THE ARCHITECTURE OF METRICOM'S MICROCELLULAR DATA NETWORK™ (MCDN) AND DETAILS OF ITS IMPLEMENTATION AS THE SECOND AND THIRD GENERATION RICOCHET™ WIDE-AREA MOBILE DATA SERVICE.

Michael W. Ritter, Robert J. Friday, Michael Cunningham, Presented By Terry G. Logan
Metricom, Inc.
333 W. Julian St.
San Jose, CA 95110

***Abstract*- We describe the architecture of the MicroCellular Data Network™ (MCDN) – a wide-area, wireless, data network. This network is designed to provide its users with fast, mobile, data access to any of the available wired data networks, typically the Internet or a corporate LAN, solving the first-mile problem. Metricom sold the first generation implementation of this architecture as the Utilinet line of private networks. We do not describe it here. We describe the second-generation implementation of this architecture, currently deployed in Washington D. C., Seattle, and the San Francisco Bay Area and sold as a service under the brand name Ricochet. We also describe the third-generation implementation, based on the MCDN(3G) architecture, which provides 10x improvements in performance and is currently deployed nationwide in 14 metropolitan areas covering over 35 million people [1]. We show some laboratory measurements of the throughput and fairness of the system. Finally, we address the possibilities of extending the architecture to a fourth generation with another factor of 10x improvement.**

I. INTRODUCTION

A. What is MCDN?

MCDN is the underlying architecture for the Ricochet system. Ricochet is a wide-area, wireless, packet data network and also the brand name under which the service is sold. This architecture (see Fig. 1) has seven physical components: wireless modems, clusters of MicroCells, Wired Access Points (WAPs), a nationwide wired backbone, a Name Service, a Network Management System, and Gateways. The gateways convert packets from the MCDN protocol back to the original protocol provided by the user's device, typically IP.

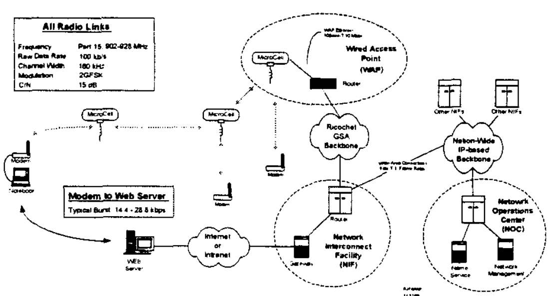


Fig. 1: Ricochet Network Architecture (Physical)

A meshed network of MicroCells surrounds every WAP, where packets are converted from RF onto the high-speed nationwide backbone by MicroCells connected to the backbone at the WAP. Each MicroCell is a wireless packet router. The MicroCells route packets to and from the WAPs and to and from the wireless modems that convert packets from the RF into a format understood by the device the modem is attached to. Each packet has a header that allows it to be routed end-to-end independently of any other packets. The MCDN MicroCell network is a routable packet switched network like the IP-based Internet, not a circuit based network like a typical cellular telephone system or a packet over a data link system that requires negotiated hand-offs like the 2.5G or 3G cellular architectures.

B. What is the MCDN Architecture?

The MCDN architecture has three parts: an addressing scheme that allows packets to be delivered between any two entities in the system, a variety of end-to-end transports and services, and a full wireless mobile networking stack. The addressing architecture allows addresses from other network architectures (e.g., IP, X.25, and PSTN) to be included inside the MCDN architecture. This ability, along with a packet encapsulation scheme, allows MCDN to use any other network as a network layer transport and is key to the flexibility and scalability of the architecture. It also allows for simple gateways that provide interconnection with other networking architectures. The current Ricochet modem emulates a telephone modem for the user interface to the network. This is because AT commands and serial interfaces are available on almost all computers. This interface is not a requirement of the addressing architecture and other devices with other interfaces are connected to the MCDN network; e.g., many network management services are provided through directly wired connections. The addressing architecture provides a common Name Service to all network devices.

Other parts of the architecture define a variety of end-to-end transports that deliver data with different levels of service, several methods to configure and gather data from any network entity, and a common framework for interconnecting to other network systems with gateways. Finally, there is a mobile, wireless networking stack defined that implements a network architecture unique to the MCDN system. This stack includes additional services such as: self-configuration, wireless link acquisition and maintenance, and mobility; that are either different than or not available in other networking architectures.

C. *Why was the MCDN Architecture Invented?*

The MCDN architecture was invented to overcome the limitations of other networking systems when applied to the problem of providing mobile, wireless data services. Cellular architectures are designed to provide voice services that require delivery of data bits with minimal wireless delays (<100 ms) but can tolerate high bit error rates ($>10^{-3}$). In contrast, data services require low end-to-end bit error rates ($<5x10^{-6}$) to be efficient but can tolerate higher delays (>100 ms). In addition, the circuit-based architecture of a cellular network would be wasteful of bandwidth resources for a data network where the traffic is bursty and has a low duty cycle. In a routed packet network there is typically no additional delay for 'hand-offs' as each packet has the entire address required to deliver it to its destination and only the endpoints of the connection are involved in the normal maintenance of the virtual circuit. Packetization allows traffic from multiple devices to be interspersed instantaneously as required, on each link, rather than preallocating bandwidth for circuits that may not be used. This efficiency advantage more than overcomes the extra bandwidth used to address each packet. Finally, a packet-based network allows users to be continuously connected while using minimal network resources.

