From *Proceedings of the Mobidata Workshop on Mobile and Wireless Information Systems*, Rutgers, New Jersey, November 1994.

Software Architecture of the Infopad System

My T. Le
mtl@eecs.Berkeley.EDU

Srinivasan Seshan
ss@cs.Berkeley.EDU

Frederick Burghardt
Jan Rabaey
Electrical Engineering & Computer Sciences Department,
University of California, Berkeley

Abstract

This paper describes a software architecture for the Infopad mobile computing system. The software architecture, called Infonet, supports the use of multimedia services in the Infopad mobile system. First, an overview of the Infopad project is presented and the project's basic characteristics are outlined. Next, the Infonet architecture is defined and its use is illustrated by three scenarios: pad activation, application initialization, and handoff. Measurements from an existing prototype demonstrate the system's feasibility. Finally, the future direction of work on the Infonet system is discussed.

1 Introduction

Infopad is an on-going project in the Electrical Engineering & Computer Sciences Department at the University of California at Berkeley. The goal of this project is to create a mobile computing environment which supports multimedia. An overview of Infopad system is shown in Figure 1 and is described in [Sheng92].

The basis of the Infopad system design is the Pad, a portable terminal capable of displaying motion video, audio, graphics, and text. The Infopad system places no general purpose computation resources on the Pad. All computation needed to make the Pad more than a display device must occur on servers that reside on a wired backbone network. These servers use the Pad as a multimedia display and I/O device. This paper discusses the software necessary to support the operation of the Pad.

An indoor cellular radio network attaches every Pad to the wired backbone network . In each wireless cell, a base station connects the cell to the backbone network. A radio frequency (RF) link connects a Pad to its nearest base station. This radio network provides a contention-free 1-2 Mbps channel for each Pad in the Infopad system. There is also a contention-based control channel that a Pad may use to communicate with various base stations. Packets from the applications that run on the backbone network must be routed to the destination Pad via the appropriate wireless cell and base station.

Any software architecture developed for the Infopad system must support the constraints created by the above hardware topology. In addition, the design must meet several additional requirements:

- Transparency

  The system should maintain compatibility with current workstations. By providing an interface similar to a fixed location workstation with a local computation, the Pad can use standard desktop applications.

- Scalability

  The system should scale easily to add more users, to use faster communications, or to support additional applications.

- Frequency and Power Allocation:

  The building in which the Infopad system operates is divided into wireless network cells. Each of the cells covers an area of 10 meters in diameter. Since a cell covers a small area, there will be multiple cells within a building. Nearby cells use different CDMA codes to avoid interference. Power and CDMA codes must be carefully allocated to maximize the performance of the wireless network.

