itself. This approach raises classic operating system questions for the router, such as resource allocation and security enforcement. More interestingly, the router will need to form

the hub of a distributed “personal area operating system” that encompasses other personal devices too lightweight to support full operating system functions themselves. Hence the router will take a role in supporting these core resource management and control functions for the user’s entire constellation of devices. Solutions specialized for this context must be developed and implemented.

2.4: Protocol design and implementation modularity

Our goal for this research is a set of software elements, user interfaces and protocols that permit a user to configure and control his personal router, and allow applications to take advantage of the network service provided by the router without having to have a detailed knowledge of the underlying technology. This technology independence is a hallmark, and strength, of the IP protocols. However, we believe that one aspect of the basic Internet model will be challenged by our approach. In the traditional model, there is no way for the application to determine what level of service is available, except by testing the network and evaluating the service received at the moment. Similarly, there is no way for the lower levels of the protocol stack to determine what the application is trying to do (for example, whether the file transfer that is now starting is a short one or a very long one). This modularity has proven to be extremely valuable to IP’s mission of being a technology-neutral bearer service; allowing IP and IP applications to “run over anything”. However, the same modularity introduces significant complexities for our task. In our system, there will have to be some “leakage” of information between the layers. The research challenge is to devise an architecture that supports this translucent modularity without unnecessarily degrading the technology independence that is the strength of IP networking.

Wireless throughput in our system can rapidly vary by many orders of magnitude from the 10 kb/s or less of CDPD to the 11 mb/s of IEEE 802.11 as new service access is negotiated and network conditions change. This variation is both faster and of wider range than can be gracefully handled within a static application structure and simple congestion control. It is likely that many applications will want to tailor their user-visible behavior to the network service available and the

level of assurance that that service will remain available for some period of time. For example, a browser may want to retrieve images only if the connection is a fast one, or a user might wish to start a file transfer only if she knows that it will complete within a certain interval. What is needed is a higher level of coupling between the currently negotiated level of network service and the application behavior. It is important to note that the level of service and the level of expectation that this service will continue to be available are points that are negotiated on behalf of the user; creating a feedback loop between the application and the underlying network system.

The need for tightened coupling between application behavior and network-level QoS has been recognized by a number of researchers, and is provided by many QoS-aware middleware efforts. We anticipate that the ideas and perhaps the mechanisms of these works can be applied in our environment.

In the other direction, the user may (directly or indirectly) put in place selection rules that specify that he is willing to pay to get a faster service only when he is doing certain tasks. This requires that the service selection agent know what the high-level application code is actually doing. Examples of such actions might be fetching an image, which might trigger negotiation of a faster channel, and reading confidential email, which might trigger negotiation of a secured VPN service rather than a general IP service. Two basic approaches may be taken.

First, the application may explicitly inform the system that it is taking an action that should trigger renegotiation of the service. A second approach, which is more suitable for packaged or general-purpose applications, depends on the supporting operating system to observe the application's behavior from "outside" and trigger renegotiation or reconfiguration on behalf of the application when required. Formulating a clean and simple way for limited information to pass across the protocol layer boundaries for this purpose without unmanageable complexity is a central goal of our research.

2.5: Integration

We believe that the problems described above must be explored as part of an integrated program of research to provide a convincing demonstration

of our vision. Both the service selection capability and the open interfaces for new devices are a central part of creating the context for a new market. We will obtain valuable results from each part of the research, but it is the sum of the parts that captures our vision of a new form of wireless access.

3: Negotiating for network services

Our model is that users will not tolerate an approach to service selection between a user and service provider that requires much manual management and control. The user may pass from provider to provider rapidly – in seconds or minutes, not hours, and may also wish for the system to select a service level based on changes of application or even the changing behavior of a single application. Each interaction may be of very small economic value - pennies or less, not dollars. For our vision to be practical and usable, the negotiation between user and provider needs to be automated. The user must conceptually set up a set of rules that describe his willingness to pay as a function of the actions he is currently undertaking, each provider must offer a set of services, and a software agent must select and activate the service that represents the best match.

In order to automate the process, there are a number of things that have to be characterized—the service, the pricing model, and so on. Further, one key input – the user's preferences or "set of rules" – must be expressed in a simple, intuitive, high-level manner to be accessible to most users. Our success in this research will depend on our ability to capture and produce reasonable representations of these preferences while meeting the challenges of simplicity and usability.

3.1: Overview

The essence of the personal router's service selection system is as follows:

1. A personal router desiring to acquire network connectivity on behalf of its user monitors its available networks for service offerings. Service providers wishing to offer service to the router make available *service/pricing profiles*, which describe the offered service, the provider, and the price. These profiles are likely to be low-level and detailed, but may take several forms, as described in Section

- 3.2. Determining appropriate and useful profile forms to support service selection is one part of this research.
2. At the personal router, the service offerings are received by a *service selection agent*.
 3. The service selection agent also receives information about the identity and current actions of the running application, either from the application itself or from an *application observer* used to support applications that are not designed to communicate directly with the agent. Determining an appropriate system modularity and API for applications that would benefit from communicating directly with the service selection agent, and developing algorithms that enable application observers to act in place of applications that cannot or choose not to interact directly with the agent, is one part of this research.
 4. The service selection agent also receives input from the user's control interface. This input is most likely to be high-level, perhaps single-dimensional, and virtually certain not to be expressed in terms of traditional network performance parameters such as throughput. One example of a possible user interface is discussed in Section 3.3. We note in passing that any such interface must meet the design principle of "no surprise"; that is, the interface must present a view of the system's behavior that allows the user to form intuitive mental models about its operation. Determining appropriate methods, interfaces, coordinate spaces, and modalities for both naïve and sophisticated users to express preferences or rules that control the service selection process is one part of this research.
 5. The agent must then, based on input about available service/pricing profiles, the identity and current actions of the running application, and the user's preferences and rules, select a particular service provider and profile. The level of difficulty inherent in this task depends on the level of mismatch (ie, the degree of change in coordinate space required) between the user's preferences or ruleset and the semantics of the service/pricing profiles. When user preferences and service/pricing profiles are in

the same coordinate space, the selection process is simple. When the coordinate spaces differ greatly (for example, the service profiles are in terms of "throughput" and "latency" and the user preferences are in terms of "good" and "excellent"), then the mapping function is complex, and most likely not algorithmic. In the most extreme circumstance, it is necessary to derive a mapping empirically, by watching the user's control behavior over time as he responds to application performance.