Standard packet based networking architectures were deemed too inflexible to easily support mobility and scalability. In particular, the IEEE Wireless LAN standards (802.11) were in a state of flux and were not designed as a publicly available WAN supporting roaming and mobility. Likewise, in the current Internet Protocol (IPv4) architecture, the addresses are tied to the physical instantiation of the underlying sub-networks and must be carefully administered. In contrast, the MCDN architecture must support mobility as a fundamental network service. A user's network element may appear at any place in the entire system, without any manual configuration of its address. The solution proposed for IP, mobile-IP, was incomplete at the time the MCDN architecture was developed. It is also inefficient in its use of network resources because of its requirement for a fixed network entity to act as a proxy and relay packets for each network element that is mobile. The MCDN architecture allows the network elements to monitor and modify the addresses in the headers of the packets between any two entities on the fly so that packets can continue to be sent between the two entities even as both network elements roam throughout the system without the requirement of a fixed proxy.

II. THE MCDN ARCHITECTURE

A. *Name Service*

1) *Motivation:* The key to any addressing architecture is the ability to name and route between any entities in the system. The MCDN Name Service provides this by mapping any pair of names of entities to a list of addresses that can be used to route packets between the two entities. This route is referred to as an "MCDN Path." The MCDN Path consists of a list of addresses of waypoints between the two entities. A waypoint can be the name of an actual entity in the MCDN system or a description of a set of entities (all of the radios at a WAP, for instance.) A waypoint can also be an address in another networking architecture such as an IP address or a telephone number. The MCDN addressing architecture defines the format of the Path and defines how to encapsulate a packet from a foreign networking system. With these two definitions, it is possible to deliver a packet between any two entities in the MCDN system, using any of the recognized foreign networking architectures as delivery systems. Implementing the MCDN addressing architecture on any foreign networking requires a simple MCDN gateway that understands the foreign network system and the aforementioned definitions for encapsulation and address format. Placement of this Path gateway between the existing MCDN system and the foreign networking system allows any of the elements in the foreign networking system to take advantage of and become routable and addressable entities of the MCDN system.

The MCDN Path (see Fig. 2) is defined as an ordered list of TLVs (Type, Length, Value fields) with a bit that defines the direction to read the list (d) and a Marker field that defines which TLV is the address of the entity to route to next. The Type defines the type of address (IP, Geographical, X.25, telephone number, etc.), the Length field defines the length of the address in 16 bit chunks, the values are the actual bits of the address in a standard or native format where available. The Version field allows the protocol to be modified in the future in a backwards-compatible manner.

The Protocol field is analogous to the field in TCP/IP. Some protocols require the Port fields to further demultiplex the packets. The Packet ID field along with the source address (first TLV) uniquely identifies the packet. The Time-to-Live field prevents the packet from looping forever under routing faults. The Name Service allows any entity to request a route to any other entity in the network. The Name Service is implemented by everyone, but most entities only forward Name Service packets to some other entity that claims it is closer to the Name Server than they are. Eventually, the request climbs up the hierarchy to reach a Name Server. The Name Server is implemented as a distributed database with a hot-standby replacement and is designed as a redundant, highly available system itself.

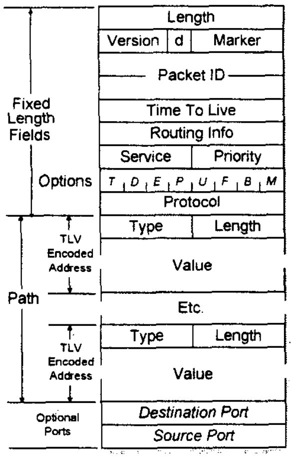


Fig. 2. The routing header including the MCDN Path

2) *Registration:* In order for the Name Service to construct an MCDN Path between any two entities in the system, it must know how to reach each of those entities on its own. Having every entity in the MCDN system register whenever it appears in a new place in the system does this. This Registration packet can be sent to any entity that advertises it provides Name Service. After an entity is registered and proved to itself that it has a path to the Name Server, it advertises that it provides Name Service; thus, the registration capability is bootstrapped throughout the network. The Name Service is implemented as a set of IP addresses to provide scalability and robustness. Once the MicroCells at the Wired Access Points are configured with this set of addresses, the rest of the wireless network finds the service automatically.

The Registration packet is a special packet that records the addresses of each Path gateway that it is sent through before it reaches the Name Server. The Name Server then stores this path in a database keyed by the unique MAC address of the registering entity. A typical MCDN registration packet would record the 802.3 MAC address of the modem, the geographic Wide-Area Networking (WAN) address of a nearby MicroCell, the WAN address of a MicroCell at a WAP along with the its IP address, and the IP address of the Name Server.

3) *Lookup:* Whenever one entity in the MCDN system wishes to exchange information with another entity in the system, it sends a Lookup packet to the Name Service requesting a name to route (MCDN Path) translation. The Name Service forwards the Lookup packet to the Name Server using the same method as in registration. The Name Server constructs the MCDN Path by concatenating together the two registration paths stored in its database corresponding to the requesting entity and the looked-up entity.