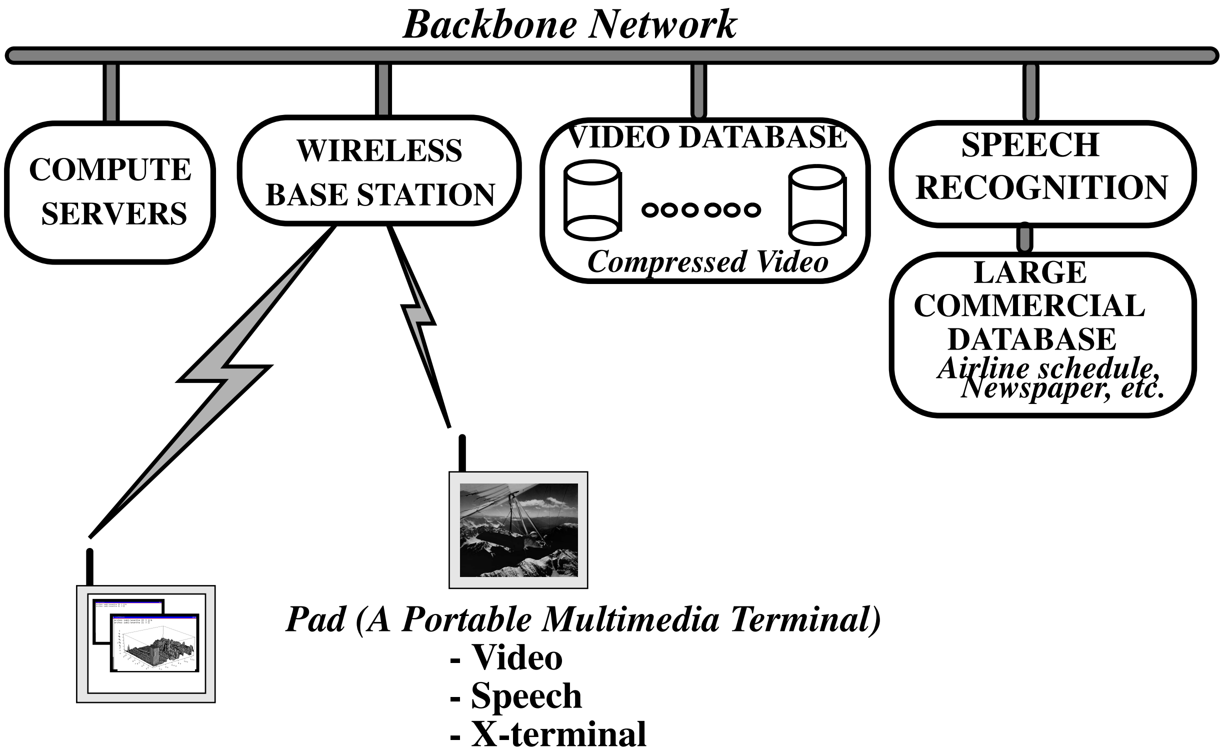

FIGURE 1. Infopad Mobile Computing System

- Mobility

  The system must make the mobility of the Pad transparent to communication between the applications and Pad. This requires the communication handoff between cells of the wireless network be done in a quick and efficient manner. Techniques for this handoff are discussed in [Keeton93].

- Quality of Service

  Applications supported by the Infopad system have different communication requirements. For example, video requires high bandwidth and low reliability while user input/commands require relatively low bandwidth and high reliability. To support these requirements, the system must provide the ability to request and guarantee different Qualities of Service (QOS) on the wired and wireless networks.

- Low Latency

  Since no general purpose computation occurs on the Pad, applications and user interface software execute on computers connected to the backbone network. To provide good interactive response to the user, low latency network communication is needed between the Pad and these applications.

Several items, such as support for multiple administrative domains and security, were not given special emphasis in the software architecture. We assumed that the Infopad would be operating in a trusted environment and within a single administrative domain. Infonet, a software architecture for the Infopad system, was developed using the requirements listed above. The goals of the Infonet system differ substantially from other efforts. The most important difference is that Infonet attempts to support mobility and multimedia service without the use of processing on the mobile device [Goodman91] [Sheng92].

This paper presents the architectural model that was chosen to support the wireless link and multimedia applications. In addition, data collected from an existing prototype are presented to demonstrate that the architecture proposed is feasible. Finally, we discuss the future direction of the project.

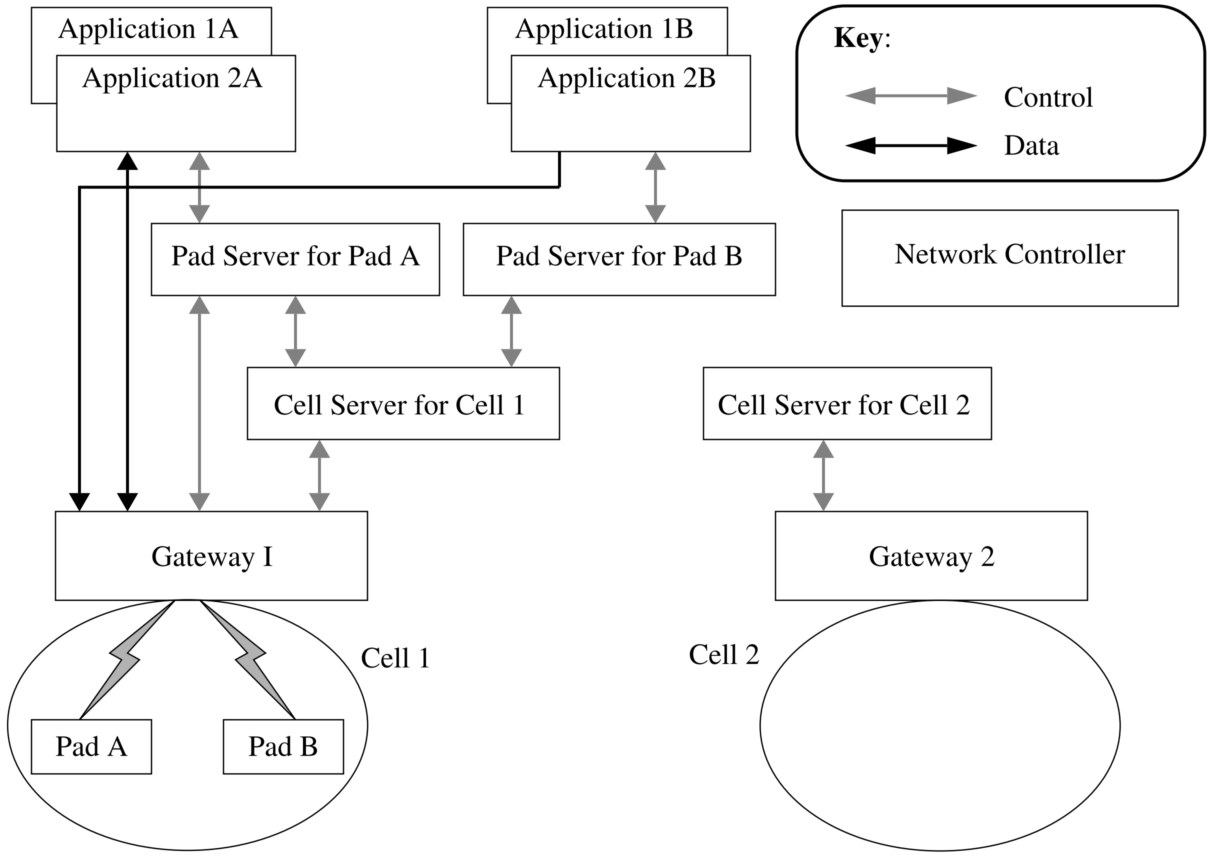

FIGURE 2. Infopad Architecture

2 Software Architectural Model:

The Infonet architecture (Figure 2) is composed of the following six groups of software elements:

- Applications:

  Applications generate data streams to the Pads. Applications include: speech recognition, video playback, text and graphics processing.