A brief discussion of approaches to describe service/pricing models is in the next section. The development of algorithms and strategies to compute or derive a particular user's mapping between high-level preferences and low-level service pricing profiles is a key part of this research.

Given a mapping algorithm, the selection agent then positions each offered profile in a $N+1$ dimensional coordinate space, where there are N dimensions of user preference and the remaining dimension is price (note that preferences for different pricing schemes, such as a desire for flat-rate, are themselves user preferences, and constitute an additional dimension). The set of profiles (or in some cases operating points within a single profile) that define the largest surface in this space are accepted; the remaining profiles are rejected because they offer less value to the user than the accepted ones. Of these, the profile that is closest to the user's price-performance tradeoff is selected, and a "contract" or arrangement is made with that profile's service provider. Note that the duration of this "contract" may be anywhere from seconds or minutes to a much longer period, and changes in application, application behavior or network environment may trigger reselection. Different contract timescales, and their impact on the complexity of the required profiles, are discussed in Section 3.4.

6. Finally, with the service selected, IP packets are routed to and from that service provider. Note that we depend on two lower-level mechanisms to enable this capability; mobile IP or its research successors to deliver packets as the personal router moves between different provider networks, and support for

long-lived transport associations that survive transition between service providers. These capabilities have significant implications for performance and security, and require further research, but such research is not the main emphasis of this proposal.

7. With service established, a payment scheme may be activated. Appropriate payment schemes take many forms, and the personal router system must support a broad range. For example, the user and the service may have a long-term contract, or the user may have agreed to pay a small incremental rate per minute to the provider, or the user may have purchased a long-term service contract with a third party that is in turn responsible for paying the user's selected provider on an incremental basis. Developing a framework for managing the information needed to implement this broad range of economic and business models for competitive service acquisition is one part of this research.⁴

3.2: How to characterize pricing/service models

Models for pricing are actually harder to construct than models for network usage, because pricing can be done in rather arbitrary ways that do not necessarily match real costs or real usage. One only has to look at pricing for telephone service (flat rate vs. metered, Friends and Family vs. 800 calling, and so on) to recognize that options for pricing are limited only by marketing creativity, and any pre-set parameterization of this is likely to constrain the options that can be offered.

We identify two possible approaches for characterizing service/pricing options offered by providers, which we call *performance-labeled* and *application-labeled*. Performance-labeled services are characterized in terms of technical performance parameters such as throughput. Committed sending rates, token buckets and other usage profiles are examples of performance labels. In contrast, application-labeled options are

⁴ However, supporting activities such as the design of a new micropayment protocol or new form of electronic cash are not, unless it turns out that some presently unrecognized aspect of the research problem requires us to do so.

described in terms of what application they are suited for, and a relative ordering for that application. This form of label is closer to what a marketing department might devise: "Web turbo service".

Both of these representations require additional information to specify the pricing, which may be a function of the duration of the contract. (One can imagine buying service for a period measured in second, minutes, or even hours or months in certain cases. Section 3.4 discusses the relationship between timescale of the service commitment and the complexity of the service description required.)

Each of these schemes has advantages and disadvantages.

Performance-labeled services: These specifications describe the offered service in terms of performance parameters, such as throughput or burst tolerance. They require the user's selection agent to find a mapping from a set of performance parameters to an overall desirability for a specific application. A service that works well for streaming MP3 playback is not necessarily best for using the Web. So a set of *application-specific rules* are required to describe the sort of services that work well for an application. These rules can be supplied by the application designer, by a third party, or by a sophisticated user. A more ambitious option would be that the agent could observe actual application usage patterns, and propose and try rules automatically.

The limitation of the performance-labeled approach is that some framework for service description must be developed and adopted, and only services that can be described in this framework can be marketed. So of necessity it is impossible to create and market arbitrary performance-labeled services. Our prior research into Expected Capacity Profiles [Clark98] has well positioned us to develop models of bursty usage, and to link these to pricing options. However, additional research may yield more general forms of performance labels; for example we are currently evaluating a new service specification called Bounded On-period Peak Rate filtering.

Application-labeled services: These specifications are not based on technical performance parameters. Instead, they identify an application,

and some ranking of service merit for that application. Here the agent can attempt to improve the service by moving to a better grade of service, but cannot evaluate what the service is. This is actually easier to implement, since it gives the selection agent less to do. However, it creates the problem that it is impossible to compare services from competing providers, and it is hard to know what to do with a new application for which there is no named service.

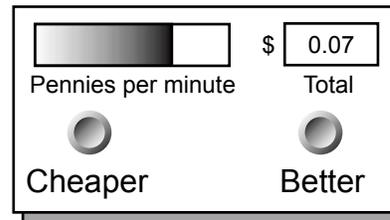
Our research will explore the limits of both options, which define the endpoints of a spectrum. It is likely that a hybrid solution is the correct one, in which services are described by technical performance specifications, and in addition may carry hints about which applications they are considered useful for. Additionally, there might be attached text descriptions of the performance issues that do not fit within our specification framework.

3.3: Creating a user interface to service selection

We assume that the selection agent makes service choices based on high-level user guidance and the specifics of what services are offered, what applications are running, and so on. We assume that most users, unless they are unusually sophisticated, will want only a high-level control over this process. Therefore some user interface will be required. The user will want some way to control the overall behavior of the system, and to monitor any current rate of charging.

As a starting point, we propose a simple user interface that attempts to map user control onto a single dimension of service quality, which we call the “better-cheaper” spectrum. In this model, the user interface consists of two buttons (“better” and “cheaper”), a meter that shows the money consumption since last reset (the “taxi meter”) and a meter that shows the derivative of this meter (the “money speedometer”). In this model, if the user is running an application and is not satisfied with the current performance, he clicks the “better” button, and his agent selects a new service from among the service options provided by the various providers. “Cheaper” has the

opposite effect. A conceptual graphic of this interface is shown below.⁵



It is interesting to note that key to making this interface intuitive is that each push of the button cause a noticeable, and roughly similar, change in the relevant metric. For example, each push of the “Better” button might roughly double application performance, while each push of the “Cheaper” button might cut the cost in half. An implication of this is that the scale going up is not the same as the scale going down. In one case the quality doubles, in the other the price halves, but there is not necessarily a one-to-one correspondence between the two.