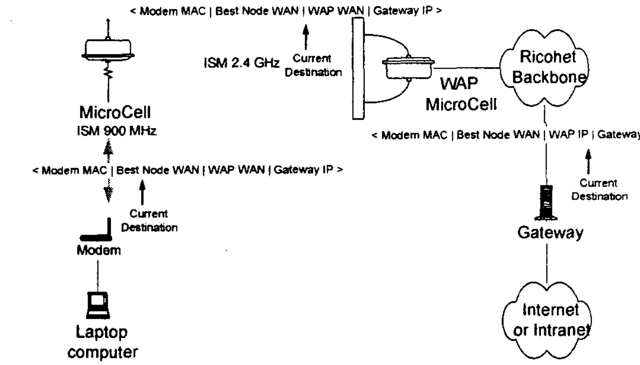


Fig. 3a: Path modification during packet routing from modem to gateway. When the packet passes through the Path gateway in the WAP MicroCell, the WAP WAN address is replaced by the WAP IP address (the return address), so that the packet can be returned correctly.

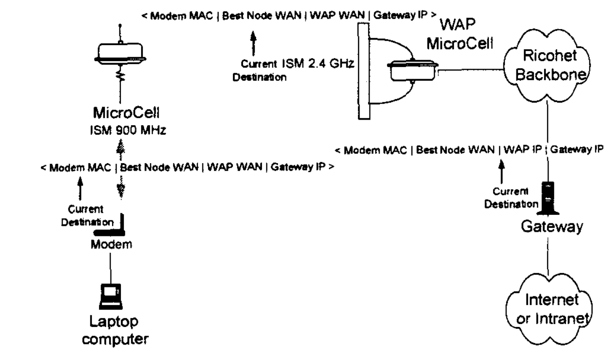


Fig. 3b: Path modification during packet routing from gateway to modem. When the packet passes through the Path gateway at the WAP MicroCell, the WAP IP address is replaced by the WAP WAN address (the return address), so that the packet can be returned correctly.

The Name Server also removes its own address from the registration paths and picks the correct address of each of the Path gateways along the path. The Path gateways between the different networking systems modify the path (see Fig. 3) as the packet traverses through them so that the return address is in the correct format; viz., the MicroCell at the WAP replaces its WAN address with its IP address or vice versa.

B. Mobility Service

1) *Motivation:* In the current implementation, there are three distinct routing systems supported: our entire wired backbone uses IP for its routing system, the wireless wide-area network uses geographically or throughput-based routing from MicroCell to MicroCell, and the modem to MicroCell link is purely a broadcast link. This last link, which has no routing, per se, forces another key capability in the MCDN architecture: the ability to reliably deliver data packets while a device is mobile. Obviously, a pure broadcast system would fail as modems could move out of range of any one MicroCell. Every MCDN entity plays a role in providing connectivity to mobile entities. This Mobility Service is

implemented via the following capabilities: instantaneous MicroCell handoff, WAP handoff, and forwarding.

2) *Instantaneous MicroCell Handoff:* Instantaneous Handoff between MicroCells takes advantage of the addressing architecture of the MCDN Path. Since there is enough information in the Path to route between any two entities, the modem can send the packet to any MicroCell at any time and the packet will be routed correctly. Thus, there is no setup time or latency caused by handoff negotiations. However, once a modem has moved, the original MCDN Path is now incorrect. The packet will use the Path to return to the modem, but it will be routed to the old MicroCell, which, presumably, can no longer deliver the packet to the modem. To address this need, the modem has the ability to modify the Path before sending the packet. The modem changes the second element of the Path to be the current MicroCell that it has the best connectivity with. This way, when a response packet returns, it will be delivered to the MicroCell that has the best chance of sending this packet directly to the modem.

3) *Forwarding:* The Instantaneous Handoff works well as long as a modem can be guaranteed to hear both the old and the new MicroCell during the time packets are in flight. However, this cannot be guaranteed in all cases. To ensure that all of the packets can be delivered, the modem will send a routed forwarding message to the old MicroCell telling it where to forward any future packets it might receive. The MicroCell holds the forwarding addresses in its cache for a suitable amount of time (much larger than the round trip time of packets) before aging them out.

4) *WAP Handoff:* As the modem travels from MicroCell to MicroCell, eventually it will move far enough so that the best path into the wired backbone is not through the original WAP picked by the Name Service. Analogously to the behavior of the modem during Instantaneous Handoff, the MicroCell recognizes that the WAP address in the Path is no longer the most efficient return route of the packet and modifies it to include a more appropriate return WAP address. The MicroCell then reroutes the packet to the most appropriate return WAP.

C. Network Service

1) *Introduction:* The network system implemented for MCDN is a complete network stack (from the physical layer through the application layer) and follows the TCP/IP Architecture in overall form. Fig. 4 shows the various components of the network stack. Fig. 5 shows a typical packet being transported through the MCDN system and its encapsulation. A packet enters the system as a stream at the modem (through some standard physical layer such as RS-232C, USB, or PCMCIA.) The modem then cuts the stream into packets. It attempts to cut the stream on logical packet boundaries, either by time or size.