- Pad Server:

  For each Pad, there is a single Pad Server running on the backbone network. The Pad Server is responsible for managing and controlling access to the Pad. It allocates the terminal bandwidth and resources (speaker, microphone, etc.) among the different applications. To perform its duties, the Pad Server must maintain information about the Pad's state, including radio bit-error-rate (BER), received radio power level (RSSI) and current location.

- Cell Server:

  There is a single Cell Server associated with each cell. This Cell Server controls the allocation of resources among the Pads within the cell. Since resource allocations within a cell may affect nearby cells, each Cell Server must occasionally negotiate with the Cell Servers of nearby cells. A Cell Server must maintain information about the current status of its cell, such as radio bit-error-rate (BER), frequency, codes, and power to the pads in the cell.

- Gateway:

  In addition to a Cell Server, a Gateway is also associated with each cell. As shown in Figure 2, the Gateway connects a cell of the Infopad's proprietary wireless network to a standard backbone network, such as Ethernet or Asynchronous Transfer Mode (ATM). The Gateway is responsible for converting protocols between the wired network and the wireless network.

- Network Controller:

  The network controller's main responsibility is to enable the creation of connections among the other entities of Infonet architecture. It must provide name service support for the Pads, Pad Servers, Cell Servers and Gateways and maintain information about the physical location of these elements. In some network technologies, such as ATM, the network controller may also route and create the connections between the different entities.

- Pad:

  The Pad is designed as a simple multimedia display device with little intelligence. All computation and processing is done remotely. The Pad is designed to operate with the following applications: speech recognition, video playback, text and graphics processing.

In the following section, three scenarios are presented to illustrate how the system provides services to a Pad.

3 Scenarios

The following three scenarios represent some of the basic activities the Pad must support. In the first scenario, the activation of a new Pad is discussed. This initialization creates an environment that allows the users to login and request applications to be run. The second scenario describes how applications are connected to the Pad device. In this scenario, the system allocates resources to meet the QOS requested by the application. The final scenario demonstrates the mobility of the user in the Infopad system. As the Pad travels between cells, the task of forwarding data between the wired and wireless network is transferred to a different gateway.

For each scenario, a short description of the actions desired from the user's perspective is given. How the basic blocks of Infonet software architecture interact to produce this desired result is also described. Numbered lines correspond to control messages exchanged between basic blocks of software architecture.

3.1 Pad Activation

The first activity for a user is to power up a Pad. The system must create an environment that allows the user to login and request services. The system should also allow users to suspend the operation of a Pad and resume work later. To perform the Pad activation, a new Pad Server must be started or an existing Pad Server must be reconnected to the Pad. Also, the system needs to create the necessary control connections between the Pad Server, Gateway and Cell Server. Figure 3 illustrates the communication involved in this task. The communication is described below:

1. Each Pad in the system is hardwired with a unique ID. When a Pad is powered up, it transmits the ID to the nearest Gateway across a wireless control channel (message 1).
2. The Gateway recognizes this as a request to resume operation of a Pad or to activate a Pad for the first time. The Gateway forwards this ID to the associated Cell Server (message 2).
3. The Cell Server allocates a wireless channel, bandwidth, and associated resources for the Pad to operate within the cell. The Cell Server transmits information about this resource allocation to the Pad (message 3).
4. Also, the Cell Server must notify the Network Controller of the Pad's current location (message 4).
5. The Cell Server then queries the Network Controller to locate a previously running Pad Server (message 5).
6. If a Pad Server is already associated with this Pad, the Cell Server informs the Pad Server of the Pad's current state and location (message 6). This allows the Pad and Pad Server to reconnect and resume operation.