It is possible to imagine a multi-dimensional control panel, and such a thing may prove necessary as we gain experience with the system. But our goal is to put as much intelligence as possible into the selection agent, and thus we take as a starting point a simple, single-dimension control at the user level, and a complex, application-specific transformation within the selection agent.

An important dimension of this selection process is that the application can adapt to a selected service, as well as drive the selection. So the *application-specific rules* mentioned above can provide instructions for the application as well as

⁵ One additional component of this proposed interface that is not shown here is the “flat rate light”. This indicator is solidly lit when the current service has no *incremental* cost, perhaps over that of a long-term contract in place between the user and current service provider. The indicator is flashing when the present service is incrementally charged, but pushing one or the other button would eventually arrive at a flat rate service. The indicator is dark when no flat rate service is available. We believe that this simple mechanism will address people’s frequent desire to use a flat-rate service when available, even if it is not necessarily economically efficient.

the selection agent. For example, one could write a rule that says that for throughputs less than a certain amount, a Web browser should retrieve only text, and not images. Given that rule, as the user clicks the “cheaper” button, the application behavior may change, as well as the perceived performance.

3.4: The time scale of contracts

Most Internet services today are priced based on long-term contracts—monthly fees, yearly commitments, etc. It has been widely observed that these contracts seem to be preferred in the market, since they remove pricing risk from both parties. However, in the case of a dynamic market between mobile users and very local providers, the obvious market is a short-term service contract. At the extreme, one could negotiate for service on a second by second basis, buying high capacity to download a Web image, a small capacity to check for mail, and so on. On a slightly longer time scale, one might purchase profiles with a peak and an average rate. It is our hypothesis that for contracts of a few seconds or a minute duration, that great complexity in the specification framework is not necessary or justified. It is when a contract needs to describe usage profiles over a number of time-scales at once that the specification becomes complex. So by using a combination of performance parameters and a repeating negotiation, that more complex overall service relationships can be constructed.

We believe that with suitable controls and protections, most users will be comfortable with a short-term pricing model. One of the advantages that arises from the fact that individual transactions are small is that a small error in picking the right service at any moment is not likely to trigger a significant economic loss. This permits the negotiation to be structured in a way that economists call an “experience good”. That is, the user can try a service and evaluate incrementally whether the cost and performance meet the needs. This is the purpose of the “taxi meter” in the user interface, which can be supplemented with rules that provide constraints on what the agent is allowed to select.

4: The personal router

The personal router is a key device in our vision of a service-rich environment. It sits at the boundary of two worlds; the user’s constellation of personal devices and “digital jewelry” on the one hand, and the outside world of competing network service providers on the other. To fulfill this role, the router takes on a number of functions.

- It provides basic network connectivity between the two worlds. This includes simple packet forwarding, mobile networking functions, wireless protocol functions such as energy management, and higher level functions such as firewalling, managing security associations, and the like. In essence the personal router acts as a boundary router between two different administrative, security, and policy regions of the Internet, with all of the functions that this implies.
- It implements the service selection framework described in Section 3. To do this it implements the selection protocols, communicates with service providers on behalf of its user, communicates with the user’s applications to obtain usage information and requirements, and implements or communicates with the user interface that gives the user control of the selection process.
- It implements the core of a dynamic application composition and adaptation framework. This framework supports the creation of applications that span multiple personal devices, perhaps including the router itself, and that reconfigure and respond dynamically to changes in user requirements and the user’s network environment. This application framework is an important part of the overall personal router vision, but we are not requesting support for it in this proposal, and it is possible to make significant progress on the core personal router and service selection components using a simple, skeleton application adaptation model and framework. Hence, we do not discuss it in depth here.

We emphasize that while the overall concept of the personal router is crucial to our goal of a rich, competitive environment for mobile applications, devices, and service providers, substantial

portions of the device's functionality can be realized by leveraging existing work. Our strategy is to do this wherever possible, reserving our research effort for those components of the router that are truly new. Our research and development plan for the personal router leads to a low-risk high payoff realization of a device that is realistic in form as well as function, and hence can be used to support real experiments and deployment in a wide community. This approach takes the project a substantial step beyond the alternative of simply implementing the router's software and hardware functions in an ad hoc manner on an existing platform such as a laptop PC. This is because the size, power, and interface restrictions of that type of platform mean that it can not be used to support realistic applications and usage patterns in a post-PC mobile environment.

4.1: Hardware platform

Our approach to constructing the personal router is based on leveraging our current *LCS Communicator* project [Wroclawski99]], in which we are developing a portable user device closely resembling a cell phone in form factor, but equipped with an 800x600 SVGA color micro-display, a small camera, a PC Card (PCMCIA) slot that can host a wireless LAN or cellular modem card, and a general purpose processor and memory. The Communicator software architecture integrates this device with a distributed back-end conversational language framework, providing an overall platform for delivering applications with an I/O-rich, natural language interface to the mobile user. One portion of the Communicator software implements a mobile IP environment that supports manual selection of, and transition between, different service providers.

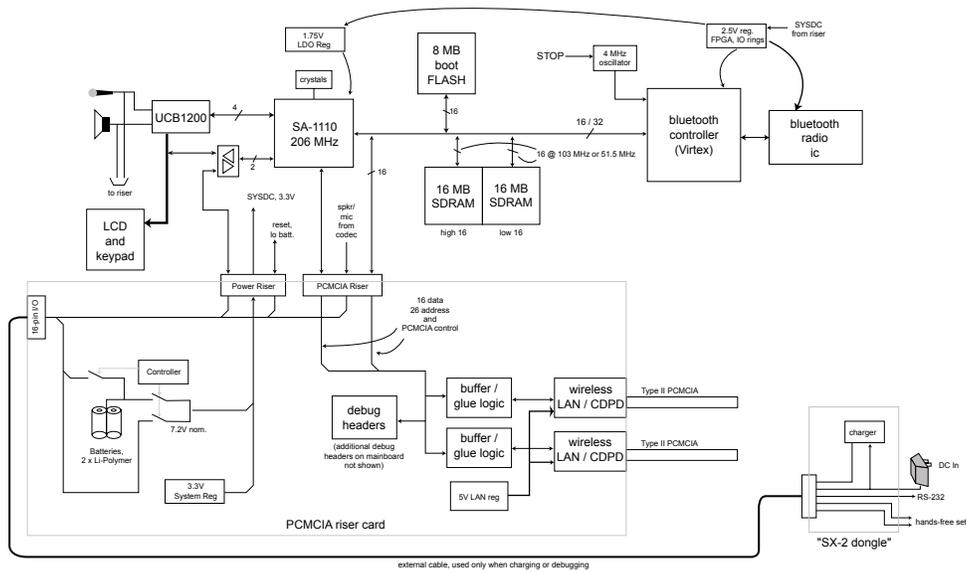
The Communicator device will be used to experiment with a variety of speech-based applications in the highly mobile wireless environment, and provides an interesting platform for further research in human factors and application design. However, the device takes an essentially traditional perspective on the matter of form factor and delivery model for post-PC applications. Our proposal for a personal router reflects our conclusion that we need to explore alternative form factors and device modularities.