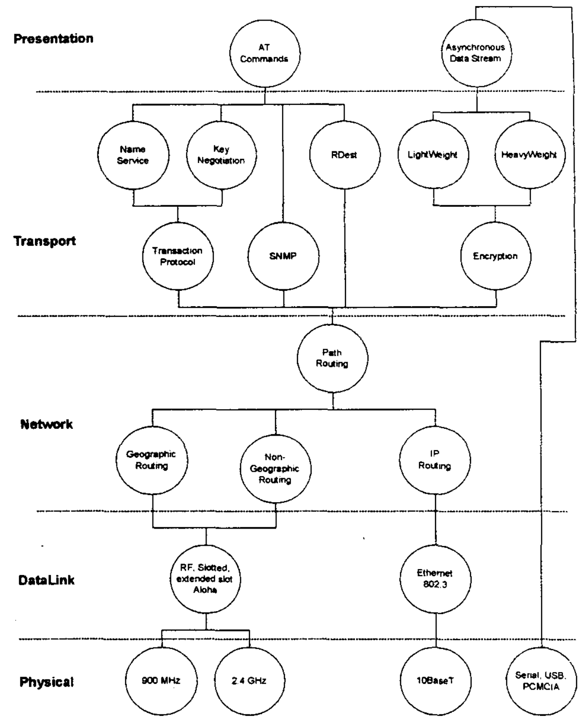


Fig. 4: MCDN network layers.

The packet is then encapsulated in the MCDN transport protocol that connects the modem to the endpoint in the system. This layer includes all of the routing and virtual connection information and ensures that the packet will traverse the MCDN system correctly. The packet is then encapsulated in the Data link and Physical Layer (for the RF portion of the network) and broadcasted to another radio in the system. For user data, the MDCN protocols attempt to make this RF link reliable by using various types of retries and forward error correction.

The packet is then routed through the MCDN system, from MicroCell to MicroCell, until it reaches a Wired Access Point. The WAP consists of a MicroCell attached to a high-speed backbone, typically a wired system using IP. The WAP MicroCell first strips off the RF physical and data link layers, it then reads the MCDN routing layer header to determine which IP address is the next destination for the packet. It then encapsulates the packet in UDP and forwards it to the next IP address, typically an MCDN gateway. At the gateway, the routing layer is stripped off and any end-to-end transport machinations required are done. The transport layer is then stripped off and the packet is delivered. In the current system, the most common delivery method is to encapsulate the PPP packet in an L2TP tunnel to a remote L2TP Network Server (LNS). The LNS terminates the PPP connection and delivers the user's IP packet.

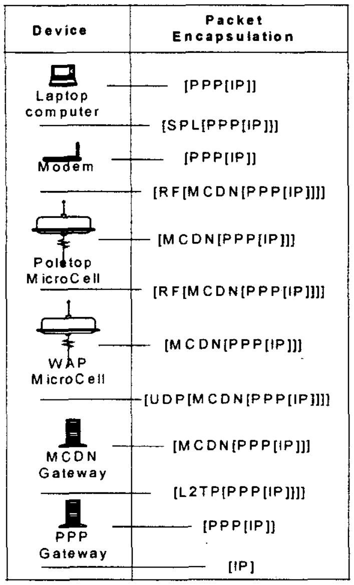


Fig. 5: Layer encapsulation through the MCDN system. The illustration shows the encapsulation of a typical IP packet as it traverses the system, both when it is inside a device, as well as when it is on the physical transport between devices. PPP is the point-to-point protocol, RF stands for the proprietary data link and physical layer MCDN protocol described in this document. MCDN is the routing and transport layer protocol described in this document, UDP is the User Data Protocol, and L2TP is the Layer 2 Transport Protocol.

2) *Transport and above*: There are several end-to-end transports, as well as network management, provided by the network stack. The system supports several modes of virtual circuits between any two entities in the system, as well as a service analogous to UDP, a best-effort packet data service. The UDP-like service is called Star Mode [2] and allows a user to address a packet to any other entity in the system by appending the entity's logical name to the front of the packet. The system also provides a TCP-like circuit service called Heavyweight Transport that guarantees end-to-end, in-order delivery of every byte. The system also provides a telephone modem-like service called Lightweight Transport that guarantees end-to-end, best-effort, in-order delivery. This transport has most of the characteristics of a modem connection over a possibly noisy circuit and is most efficient when running PPP and TCP/IP over the system. The system also provides a service analogous to AppleTalk's Transaction service [3]. This service provides a foundation for a multi-layered request-response protocol and is used to provide Name Service as well as key negotiation and accounting data.

The network management of the system is provided by an SNMP proxy that allows direct SNMP requests to any MCDN named entity on the system and forwards SNMP Trap packets to the network management system. There is also the capability of connecting directly any remote destination (RDest) in the system to monitor or configure it through the AT command interface. Code can be remotely downloaded onto any of the wireless networking entities. There are also a set of services to diagnose network problems such as Ping and Trace, analogous to the services available in IP.

3) *Network and below (Wireless):*

a) *Routing Layer:* The wireless network provides unique services for the first three layers of the network stack: Physical, Data Link, and Network in addition to all of the services described above. The Network Layer of the MCDN system is particularly complicated as compared to typical wired systems because it must be capable of handling mobility without user intervention. In addition, the current implementation complies with the FCC part 15.247 rules that require the physical layer to be based on a spread spectrum technology. The FCC rules also limit the transmit power to one watt, which severely limits the range of the system; this naturally led us to the design of a microcellular system with routing to provide coverage.