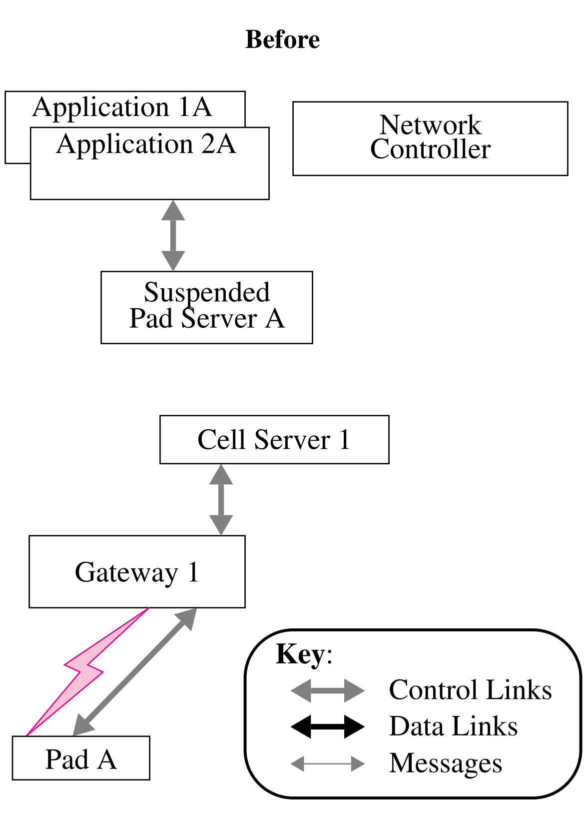

FIGURE 3. Pad Activation

7. If no Pad Server is present in the network, the Cell Server must initiate the start-up of a new Pad Server. The Pad Server executes the applications to authenticate the user. After authentication has been successful, other applications can be started. This authentication may be done by providing a login prompt and using handwriting recognition.

3.2 Application Initialization

When the an Application is executed, it attempts to create a connection with a certain QOS to the Pad. This requires that connections be created between the Application, Pad Server, and Gateway. Figure 4 depicts the communication necessary for an application to be connected to a Pad. The communication is described below:

1. The Application starts by transmitting the connection request and desired QOS to the Pad Server (message 1). This QOS is expressed using parameters such as bandwidth requirement, data loss allowance and latency limit.
2. The Pad Server determines if there are sufficient resources available on the link to the Pad. If there are enough resources, the Pad Server provides the application with a connection to the Pad device (5). If the resources are unavailable, the Pad Server attempts to negotiate with the Cell Server for more of the cell's resources (message 2).
3. To satisfy the Pad Server's resource request, the Cell Server may have to negotiate with the adjacent Cell Servers (message 3). This negotiation is needed due to the fact that resource allocation in one cell may affect communication in nearby cells.
4. Once the negotiation with adjacent cells is complete, the Cell Server notifies the Pad Server if the desired resources are available.
5. If the answer is affirmative, the Pad Server creates a connection (5) between the Pad device and application. If refused, the Application must try to create a lower QOS connection.

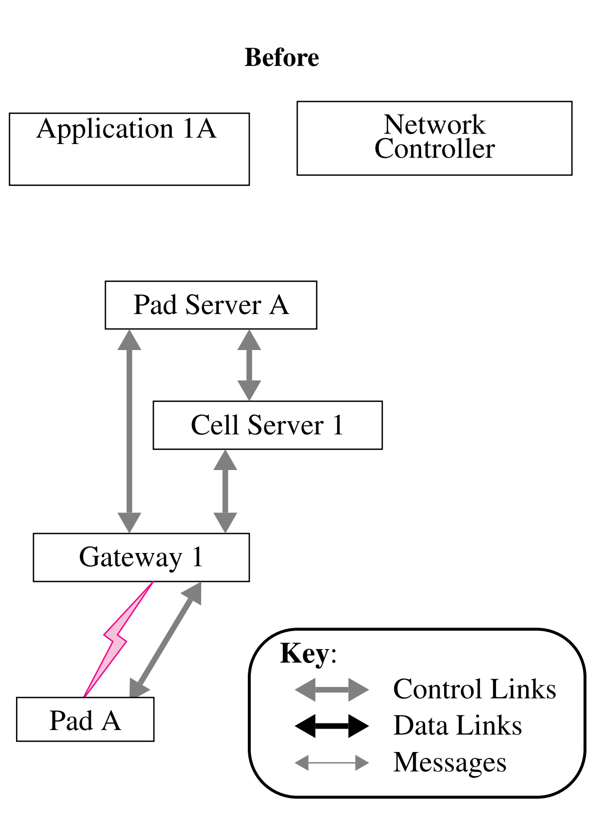

FIGURE 4. Application Initialization

This mechanism does not support feedback of changes in QOS guarantees to the application. These QOS variations may result from changes in the quality of wireless communications (due to interference or mobility). Methods to report QOS modifications to the application are being investigated.

3.3 Mobility

The Infopad system's wireless network is divided into geographically defined cells. A Gateway process forwards data between the wired network and a single cell of the wireless network. When a Pad moves between cells, its user would like to continue using applications without noticing any disruption. To support this, communication must be automatically rerouted to use the Gateway of the current cell. This process of rerouting communication from the old Gateway to the new Gateway is called handoff. Other schemes, such as Mobile IP, have been developed to deal with routing data to mobile computers. However, these schemes expect intelligent processing in the mobile device. They were designed to support devices that can execute applications and process a full network protocol stack locally. Since the Pad has no processing capability, different methods are used to support handoff in the Infopad system.