Our plan for the personal router is to repack the LCS Communicator device, remove the user interface elements such as micro-display and camera, add a second PC Card slot (so that the device can interface to two forms of wireless access to the Internet at the same time, initially CDPD and 802.11 LAN), and add a mechanism for short-range access to the user's other personal devices. Our initial choice for personal area interconnection is expected to be Bluetooth radios. Bluetooth is not ideal for this use, because a Bluetooth picocell supports only 8 devices, and because the current Bluetooth protocols are not ideally suited to our model of general-purpose IP interconnection. However, the wide commercial availability of Bluetooth technology and consumer devices makes this a low-risk initial choice. At a later date, we will evaluate the use of other technologies such as the rapidly emerging Ultra-Wideband, or pulse, radios, which may provide dramatic bandwidth-power product improvements over current technology. However, this later work is outside the scope of the current proposal, and is not discussed further here.

We will base the hardware on the existing chip-level design for the Communicator, but will have to modify the schematics and netlists, produce a new circuit board, and design and fabricate new packaging for the device. Initial engineering estimates suggest that approximately 10% of the electronics design will be new. Crucially, we will be able to reuse virtually all of the power supply and management components, which are the most difficult part of a portable design.

While this work is very low risk, given our prior development of the original Communicator, it will require a reasonable investment of engineering effort. We propose to share the cost of this development project among a number of sponsors (funds from other sponsors are already in hand), and are asking NSF to provide support for only part of the design effort, as well as the cost of obtaining devices to support ongoing research once the design is proved successful.

A block diagram of the personal router hardware platform is shown below. Key elements of the device are the StrongArm processor core, the PC Card I/O subsystem, the personal area radio interface and IC, and the power supply and energy management components.



Personal Router Block Diagram, 1/16/00

4.2: Personal router software

Software required for the personal router includes the operating system, network protocols and routing functions, the service selection framework described in Section 3, a basic framework for implementing dynamically composed distributed applications across the user's personal digital devices, and, in some circumstances, software to support a simple user control interface.

The initial base operating system for the personal router will be a version of Linux derived from that developed for the LCS Communicator. Drawing on code developed for the Communicator and by others in the community, this base operating system supports a number of capabilities needed by the personal router. Additional operating system capabilities specific to the personal router will be added as required. This operating system environment will serve as the base for our implementation of the service selection framework described in this proposal, as well as continued development of the application composition framework mentioned at the start of this section.

Important software functions to be provided by the router's operating system are listed below.

- Basic operating system functions, plus remote booting, downloading and debugging of operating system code.
- Basic IP network protocol functions, including packet forwarding, routing, and upper-layer protocol stack.
- Support for mobility, including mobile IP and recent research community enhancements.
- Support for lower-layer wireless communications protocols, presently including CDPD and a host-based implementation of 802.11. This host-based 802.11 implementation allows us to modify the protocol as required to support research enhancements such as QoS management, energy management, and alternative roaming and binding algorithms.
- Support for IP-level QoS control based on IETF standards and our continuing research in this area.
- Support for IPSec and encrypted tunnels (VPN's), providing the base capability for applications to select secure communications services.

- Power monitoring and management functions for the personal router hardware.

Of these capabilities, most are presently implemented by the version of Linux used by the LCS Communicator. In the course of developing the personal router, some level of further refinement will be required in the areas of security and authentication capabilities, and in more advanced wireless lower-layer capabilities. Beyond this, we expect to incorporate continuing research results in the areas of QoS control, mobile networking protocols, and energy-efficient communication algorithms presently being developed by our group and other researchers.

Also incorporated within the personal router will be a Java VM and supporting environment. This environment will have several distinct uses. First, it will be used to implement the mechanisms that implement the personal router's service selection framework. We choose to implement the framework in the sandboxed Java environment for robustness and security, because we anticipate that our research may eventually lead to service selection agents that are dynamically created or passed to the personal router by other components of the system. Second, we anticipate the Java-based component software will be used to underpin our application synthesis framework. Because the personal router also serves as a possible execution platform for portions of a distributed application, provision of an efficient Java environment on the router is crucial. Finally, as a matter of consistency, we choose to make the software interfaces that control the router itself available through Java remote invocation mechanisms; allowing the router to use any Java-enabled device with UI capabilities to serve as the control interface between the router and its user.

4.3: Security, authentication, and privacy

The personal router project raises a number of issues related to security, authentication, and privacy, which range in complexity and difficulty. Two of particular interest are the model of authentication used by the personal router, and the tradeoff between dynamic online selection and privacy.

For a personal router to acquire services from the network it is necessary that the router be able to identify its user to at least some entity in the network, so that on-demand services can be charged to a user. Various authentication schemes are possible, corresponding to various business and payment flow models.

A key issue is that the router acts as an agent of the user, but must do so cautiously. If, for example, a stolen router can be used to acquire network service that is charged to the original owner, it is likely that stealing routers will become commonplace.

Instead, the router can derive its identity from its user. Two models are of interest. The "passive", or open model, allows a router to take on the identity of any user's personal area network and use it to acquire services for that user. This would be appropriate when personal routers are passed from user to user, or perhaps used to provide network services for a group of users. The "active", or closed model assumes that a router belongs to a specific user. In this model the router, before agreeing to acquire service for the user, the router challenges the current user to prove his identity, and refuses to provide service if the user's identity is not what is expected.