The Wireless Network layer provides routing for any packet between any two entities on the wireless network. There are two distinct versions of the routing: geographically based and latency based. The geographical system allows for minimal routing traffic but is entirely based on local information, thus may not make the best choice for system efficiency. It is based on a greedy algorithm: the MicroCells know all of the adjacent MicroCells's latitudes and longitudes (encoded in the WAN address, see Fig. 6.) Each MicroCell also keeps a running average of the amount of time it takes to deliver a packet to any of the surrounding MicroCells. Since the delivery address in the MCDN Path for the wireless network is a WAN address, the MicroCell can calculate which neighboring MicroCell it should attempt to deliver the packet to in order to maximize the velocity of the packet over the ground. The running average of delivery time causes the network to spread the load out evenly over the different MicroCells available. To handle instantaneous load (each radio in a MicroCell is half-duplex, thus can only hold a conversation with one other MicroCell at one time) a MicroCell builds a scan table. The scan table consists of an ordered list of the next best MicroCells available to receive packets. If the best choice fails often enough for one delivery attempt so that the second choice would on average provide better service, the MicroCell 'scans off' to the next best MicroCell and continues on down the scan list in turn until the packet is successfully delivered or has timed out.

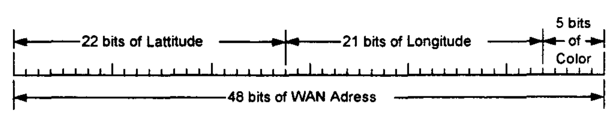


Fig. 6. The latitude and longitude are encoded as binary decimals in units of 2π radians, the color is a designator that can be used to further identify units for routing preferences. The South Pole, with all bits set, is designated as the broadcast address.

The Geographically based routing system works well in a highly connected mesh. In the field, the connectivity is sometimes so poor that the next-hop choices are severely limited in such a way as to make the system unusable unless the WAN addresses of the MicroCells are 'tuned' to provide apparent perfect connectivity. This requires excessive configuration and can be a burden on the network management system. In contrast, the throughput-based algorithm is self-configuring. It is based on a modified Bellman-Ford type routing system from the MicroCell to the WAP. Each MicroCell advertises the throughput from it to the WAP, including any queue delaying. In order to prevent looping the MicroCells also advertise hop counts to the WAP. Each MicroCell then picks a set of best next hops to form a scan table for delivery of data to the WAP. On the way into the WAP the individual hops are collected in the MCDN Path and the return packet attempts to retrace the route back out to the original MicroCell. In the case that one of the intervening MicroCells has failed, the system falls back to the geographically based routing system.

The routing layer uses fair queuing based on the final destination of the packet. There are individual queues for each destination and the second packet in the queue is not delivered until the first packet is delivered or expires.

b) *Data Link Layer*: The Data Link Layer uses a modified Aloha algorithm (based on proposals by Phil Karn) for its Media Access Control protocol. The modifications take advantage of the frequency hopping physical layer to improve the capacity of the system when lightly loaded. The MAC protocol uses three modes depending upon the size of the packet, the amount of data to be delivered, and the channel quality and system load (see Fig. 7.) The most efficient exchange of data is purely Aloha. With light load and a good channel, this rarely fails. The sending station sends a data packet and the receiving station acknowledges (Acks) the receipt of the data. Each packet is protected by a four-byte CRC. If the data packet is large or the probability of the destination being busy is high, the MicroCell uses a four-way handshake for the data exchange. First, a small request packet is sent to the destination. If the destination is not busy it responds with an Ack. Upon receiving an Ack, the originator sends the data packet to the destination. Upon successful receipt, the destination Acks the data packet, finishing the transaction. If multiple packets are queued up at either end of the transaction they can be piggy backed on the Acks until the queues are drained.

To improve the coverage of the system, the air links are stretched to their statistical limit using retries and FEC and can support our quality of service at typical bit error rates around 10^{-3}. Thus, packet failures are prevalent, including Acks. Ack failures can cause packet duplicates to appear (due to the 'scan off' capability in the routing layer.) Thus, each packet is uniquely identified and dropped if already received. Upon failure of a packet, the backoff is a combination of linear and random exponential to an individual MicroCell determined by the configuration of the transmitting MicroCell. The default configuration is two linear retries and six exponential backoff periods. After all of the backoff periods have expired, the backoff timer is reset to zero.

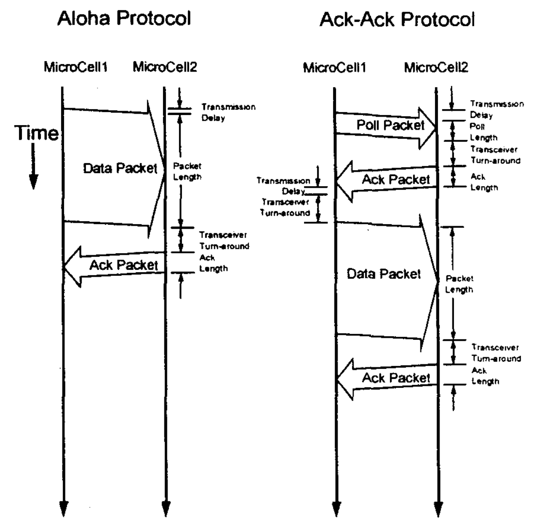


Fig. 7: For small data packets a pure Aloha MAC is used. For larger data packets or if the destination MicroCell is busy, the Ack-Ack protocol is used to increase efficiency. The third mode for the MAC protocol allows data to be piggybacked upon any of the Acks.