There are two important variations of handoff in the Infopad system, Requested Handoff and Lost Contact/Unexpected Arrival. A Requested Handoff occurs when the system determines that the quality of the link between the Gateway and Pad has degraded significantly. The Pad Server is responsible for tracking the link quality and initiating Requested Handoff. During handoff, the system attempts to find a new cell that would provide better quality communication to the Pad and reroute communications to this cell. Alternatively, the link quality between the Gateway and Pad may degrade to the point of breaking the connection. At this point, the Pad must begin negotiation within a new cell and perform a Lost Contact/Unexpected Arrival handoff. We expect Requested Handoff to occur more frequently than Lost Contact/Unexpected Arrival. The communication involved in Requested Handoff is shown graphically in Figure 5. The communication is described below:

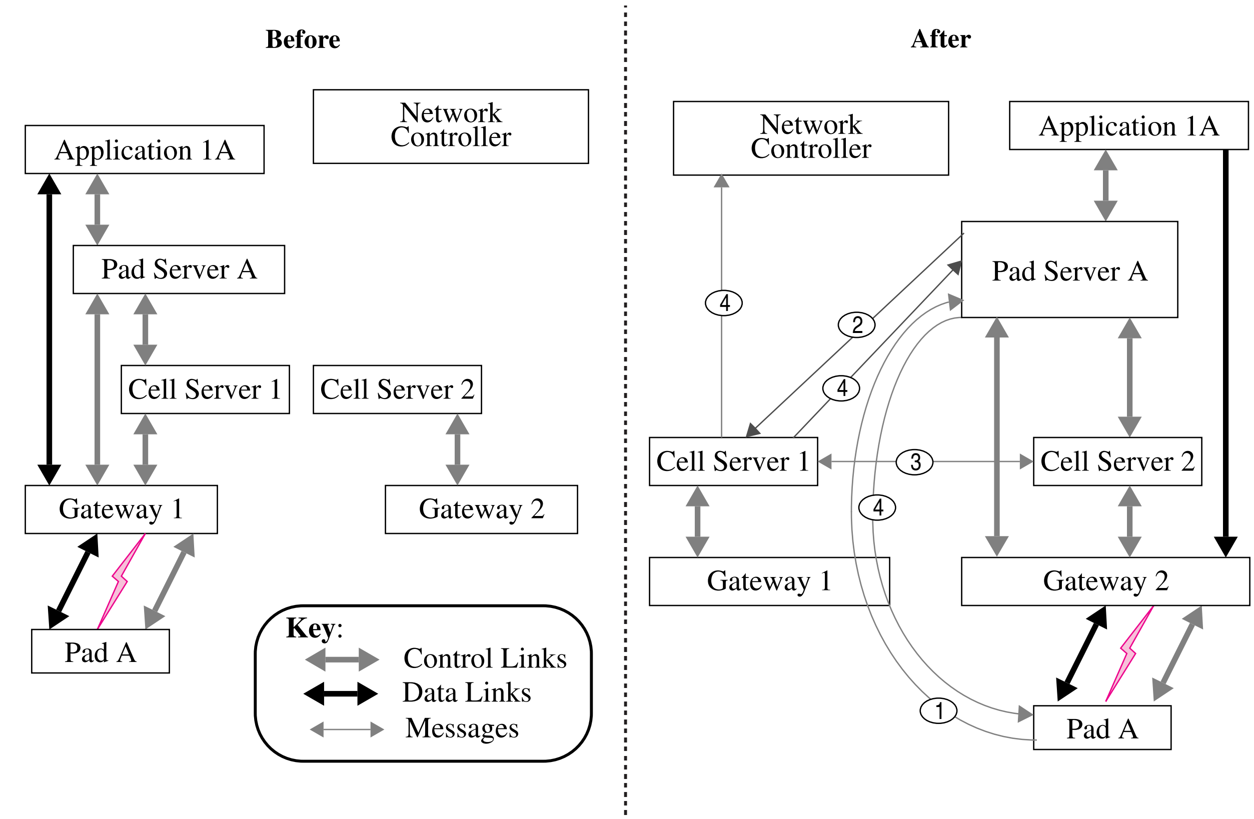

FIGURE 5. Requested hand-off.

1. For a Requested Handoff to occur, some part of the system must determine that a handoff will improve the communication quality. The information necessary to perform this decision is the BER of data received by the Pad and the received signal strength of transmissions in nearby cells. The Pad Server queries the pad for this information (message 1). Characteristics of RF communication (such as local fades) result in short lived drops in communication quality. To avoid performing a handoff in these situations, successive queries must indicate that a handoff is desirable.
2. After these queries, the Pad Server informs the Cell Server to begin a Requested Handoff (message 2). This handoff request contains the information the Pad Server has about the signals received by the pad.
3. The Cell Server uses this information to begin negotiating with the servers of the appropriate adjacent cells (message 3). During this negotiation, the Cell Server attempts to obtain the resources needed by the Pad in the adjacent cell (radio channel, bandwidth, etc.).
4. Once the necessary resources are allocated, the Pad Server, Pad and Network Controller is notified of the new cell and Gateway (messages 4).
5. At this point, the Pad modifies its radio configuration to communicate with the new Gateway and the Pad Server reroutes its communication and the applications' communication to the new Gateway. This completes the Requested Handoff.