One technological solution is to provide a "signet ring" or other device as a complement to the router, and require that the router be in contact with the ring to operate. It need not be in constant contact, but can refuse to operate if it cannot renew contact every so often.

We will explore what technological solutions may be possible, and implement one or more in our personal router. For our first round of research, a possibility is to require that the user re-identify himself to the device periodically with some sort of challenge-response protocol based on a PIN and a crypto algorithm. An alternative, derived from related work in our laboratory, might be speech recognition. Further into the project, we plan to implement the signet ring approach.

A much more difficult issue is that the user of this system in its simplest form leaves a detailed trail behind him of where he has been, since he must constantly select or negotiate service contracts, and must identify himself to do so. This issue is similar to the consequence of using

a cell phone, which constantly informs the cellular system where a person is (to the granularity of a base station) so that one can receive a call. One might propose a payment scheme based on anonymous cash to avoid this consequence, but we do not propose to implement such a scheme in this round of research.

5: Specific research tasks

Automated service selection

To carry out this research, the following tasks are required:

- Development of models for service level agreements.
 - Development of models for pricing of service.
 - Development of an approach for passing usage and service requirements information among the protocol layers from application to selection protocol.
 - Development of a protocol for intelligent negotiation between provider and user for selection of service/pricing options.
 - Development of agent software for mobile computer (e.g. laptop) to implement the agent selection protocol.
 - Development of agent software for base station (based on router software running on open PC platform) to implement the agent selection protocol.
 - Development of prototype billing agent to manage payments during initial experiments.
 - Development of user interface tools for laptop to permit user to express usage preferences, and to monitor cost and service quality on an ongoing basis.
 - Development of methods and protocols that prevent theft of wireless service.
- Modification of Communicator schematics to remove U/I elements, add second PCMCIA slot, personal area radio interface.
 - Fabrication of hardware; PCB layout and fab, mechanical design, case fabrication
 - Port of Communicator system software to personal router.
 - Development or port of router code for personal router.
 - Port of agent selection code.
 - Redesign of user interface software for control of selection code, using other user device as human interface.
 - Selection or modification of other devices to interface to router, and demonstration of modular applications running on these devices.

6: Relation to long-term research goals

Our long-term goal is to move the Internet protocols from their current focus on interconnection of PCs to a post-PC world of embedded devices, portable user interface elements, consumer networking, and applications that reconfigure themselves in the context of a changing base of hardware. This long-term vision has been articulated in a position paper⁶. This project is an essential step in that overall program.

Concretely, this project builds on a number of prior projects done in the Advanced Network Architecture group at the MIT Laboratory for Computer Science.

- Our works on Quality of Service over the last 10 years [Wroclawski 97A, Wroclawski 97B, Clark98]) have equipped us to understand issues in service definition and characterization.
- Our work on Internet pricing [Clark 97] has given us experience in economic issues.

Personal Router

To carry out this research, the following tasks are required:

⁶ See <http://www.lcs.mit.edu/anniv/speakers/presentation?id=041399-6>

- Our work on Internet protocols and modularity has given us a deep understanding of how to relax modularity constraints so that applications and the selection layer can interact effectively. This interest in relaxing layer boundaries is a long-standing study. (eg.[Cooper])
- Our work on the LCS Communicator has provided a hardware platform that can be easily modified to produce the needed hardware. This same work has also given us substantial experience with emerging user interface modalities such as conversational speech, and with alternative modularities for mobile digital devices.
- Our work on an open software platform for IP routing gives us a useful software base both for the base station and for the personal router.

This project will in turn underpin substantial further research directed at our long-term objective. Upon completion of this project, we will use its results and artifacts to support the next stage of our research, which will focus on an adaptive, dynamic application catalysis framework for use with the decomposed component hardware model enabled by the personal router. We plan to integrate this decomposed hardware/application framework with our current work on system software for conversational speech-based mobile devices and applications, leading to an overall model for building applications that offer rich, human-oriented user interfaces in a world of post-PC digital devices.

7: Related work

The research proposed here draws on related and previous work from a number of areas. We summarize several of these areas and certain relevant results within each here, but of necessity this discussion is limited due to space restrictions. In particular, we acknowledge the large body of academic and commercial work on wearable computing, which we have omitted discussing here in order to focus on networking and communications issues.

7.1: Intelligent agents and negotiation

Intelligent agents are a topic of much research in the CS community today. Our process for service selection can be described as agent-based, in that it takes high level rules and high level user guidance, and makes low-level specific decisions based on these external inputs.

A number of different sorts of intelligent agents have been developed that bear to some extent on our work here. One category of agent is the negotiation agent for e-commerce. [CACM 42] provides a good overview of the state of the art in this area. The Tête-à-Tête system at MIT's Media Lab [Guttman] provides the ability for a user to specify a multi-dimensional preference space (not just price but issues such as privacy, shipping time and so on), and negotiate automatically within these constraints. We believe that schemes such as these provide evidence that automated negotiation within the restricted domain of network service pricing is practical.

There are a number of rather different agent systems that have been called “negotiation agent” systems. One class of agent is a support tool for human to human negotiation, such as the *Inspire* system (<http://www.business.carleton.ca/inspire/>). This work does not directly bear on our research. Another class of negotiation agent is a program that goes through a multi-step bid-offer cycle for market purchasing. For example, see [Oliver]. This work may be of some relevance, but we hypothesize that this sort of complex negotiation is not required for our system, which is more *selection* than *negotiation*. Yet a third form of negotiation agent is represented by the Pleiades project at CMU (<http://www.cs.cmu.edu/afs/cs.cmu.edu/project/heo-5/www/pleiades.html>). This system provides personal agents that observe a user's behavior and customize themselves to the user's needs. These agents negotiate among themselves to improve their effectiveness. This sort of approach might be helpful in the part of our system that observes usage patterns of specific applications, and then attempts to find service offerings that match the user's high-level guidance from the “better/cheaper” interface. This form of negotiation, of course, has nothing to do with the negotiation between buyer and

seller in a market. Finally, there are the so-called “search agents” that look for low prices on network products, such as the C/Net web site. (<http://www.cnet.com>). These commercial services may perform some search function, but they do not appear to relate directly to our requirement to search and select among available service offerings.