The MCDN architecture is also unique in that it uses an 'extended' slot at the Data Link layer: A transaction takes place on a single set of frequencies until it is finished. Typically, the MicroCells are hopping their receivers to a new frequency every 10 ms; however, when a transaction starts the MicroCell stays on the same frequency until it is finished. After the transaction is done, the MicroCell tunes its receiver to where it would have been if the transaction hadn't taken place. In a lightly loaded network this reduces contention because the new transaction attempts to use the frequency that the destination MicroCell should be on, not the frequency that it has remained on for its current transaction. (See Fig. 8.)

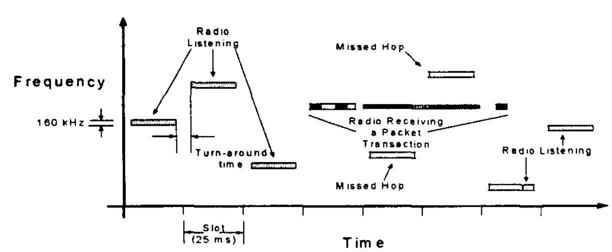
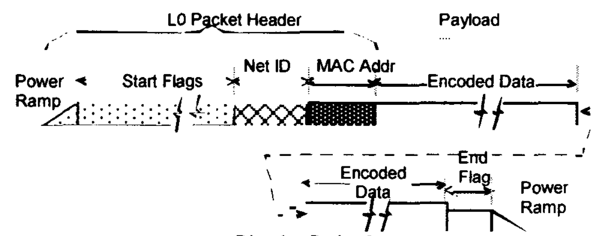


Fig. 8: Hopping sequence of a MicroCell radio and typical frequency occupation during a data transaction using the extended slot data link protocol.

c) *Physical Layer:* The physical layer is based upon a frequency hopping system designed to meet the requirements of the FCC Part 15.247 rules. These rules allow the manufacture and sale of equipment after it is certified without acquiring a license for the spectrum. There are currently three bands of interest when using these rules: 902 – 928 MHz, 2.4 – 2.4835 GHz, and 5.725 to 5.850 GHz. As can be seen, the amount of spectrum available nationwide exceeds most licensed bands by almost an order of magnitude. The rules for each band are essentially the same and require limited transmit power to reduce interference. The bands require spread spectrum physical layers, either direct sequence or frequency hopping. Frequency hopping was chosen, as it is more robust in the presence of noise and was cheaper to build. Frequency hopping radios must have a pseudo-random hopping pattern and must typically hop over 50 or more channels. This presents the network designer with two problems: first, how do the MicroCells acquire each other without any pre-configuration, and second, how do the MicroCells maintain each other after their initial contact?

In the MCDN architecture, the MicroCells acquire each other by sending short acquisition packets across the spectrum at random frequencies. An acquisition packet contains all of the information that another MicroCell needs to send a packet back to the originator of the packet (hopping sequence, timing, and MAC address.) Eventually some MicroCell will hear an acquisition packet and reply to the originator with all of the information required to send packets back to the acquired MicroCell. In this way point-to-point communication is established between two MicroCells. The next step in the acquisition process is that the originator requests the acquired radio to send the list of all neighboring MicroCells. The originator then tries to contact each of these MicroCells in turn and form its own list of neighbors. This system allows the automatic configuration of large networks of MicroCells.

In order to send a data packet to a particular MicroCell, the originator must be able to predict what frequency the destination will be listening on at that particular time. Since there is no common time source in the current implementations of the architecture, each MicroCell must maintain and correct its measurement of its neighbor's time periodically. This can be done in three ways: piggy-backed on a data packet, at the request of any MicroCell, or by listening to Heartbeats. A MicroCell calculates the maximum drift between its clock and a neighboring cell, and well before this drift prevents communication between the two radios, the MicroCell attempts to listen to the Heartbeat of its neighbor. The Heartbeat packet is broadcast periodically and contains the current time of a MicroCell (along with other information used for routing.) If a MicroCell cannot hear its neighbor's Heartbeat after several attempts, it tries to directly contact the neighbor with a maintenance packet requesting the neighbor's current time. If it cannot contact the neighbor, it drops the neighbor from its routing tables.



Ricochet Packet Layout

The bit encoding of the Ricochet physical layer is GFSK with a deviation of 70 kHz in a 160 kHz channel. The transmission bit rate is 100 kbps. The symbol encoding is HDLC, using the flag bits to recognize the start and end of a packet. The one-byte identifier in HDLC is used to designate the logical network of the device. Devices with different network IDs drop each other's packets.

III. Architecture Modifications in MCDN(3G) for the High Speed Ricochet Service

A. *Overview*

The MCDN(3G) architecture was designed to provide a Quality of Service of 128 kbps user data rate over 80 % of the time during peak usage hours. To get the higher data rate it was critical to reduce the latency in the system, including the backbone. In reviewing data from the past five years of operational experience we found that the vast majority of the traffic in the system is downloaded from the backbone to the modem. Thus, we concentrated on improving the download performance. In the end, the system was designed asymmetrically and on average provides half the performance on the uplink as on the downlink.