A Lost Contact/Unexpected Arrival handoff is dealt with similarly to a Pad activation. When a Pad loses contact with a Gateway, it attempts to begin a Pad Activation negotiation with the first Gateway it encounters. Since the Pad already has a Pad Server and applications associated with it, the initialization proceeds similarly to the resume feature of the Pad Activation. This form of handoff should take much longer to complete.

4 System Feasibility

A functional prototype (Figure 6) was built using a slightly modified form of the architecture. This prototype was used to determine the feasibility of the system architecture - in particular, whether it will meet the latency requirements for applications. It was also used to determine what architecture is required for a fully operational system. This early prototype does not support handoff or provide QOS guarantees. A more recently completed version does include handoff support similar to that described in previous sections, as well as all software modules except the Network Controller. Detailed performance measurements on the current prototype have yet to be taken. The first prototype includes:

- Pad:

  A prototype Pad has been developed, which supports full X-window graphics, bidirectional audio and pen-input. A 640x480 pixel black and white LCD screen (a Gazelle Graphic System) is used for display purposes. Color video is available on a separate display [Sheng92].

- Link:

  The link between the Pad and the system is implemented with the Ariel SBUS interface card. It is a wired full duplex 1Mbps connection designed to simulate the wireless link.

- System Modules:

  The system is composed of four application modules: customized X-Server (X11R5), Notebook Pen Application, Pen Server, and a rudimentary Video Server. It also includes three system modules: Pad Server, Cell Server, and Gateway. All modules were implemented as UNIX processes and can reside on separate machines. The modules communicate with each other via TCP/IP and Ethernet.

One goal of the prototype was to reduce pen loop-back latency to under 30ms. Pen loop-back latency is defined as the time between pen contact and the appearance of “ink” on the display. This number was chosen to ensure that the user does not experience excessive delay in the system’s visual response.

The pen data must travel from the Pad to the Gateway, Pad Server, Pen Server, Notebook Application, X Server, Pad Server, Gateway, and back to the Pad. Network communications were performed over an Ethernet network for these measurements. Figures 7 and 8 show the distribution of the latency for different test conditions.

Figure 7 shows the histogram of round-trip time when the system operated with only pen data. In this case, most delays were under the 30ms target. Figure 8 shows round-trip time when the system is operated with both the pen data and full-motion compressed video. The pen data and video streams were interleaved in the Pad Server. The majority of the delays remained less than 30ms; however, the distribution was wider than before. It is noteworthy that the two graphs are very similar, indicating that other traffic to the same Pad does not noticeably degrade visual response time. However, the use of multiple Pads in the same cell could easily saturate an Ethernet, resulting in unacceptable latencies.

5 Summary and Future Directions

The presented results clearly demonstrate that the architecture is feasible and realizable. It is apparent, however, although the current multi-Pad prototype is adequate for a system with a limited number of pads, the performance will become unacceptable as the number of Pads increases for three reasons:

- An Ethernet network operating at 10Mbps cannot provide adequate bandwidth for a large number of Pads (> 10); with each Pad consuming approximately 1Mbps.
- Collisions make Ethernet highly non-deterministic. This leads to latencies far greater than 30ms when the network is heavily loaded.