7.2: QoS-aware middleware

A number of recent research projects have addressed the broad topic of providing applications with some awareness of, and ability to control or adapt to, network level quality of service constraints. Those systems that assist the application to adjust to changing network service levels are particularly relevant to our work, because they provide frameworks that could be used to allow an application to dynamically adapt to the changing service levels that it might obtain from the network within our architecture.

The Dynamic Integrated Resource Management (DIRM) project at BBN is an example of a middleware project designed to allow an application to control the QoS of the underlying network. This system allows the application to request a specific QoS using a reservation protocol. It requires that applications take explicit account of the QoS features; it does not support the implicit augmentation of existing applications using external rules. The Adaptive Quality of Service Availability project, also at BBN, which builds on DIRM, provides a means for the application to adapt itself to changing network service, with specific attention to changing patterns of replication driven by network availability. The perspective of this work is distributed object systems, and method invocation in a variable QoS world. Again, the application must be implemented to take advantage of these tools. These projects build on their Quality Object (QuO) architecture, which extends the CORBA distributed object framework by providing a means for the application to discover, at the time of method invocation, what category of service is in force, and to adapt its behavior accordingly. These systems concern themselves both with classic performance metrics of bandwidth and delay, and more general issues of availability which are very important to our presumed system. See

[Vanegas], [Zinkey], or <http://www.dist-systems.bbn.com/tech/QuO/>.

The Darwin project at CMU is a system that provides an application a high-level, application-wide view of the network resources that have been allocated to the application, and a set of network-level resources that permit integrated resource management. This capability implies the ability to specify resource requirements at a high level. See [Chandra] or <http://www.cs.cmu.edu/~darwin/>. The Remulac project at CMU is concerned with an API and supporting tools to gather and provide to the application information on current levels of resource availability. See [Lowekamp], or <http://www.cs.cmu.edu/~cmcl/remulac/index.html>.

The Scope project at USC ISI is concerned with the issues of resource allocation with resource discovery, resource selection based on assurance credentials, distributed authorization, and online payments. It explicitly addresses the issue of purchase of resources on a short-time basis to supplement long-term resource contracts. See <http://www.isi.edu/gost/projects/scope/>.

The Agilos project at the University of Illinois is developing a framework by which a adaptive QoS middleware layer can control an agile application as the service quality of the network changes. The goal is that the application will adjust to give the best possible service under the conditions at hand. See [Li] or <http://cairo.cs.uiuc.edu/tracking/index.html>.

7.3: QoS for wireless networks

Wireless QoS: Providing wireless quality of service has been addressed in a number of projects that capture application adaptability to changing network conditions to differing degrees. For instance the frameworks in [Gerla, Lee, Sinha] provide a very limited form of QoS support (actually only one or two bandwidth levels) in an ad hoc wireless network. This work is useful in that it has brought to light many of the issues surrounding QoS in highly dynamic networks. Each project uses different mechanisms for how the QoS is signalled to the network, employing both in-band (ip-options) and out-of-band techniques (reservation requests).

The QoS problem has been further considered in [Singh, Bianchi]. The Mobiware project attempts to capture an adaptive QoS model for applications through both an application utility function and an adaptation policy. The utility function captures how an application's performance quality scales against one performance parameter (usually bandwidth). The adaptation policy captures in a limited fashion (fast, slow, or during hand-off) the time scales on which an application can adapt to changing conditions. Other QoS parameters are suggested in [4] such as probability of seamless service and a loss profile that captures the preferred way in which data should be discarded in the event of congestion.

In [Toh, Campell, Karol, Indu, Choi, Raychaudhuri, Yuan] the problem of QoS is addressed in the context of wireless ATM. The performance of these approaches depends largely on the various medium access control (MAC) protocols proposed. In [Chen] the comparative performance many of these access protocols is investigated. The hand-off techniques and MAC protocols are particularly relevant pieces of work.

Mobile IP: An extensive amount of work on Mobile IP has been done by the IP Routing for Wireless/Mobile Hosts Working Group of the IETF [MobileIP]. The group has developed routing support to permit IP nodes to seamlessly move between IP subnetworks. The Mobile IP method provides transparency to active TCP connections and UDP port bindings. Future planned work of the group dealing with micro mobility (movement within a subnet), QoS using diff-serv and/or int-serv/RSVP, and location privacy will be of particular interest.

Extensions to the Mobile IP framework have been proposed such as Cellular IP [Valko] that may improve performance in certain scenarios. It has been demonstrated in [Caceres] that Mobile IP with fast moving hosts is not an optimal solution when hand-off latency increases considerably, as the distance between home agent and mobile agent increases. The Cellular IP extensions support local mobility (mobility inside an access network) by creating a hierarchy in which the majority of the hand-offs occur locally. Other projects [Fladenmuller, Williamson] are examining performance

characteristics and optimizations for Mobile IP. We expect to leverage their results as well.

Other related mobile computing projects include the Mobile People Architecture (MPA) [Roussopoulos], the Iceberg project [Joseph], and the TOPS [Anerousis] architecture. All three of these projects attempt to provide user level mobility within one or more network types. The MPA relies upon a person-level router, the Personal Proxy, that tracks a mobile person's location, accepts communication on the user's behalf, performs any conversions, and then forwards communications to the user. The Iceberg project and the TOPS architecture rely upon similar tracking proxy nodes within the network.

7.4: Power aware computing

Because a mobile host relies on a limited battery as a source of power, power-conservation is an important factor in mobile system design. Scientists are therefore currently examining power-conservation techniques that address every aspect of a mobile host's operation, including wireless communication. In the area of wireless communication, this research has focused on two topics, using communication to migrate computation away from the mobile host and redesigning standard protocols to use less power. Certain mobile host applications, for instance sensor-data analysis or natural language recognition, are compute-intensive and therefore will use a significant amount of a mobile host's power supply. Some scientists have recently proposed that a distributed computation model may help to offset this problem, where mobile hosts would negotiate to offload computations onto other systems with larger battery supplies, thus trading off computation costs for communication costs [Rabiner]. (In our case, an interesting speculation is that the personal router becomes the "large" host, and the computation is offloaded from smaller, lower-power personal devices. A more significant amount of research has gone into reducing the high cost, in terms of power, of actually communicating over a wireless medium. Over the last two years, scientists have begun to seriously examine the power consumption of network protocols at all levels of the protocol stack, including the physical, MAC, IP, TCP, and application layers

[Kravets99, Kravets98, Stemm, Sivali, Chlamtac, Rabiner, Zorzi, Chock]. Of particular relevance to this work is the research that has focused specifically on power-conservation for quality-of-service support [Lettier, Goodman] and for personal-area networks [Gomez]. In general, we believe that we will need to build upon all of these power-conservation techniques in designing the personal router.