Metricom procured a limited amount of licensed spectrum that allows higher transmit powers and thus greater coverage. The disadvantage of this spectrum (WCS) is that it was difficult to construct a transmitter without physically large or overly expensive filters because of the stringent limits on out-of-band emissions required by the FCC. This effectively limited the high power transmissions to WAP sites, thus causing us to introduce a one-way asymmetric data link into our architecture. Finally, in order to reduce latency in the system, we added the second FCC Part 15.247 band at 2.4 GHz to the system. The modem to MicroCell link remained at 900 MHz (better for building penetration and backwards compatibility), while the backhaul to the WAPs was moved to the 2.4 GHz band. This allowed MicroCells to become effectively full duplex, thus increasing the throughput in the system by a factor of two (see Fig. 9.)

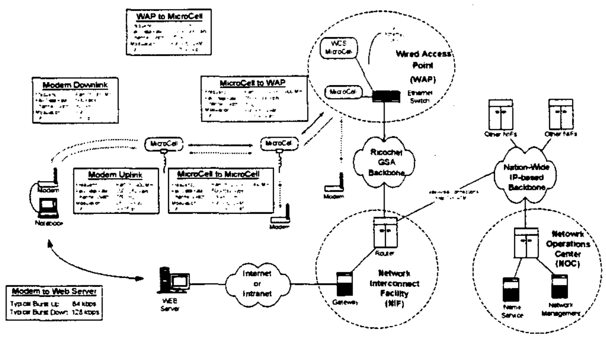


Fig. 9: MCDN(3G) Network Architecture (Physical)

B. Routing Layer

The routing layer has been modified by the addition of a one-way link from the WAPs to the MicroCells over the spectrum. When a MicroCell turns on and acquires the surrounding MicroCells it determines whether it can support the minimum grade of service on the 2.4 GHz band. If it can't, it then tries to find a WAP transmitter on the 2.3 GHz WCS band that can provide the required grade of service. If it does find a set of transmitters, it picks the one that maximizes its throughput and attaches itself with a connection message. All outbound packets, from the WAP to the MicroCell, will go through the chosen WCS transmitter. The return path is through either the 900 MHz or the 2.4 GHz band. To implement this asymmetric path, the MicroCell modifies the MCDN path of any inbound packet by adding the wired address of the WCS transmitter, so that the return path is correctly set. When the inbound packet gets to a WAP, the MicroCell at the WAP replaces its WAP address with the WCS transmitter's wired address rather than its own wired address. Thus, the return outbound packet is sent to the WCS transmitter rather than a MicroCell at a WAP.

C. Data Link Layer

The Media Access Control protocol used on the Data Link in the MCDN(3G) system on the Part 15 bands is identical to that used in the previous system. The header portion of the Data Link has been expanded with a new set of headers; the network ID is no longer in each packet and is determined at acquisition time only. The headers include packet size information so that the data link no longer depends solely upon framing to determine packet boundaries.

Two additional network services have been added at the Data Link layer for the MCDN(3G) architecture. The first is fragmentation. The data portion of a packet can be encoded into fragments, each of which includes its own, independent, error correction. If a fragment is received in error and not correctable, the receiver can request that the sender retransmit only the fragment in error, rather than the whole packet. This improves the latency and throughput of the system in a high BER environment.

The second service added to the Data Link layer is the ability to attach additional options to the packet, similar to IPv6 routing options. These options are encoded using TLVs (Type, Length, Value) and have a global definition for each type in the system. It is a simple matter to add new advisory information to the packets in a backwards-compatible manner.

D. Physical Layer

The physical layer has been expanded to include a new set of modulations (see Table I.) The acquisition is done using the old physical layer. A table of supported modulations is now traded between acquired entities, describing each modulation the entity is capable of receiving, which allows the modulations to be switched. The packet format now has a tone at the front instead of a flag byte (see Fig. 10.) The tone is used to begin frequency determination and time synchronization of the symbols. Following the tone is a set of Barker bits [5] used to determine timing and whose parity indicates the modulation of the rest of the packet. After the Barker bits there is an equalizer training sequence. Next, the header of each packet begins. The header includes link address information and packet length. The header is encoded in the most robust modulation with additional forward error correction. After the header, the modulation switches to the modulation type indicated by the Barker bits and the data of the packet begins.

Since the modulations are self-described in each packet, they can be varied throughout the system on a packet-by-packet basis to increase the throughput on individual links. The TLV system described above is used to piggyback information on the Ack from the receiver to the transmitter to describe the channel quality. The transmitter uses this information to 'gearshift' its modulation to maximize the throughput on the link.

TABLE I

MODULATIONS AVAILABLE IN MCDN (3G)

Modulatio n	Symbol rate	Signaling Rate
FSK	100 kbps	100 kbps
4-FSK	100 kbps	200 kbps
8-FSK	100 kbps	300 kbps
QPSK	250 kbps	500 kbps
16 QAM	250 kbps	1000 kbps
32 QAM	250 kbps	1250 kbps

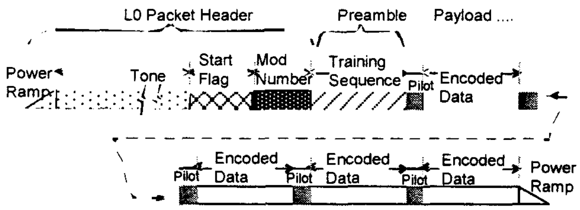


Fig. 10. Format of an MCDN(3G) Packet.