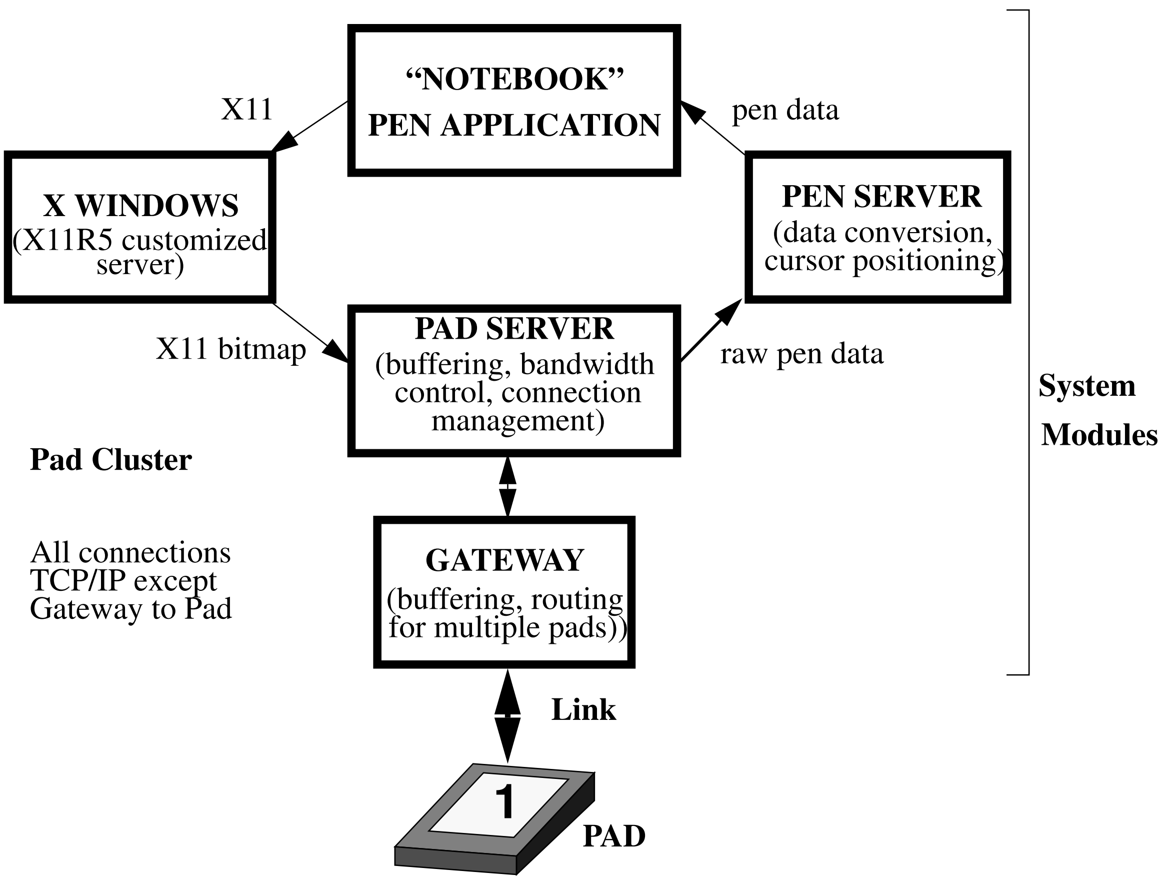

FIGURE 6. Portion of First Prototype

- Throughput restrictions in the Gateway arise from high read and write system call frequency, particularly with small packets.

The next Infonet system will use an ATM network and a hardware implementation of the Gateway. The ATM network will provide more deterministic performance and higher bandwidth. This and the hardware Gateway will eliminate many of the current problems.

The prototype architecture described in this paper will be used as a testing environment. The performance of Infopad will be measured and experiments with various hand-off, bandwidth control, error-correction and power control mechanisms will be performed on this prototype. Since the goal of the project is to provide multimedia services in a ubiquitous environment, future efforts will be focused in two areas:

1. Quality of Service:

   The applications' QOS requirements (bandwidth, delay, reliability) need to be translated into backbone and wireless networks parameters (power level, error-correction, number of Pads per cell, etc.). The network parameters will be incorporated so that QOS can be provided effectively.

2. Mobility:

   A protocol for mobility is needed to ensure that service is not interrupted when the Pad moves from one cell to another. The protocol must keep track of the mobile Pads' locations and deal with handoffs in a quick and efficient manner.

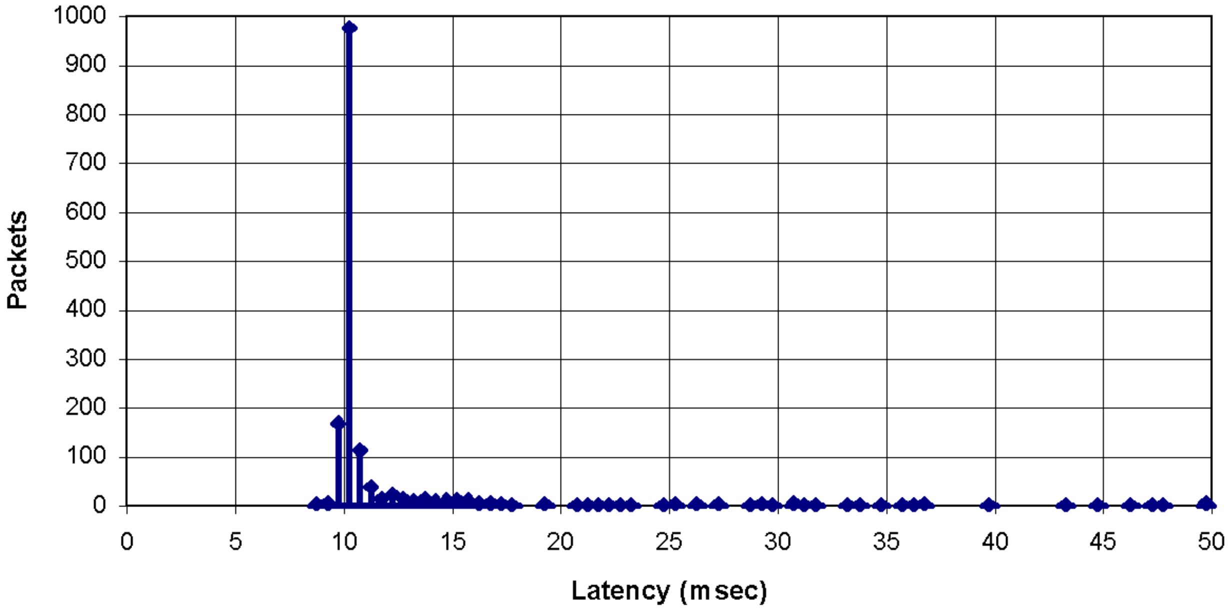

FIGURE 7. Round-trip delay of pen data (pen data only).