8: Citations

- [Anerousis] N. Anerousis, R. Gopalakrishnan, C.R. Kalmanek, A.E. Kaplan, W.T. Marshall, P.P. Mishra, P.Z. Onufryk, K.K. Ramakrishnan, and C.J. Sreenan. "TOPS: An Architecture for Telephony Over Packet Networks," *IEEE Journal of Selected Areas in Communications*, 17(1), January 1999.
- [Barford] P. Barford and M. Crovella. "Generating Representative Web Workloads for Network and Server Performance Evaluation," *SIGMETRICS '98*.
- [Bianchi] G. Bianchi, A. Campell, R. R.-F. Liao. "Programming QoS in Mobile Multimedia Networks"
- [Braden] Braden, R., Clark, D. and S. Shenker, "Integrated Services in the Internet Architecture: An Overview", Internet Request for Comments RFC 1633, July 1994.
- [CACM 99] Communications of the ACM. Vol 42(3), March 1999. Special issue on Multi-agent Systems on the Net and Agents in E-commerce.
- [Caceres] Ramon Caceres, V. N. Padmanabhan, "Fast and Scalable Handoffs for Wireless Internetworks," *Proc. ACM Conference on Mobile Computing and Networking (Mobicom '96)*, pp. 56-66, 1996.
- [Campell] A. Campell, R. Liao, Y. Shobatake, "QoS Controlled Handoff in Wireless ATM Networks". *Proc 6th WINLAB Workshop on 3rd Generation Wireless Information Networks*, New Jersey, USA, March 20-21, 1997.
- Chandra] Prashant Chandra, Allan Fisher, Corey Kosak, T. S. Eugene Ng, Peter Steenkiste, Eduardo Takahashi, Hui Zhang, Darwin: Resource Management for Value-Added Customizable Network Service, Sixth IEEE International Conference on Network Protocols (ICNP'98), Austin, October 1998
- [Chen] Jyh-Cheng Chen, Krishna M. Sivalingam and Raj Acharya. "Comparative analysis of wireless ATM channel access protocols supporting multimedia traffic" *Mobile Networks and Applications* Volume 3 , Issue 3 (1998).
- [Chlamtak] Imrich Chlamtac, Chiara Petrioli, and Jason Redi. Energy conservation in access protocols for mobile computing and communication. *Microprocessors and Microsystems Journal*, 1998.
- [Chock] A. Chockalingam and M. Zorzi. Energy efficiency of media access protocols for mobile data networks. *IEEE Transactions on Communications*, 46, November 1998.
- [Choi] Sunghyun Choi and Kang G. Shin. "A cellular wireless local area network with QoS guarantees for heterogeneous traffic" *Mobile Networks and Applications* Volume 3 , Issue 1 pp. 89-100 (1998).
- [Clark 97] D. Clark. "Internet Cost Allocation and Pricing". L. McKnight

- and L. Bailey ed. *Internet Economics*. MIT Press, 1997
- [Clark 98] D. Clark and W. Fang, "Explicit Allocation of Best Effort Packet Delivery Service", *IEEE/ACM Trans. on Networking*, vol. 6(4), August 1998, pp. 362-373.
- [Conti] Marco Conti, Claudio Demaria, and Lorenzo Donatiello, "Design and performance evaluation of a MAC protocol for wireless local area networks" *Mobile Networks and Applications* Volume 2, Issue 1 (1997)
- [Cooper] Cooper, G., *An Argument for Soft Layering of Protocols*, 1983. MIT-LCS-TR-300, August 1983
- [Edell] Richard Edell, N. McKeown and P. Varaiya, "Billing Users and Pricing for TCP," *IEEE Journal on Selected Areas in Communications*, Sep. 1995.
- [Fladenmuller] Anne Fladenmuller and Ranil De Silva, "The effect of mobile IP handoffs on the performance of TCP" *Mob. Netw. Appl.* 4, 2 May. 1999.
- [Gerla] M. Gerla, J T-C Tsai. "Multicluster, mobile, multimedia radio network," *Wireless Networks* 1 1995, pp. 255-265.
- [Gomez] Javier Gomez, Andrew T. Campbell, Mahmoud Naghshineh, and C. Bisdikian. Power-aware routing in wireless networks. In *Sixth IEEE International Workshop on Mobile Multimedia Communications (MOMUC)*, November 1999.
- [Goodman] David Goodman and Narayan Mandayam. Power control for wireless data. In *The Sixth IEEE International Workshop on Mobile Multimedia Communications (MOMUC)*, November 1999.
- [Guttman] R. Guttman and P. Maes. "Agent-mediated integrative negotiation for retail electronic commerce". In *Proceedings of the workshop on Agent-Mediated Electronic Trading AMET'98*, Minneapolis, May 1998
- [Indu] Indu Mahadevan, and Krishna M.Sivalingam. "An architecture for QoS guarantees and routing in wireless/mobile networks" *Proceedings of first ACM international workshop on Wireless mobile multimedia*. October 25 - 30, 1998, pp. 11-20 Dallas, TX USA
- [Johnson] D. Johnson. "Scalable Support for Transparent Mobile Host Internetworking" *Mobile Computing*, 1996.
- [Joseph] A. D. Joseph, B. R. Badrinath, and R. H. Katz. "The Case for Services over Cascaded Networks," In *Proceedings of WOMMOM '98*, October 1998.
- [Karol] M. J. Karol, Z. Liu, K. Y. Eng, "An Efficient Demand-Assignment Multiple Access Protocol for Wireless Packet (ATM) Networks", *ACM Journal on Wireless Networking*, December 1997.
- [Kravets98] Robin Kravets and P. Krishnan. Power management techniques for mobile communication. In *The Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, October 1998.
- [Kravets99] Robin Kravets, Karsten Schwan, and Ken Calvert. Power-aware communication for mobile hosts. In *The Sixth IEEE International Workshop on*