The additional performance required in the high speed Ricochet system (10x) led to the requirement that the bit rate on the downlink to the modem had to increase by a factor of five. The only effective way to do this and keep the same coverage was to double the channel bandwidth to 320 kHz. This, in turn, led to a decrease in capacity because of fewer channels. The limitation in the capacity was due to self-interference of the uplink channel of one modem with the downlink channel of another modem. To eliminate this interference, the band was split into two logical sets of channels, uplink and downlink, that are on different sets of interspersed frequencies.

For mobility support the physical layer has the ability to intersperse pilot symbols in the data portion of the packet. A pilot symbol is a known symbol that allows the receiver to calculate the changes in the channel (phase and intensity) during the packet and adjust the equalizer and data symbols on the fly. This increases the physical layer robustness under Rayleigh or Doppler fading.

IV. Future Work

A. Capacity and Performance

Reference [6] presents the theoretical capacity of the Ricochet system in the field and compares these predictions with laboratory and field measurements. In the laboratory, the MCDN architecture in its current implementation is capable of providing >30 kB/s of FTP user throughput on modems that support the 500 kb/s QAM modulation for the downlink and the 200 kb/s 4-FSK modulation for the uplink; with other modulations from Table I the system has been shown to be capable of supporting >60 kB/s of FTP user throughput. In addition, the lab data shows that the system allocates the bandwidth available to multiple users reasonably (see Fig. 11); however, there is opportunity to improve the Media Access Control protocol to perform better under load, as can be seen in the wide variance of total throughput evident as multiple users are present on the system.

B. TCP Tuning

Designing a system to perform well with data is different than designing a system to perform well with voice. The critical protocol used over data systems today is TCP/IP. It is crucial that the performance of this protocol be characterized for different network parameters. Reference [7] discusses this in more detail. If the effects of these parameters are not understood and optimized it is unlikely that the intended throughput will result. For example, recent simulations of TCP/IP web browsing experience over GPRS show that the expected user throughput was never above 30 kbps and the average user throughput was less than 15 kbps, even though the maximum theoretical burst rate of the system was over 100 kbps [8].

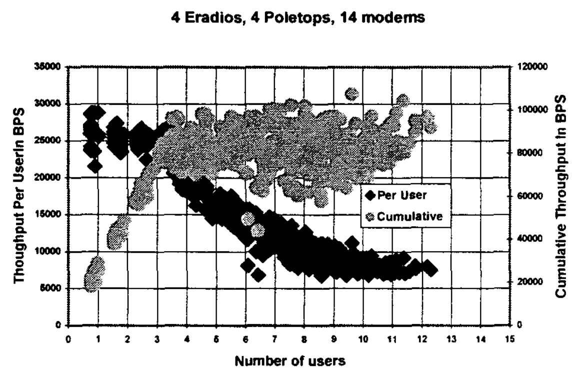


Fig. 11: The data shows the throughput per user and total throughput in bytes per second as additional modems are added to the system. The throughput is calculated by running ftp transfers of 200 kbytes in size continuously through each modem.

C. Fourth Generation

There is ample opportunity to extend the performance of the current system by another factor of 10 by using the third Part 15 band at 5 GHz to replace the 2.4 GHz backhaul used in the implementation of MCDN(3G). The 2.4 GHz spectrum could be dedicated to improving the MicroCell to Modem link bandwidth. Current experimental measurements and system design work to validate this concept is underway. Antennae Spatial processing can provide additional link budget required by higher frequencies.

Acknowledgement

Brett Galloway, George Flammer, Bob Kalkwarf, John Bettendorff, Bob Friday, Mike Pettus, and Paul Baran for design of the original Utilinet system, the precursor to the Ricochet system. The entire engineering team at Metricom for the Ricochet system.

References

[1] http://www.metricom.com/about_us/coverage_maps/index.html

[2] Unpublished, Contact Metricom, 333 W. Julian St., San Jose CA 95110.

[3] G. Sidhu, R. Andrews, A. Oppenheimer and, *Inside AppleTalk*, 2nd ed, Addison-Wesley, 1984.

[4] Phil Karn, "MACA - A New Channel Access Method for Packet Radio", Proceedings of the 9th ARRL/CRRL Amateur Radio Computer Nettworking Conference", London Ontario Canada, September 22, 1990, p. 134.

[5] I. Bar-David, R. Krishnamoorthy, "Barker Code Position Modulation for High Rate Communication in the ISM Bands", Bell Labs Technical Journal, Autumn, 1996, p. 21.
[6] R. Friday, M. Ritter, A. Srivastava, "MicroCellular Data Network (MCDN): Performance and capacity of a Broadband Mobile Wireless Technology", Networld+InterOp 2001 IEEE Broadband Wireless Summit, May 9th, 2001.
[7] A. Srivastava, R. Friday, M. Ritter, W. San Filippo, "A Study of TCP Performance over Wireless Data Networks", to be published.
[8] R. Kalden, I. Merrick, "Wireless Internet Access Based on GPRS", IEEE Personal Communications, April 2000, p. 8.