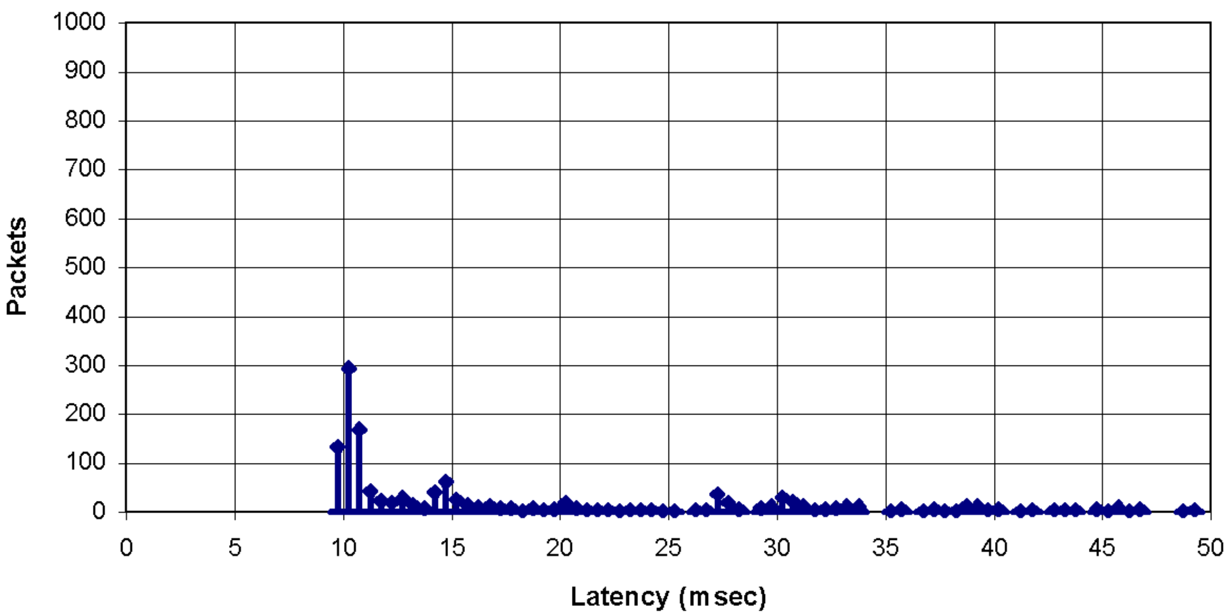

FIGURE 8. Round-trip delay of pen data (pen data and video)

The next implementation of the Infonet architecture will incorporate most of the current prototype software. The Gateway, Pad Server, and Cell Server will be modified to include more functions. In addition, a new module, called the Network Controller, will be implemented. As shown in Section 4, Ethernet cannot meet the latency requirements when there are multiple pads in the system. An ATM network operating at 155Mbps will be used in the next implementation to ensure that the Backbone network is not the bottleneck of the system.

The main weakness of the current design is the lack of support for internetworking, security and QOS changes. There is no mechanism for visitors with Pads to operate in other administrative domains. Report of QOS changes to applications is not yet supported. Also, user authentication or encryption is not available. We plan to address these issues in future designs.

Considering the measurements from the prototype, we are confident that the architecture will meet the basic service requirements. With the additional functionality, the software architecture will be able to provide the transparency, scalability, and mobility requirements.

6 Acknowledgment

We would like to thank all the members of the project for their tireless efforts in implementing Infopad. The architecture was developed by the Infonet group whose members include: Frederick Burghardt, Richard Edell, Richard Han, My Le, Ken Lutz, Jan Rabaey, Brian Richards, and Srinivasan Seshan. Our thanks to Nick McKeown and Randy Katz for their comments and suggestions in writing this paper. The Infopad project is supported by ARPA and numerous industrial corporations.

7 References

[Sheng92] Sheng, S., Chandrakasan, A., Brodersen, R.W., "A portable multimedia terminal," IEEE Communications Magazine, December 1992, pp. 64-7

[Keeton93] K. Keeton, B. Mah, S. Seshan, R. Katz, and D. Ferrari, "Providing Connection-Oriented Network Services to Mobile Hosts," Proceedings of the 1993 USENIX Symposium on Mobile and Location-Independent Computing, Cambridge, Massachusetts, August 1993.

[Goodman91] Goodman, D.J., "Trends in Cellular and Cordless Communications." IEEE Communications Magazine, June 1991, pp. 31-40.

[IEEE93] IEEE 802.11, "Draft Standard IEEE 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE P802.1-93/20b0.

[Chandra94] Chandrakasan A., Burstein A., Brodersen R., "A Low Power Chipset for Portable Multimedia Applications," Digest of Technical Papers, ISSCC 1994, pp. 82-83, San Francisco, February 1994.