- Mobile Multimedia Communications (MOMUC), November 1999.
- [Lee] S-B Lee, A Campbell. ``INSIGNIA: In-band signaling support for QoS in mobile ad hoc networks," *Proceedings of 5th Intl. Workshop on Mobile Multimedia Communication* 1998.
- [Lettieri] Paul Lettieri and Mani B. Srivastava. A QoS-aware energy-efficient wireless node architecture. In The Sixth IEEE International Workshop on Mobile Multimedia Communications (MOMUC), November 1999.
- [Li] Baochun Li, Klara Nahrstedt, A Control-based Middleware Framework for Quality of Service Adaptations, in IEEE Journal of Selected Areas in Communications, Special Issue on Service Enabling Platforms, Vol. 17, No. 9, pp. 1632-1650, September 1999.
- [Lowe] Lowekamp, B. Miller, N. Sutherland, D. Gross, T. Steenkiste, P. Subhlok, J., "A Resource Query Interface for Network-Aware Applications" , 7th IEEE Symposium on High-Performance Distributed Computing, IEEE, July 1998, Chicago,
- [McKnight and Bailey] L. McKnight and J. Bailey (Eds.), *Internet Economics*, MIT Press, 1997
- [Maes] P. Maes, R. Guttman and A. Moukas, Agents that Buy and Sell, *CACM* Vol 42(3), March 1999, pp 81-91
- [Miller] B. Miller, N. Sutherland, D. Gross, T. Steenkiste, P. Subhlok, J., "A Resource Query Interface for Network-Aware Applications" , 7th IEEE Symposium on High-Performance Distributed Computing, IEEE, July 1998, Chicago
- [MobileIP] IP Routing for Wireless/Mobile Hosts (mobileip) Charter: <http://www.ietf.org/html.charters/mobileip-charter.html>
- [Odlyzko] Andrew Odlyzko, "The Internet and other networks: Utilization rates and their implications," in Proc. of 26th Telecommunications Policy Research Conference, Oct. 1998.
- [Oliver] J.R. Oliver, On Automated Negotiation and Electronic Commerce, PhD, University of Pennsylvania. <http://opim.wharton.upenn.edu/~oliver27/html/diss.overview.html>
- [Rabiner] Wendi Rabiner, Joanna Kulik, and Hari Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. In The Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM), August 1999.
- [Raychaudhuri] D. Raychaudhuri. ``Wireless ATM: an enabling technology for multimedia personal communication" *Wireless Networks* Volume 2, Issue 3 (1996)
- [Roussopoulos] Mema Roussopoulos, Petros Maniatis, Edward Swierk, Kevin Lai, Guido Appenzeller, Mary Baker, "Person-Level Routing in the Mobile People Architecture." *Proceedings of the USENIX Symposium on Internet Technologies and Systems*, October 1999.
- [Shenker 95] Scott Shenker, "Service Models and Pricing Policies for an Integrated Services Internet," in Public

- Access to the Internet, B. Kahin and J. Keller, ed., pp. 315--337, 1995.
- [Shenker 96] S. Shenker, D. D. Clark, D. Estrin, and S. Herzog, "Pricing in Computer Networks: Reshaping the Research Agenda," *ACM Computer Communication Review*, vol. 26, pp. 19-43, Apr. 1996.
- [Singh] S. Singh. "Quality of Service Guarantees in Mobile Computing." *Computer Communications*.
- [Sinha] P. Sinha R. Sivakumar V. Bharghavan. "CEDAR: a Core-Extraction Distributed Ad hoc Routing algorithm" University of Illinois at Urbana-Champaign. 1999.
- [Sivali] Krshna Sivalingam, Jyh-Chang Chen, Prathima Agrawal, and Shaline Kshore. Design and analysis of low-power access protocols for wireless and mobile ATM networks. *ACM/Baltzer Journal on Special Topics in Mobile Networking and Applications (MONET)*, June 1998.
- [Stemm] Mark Stemm and Randy Katz. Reducing power consumption of network interfaces for hand-held devices. In The Third IEEE International Workshop on Sixth IEEE International Workshop on Mobile Multimedia Communications (MOMUC), September 1996.
- [Toh] C.-K. Toh, "Mobile Quality of Service for Wireless ATM Networks: An Adaptive Approach", *Proc. of ACTS Mobile Communications Summit*, Denmark, Oct 1997.
- [Valko] A. G. Valko, "Cellular IP - A New Approach to Internet Host Mobility," *ACM Computer Communication Review*, January 1999.
- [Vanegas] Vanegas R, Zinky JA, Loyall JP, Karr DA, Schantz RE, Bakken DE. QuO's Runtime Support for Quality of Service in Distributed Objects. Proceedings of the IFIP International Conference on Distributed Systems Platforms and Open Distributed Processing (Middleware'98), 15-18 September 1998, The Lake District, England.
- [Williamson] Carey L. Williamson, Tim G. Harrison, Wayne L. Mackrell and Richard B. Bunt "Performance evaluation of the MoM mobile multicast protocol," *Mob. Netw. Appl.* 3, 2 (Aug. 1998), Pages 189 - 201
- [Wroclawski 97A] J. Wroclawski, Specification of the Controlled-Load Network Element Service, Internet Request for Comments RFC 2211, September 1997. <ftp://ftp.isi.edu/in-notes/rfc2211.txt>
- [Wroclawski 97B] J. Wroclawski. The Use of RSVP with IETF Integrated Services, Internet Request for Comments RFC 2210, September 1997. <ftp://ftp.isi.edu/in-notes/rfc2210.txt>
- [Wroclawski 99] J. Wroclawski. et al., LCS Communicator Web, <http://lcscomm.lcs.mit.edu>
- [Yuan] R. Yuan, S. K. Biswas, L. J. French, J. Li, and D. Raychaudhuri. "A signaling and control architecture for mobility support in wireless ATM networks" *Mobile Networks and Applications* Volume 1, Issue 3 (1996)
- [Zinky] Zinky JA, Bakken DE, Schantz RE. Architectural Support for Quality of

Service for CORBA Objects. Theory and Practice of Object Systems. April, 1997.

[Zorzi] M. Zorzi and R. R. Rao. Energy efficiency of TCP. In The Sixth IEEE International Workshop on Mobile Multimedia Communications (MOMUC), November 1999.