"Begin at the beginning," the King said, gravely, "and go on till you come to the end; then stop."

Lewis Carroll, Alice in Wonderland, Chap. 12

The last thing one discovers in writing a book is what to put first.

— Blaise Pascal, Pensees, I., Chap. 19

Chapter 1

Introduction and Preview

1.1 Motivation

Among the several trends in networking and communication in the 1990s, two stand out prominently: (i) the rapid global expansion and use of the Internet for information access and (ii) the massive growth in the use of wireless cellular telephony for communication. The main reason for the first trend is the meteoric growth of the World Wide Web (WWW) [18] and the large amounts of information that one can find on it. The second trend, towards wireless connectivity, has been fueled by advances in communication hardware and cellular technologies that have enabled large-scale wireless telephone networks. Figure 1.1 shows both these rapidly growing trends.

One would therefore expect that the combination of these two growing technology trends—*Internet access* on the one hand and *wireless access* on the other—would result in a potent combination and that wireless data networks and services would dominate the marketplace. However, a survey of the commercial world shows that wireless data networks and services are floundering, with a lack of widespread proliferation of such systems.

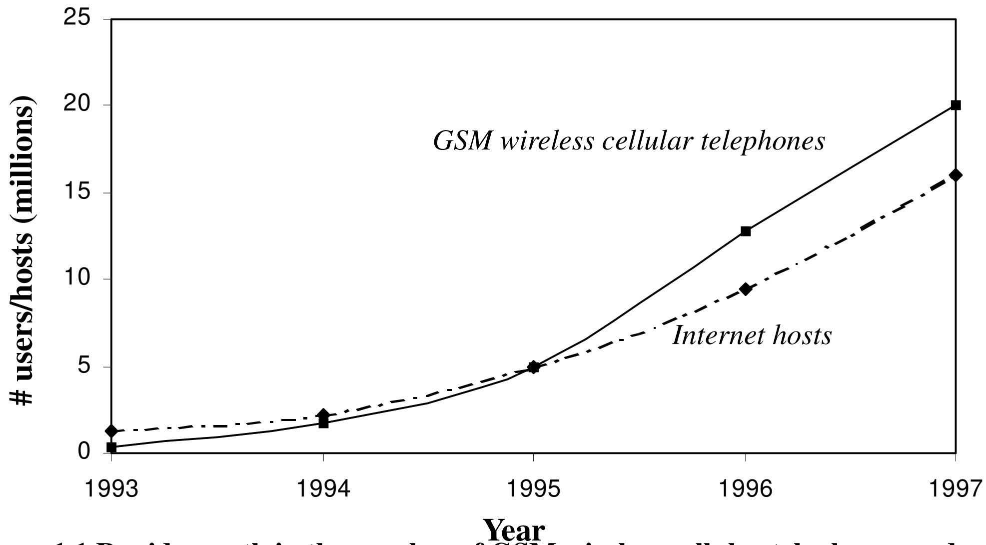

Figure 1.1 Rapid growth in the number of GSM wireless cellular telephones and hosts on the Internet as a function of time.
Sources: Ericsson, Inc. and Matthew Gray (MIT).

Why is this the case? Could it be that the inherent user demand for wireless data systems is non-existent? On the contrary, several user surveys show that this is not so, and that users do prefer the convenience offered by wireless Internet access [51]. Wireless data systems enable a plethora of new applications, especially for users on the move. For example, consider the following statement by George H. Heilmeier as a harbinger of the future [52]: *"People and their machines should be able to access information and communicate with each other easily and securely, in any medium or combination of media—voice, data, image, video, or multimedia—any time, anywhere, in a timely, cost-effective way."*

We believe that there are two important and closely related reasons that hamper the realization of this vision:

- *Poor performance:* Unfortunately, the performance of *reliable* data transport protocols over different wireless networks is unacceptably low today. Most Internet applications, including Web access, file transfers and electronic mail, require efficient reliable data transport. Therefore, the degraded performance of reliable transport protocols strongly discourages widespread use of wireless access networks, since the resulting price/performance ratio becomes too large compared to alternative wired access technologies.

- *Heterogeneity:* There is tremendous diversity in wireless access technologies ranging from wireless local-area networks to wide-area satellite systems. These technologies have different characteristics (e.g., bandwidth, latency, error characteristics, range, etc.) that make it hard to identify the reasons for poor performance. Current approaches to reliable data transport do not handle wireless heterogeneity well.

The vision of reliable and ubiquitous information access poses several challenges. In this thesis, we identify and solve the fundamental problems to efficient reliable data transport in heterogeneous wireless networks. The end result is an adaptive reliable transport protocol architecture that provides significantly improved end-to-end performance in heterogeneous networks of wired and wireless links, enabling a wide variety of wireless technologies to be profitably used in the Internet. We strongly believe that this work is an important step in making wireless devices first-class citizens in the future Internet.

The rest of this section elaborates on the issues of wireless heterogeneity and poor transport performance in wireless networks. In Section 1.2 we describe the protocol architecture of the Internet and its best-effort service model. Section 1.3 discusses the three fundamental challenges to efficient wireless data transport that we identify and overcome in this thesis. Section 1.4 presents an overview of our contributions and describes the organization of the rest of the thesis.

1.1.1 Wireless Heterogeneity

The recent growth in the number of wireless alternatives to gain access to the Internet has been remarkable. These technologies enable desktops, laptops and hand-held Personal Digital Assistants (PDAs) (e.g., [113]) to gain access to information sources on the Internet without any tethered connection. The wide range of wireless networks includes in-room Infrared networks, building-wide wireless LANs, campus-area packet radio networks, metropolitan-area cellular wireless networks, regional-area wireless cable networks and broadcast satellite networks.

Figure 1.2 shows four different wireless devices attached to a laptop computer to illustrate this point. These devices are different from each other in terms of maximum raw bandwidth, channel-access mechanisms, link protocols, frequency, power and mode of operation, etc. These differences make it hard to identify, analyze and solve performance problems.

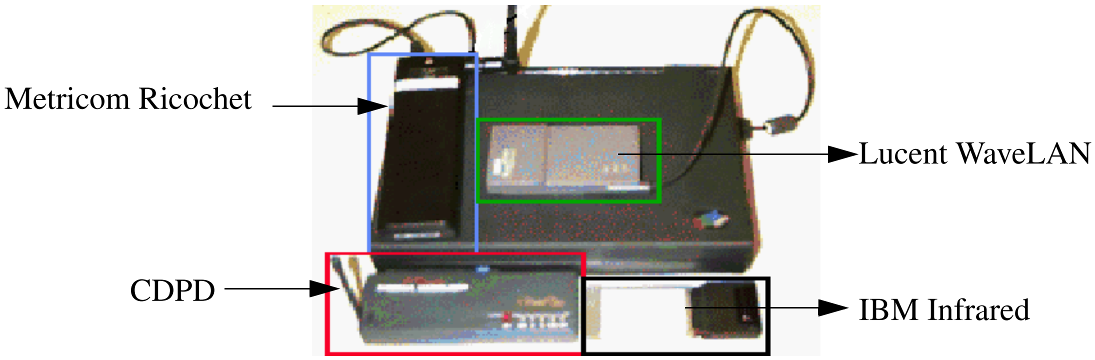

Figure 1.2 Wireless heterogeneity. Clockwise from top-left, the picture shows a Ricochet packet radio modem, a WaveLAN PC Card interface, an IBM Infrared interface and a Cellular Digital Packet Data (CDPD) modem.

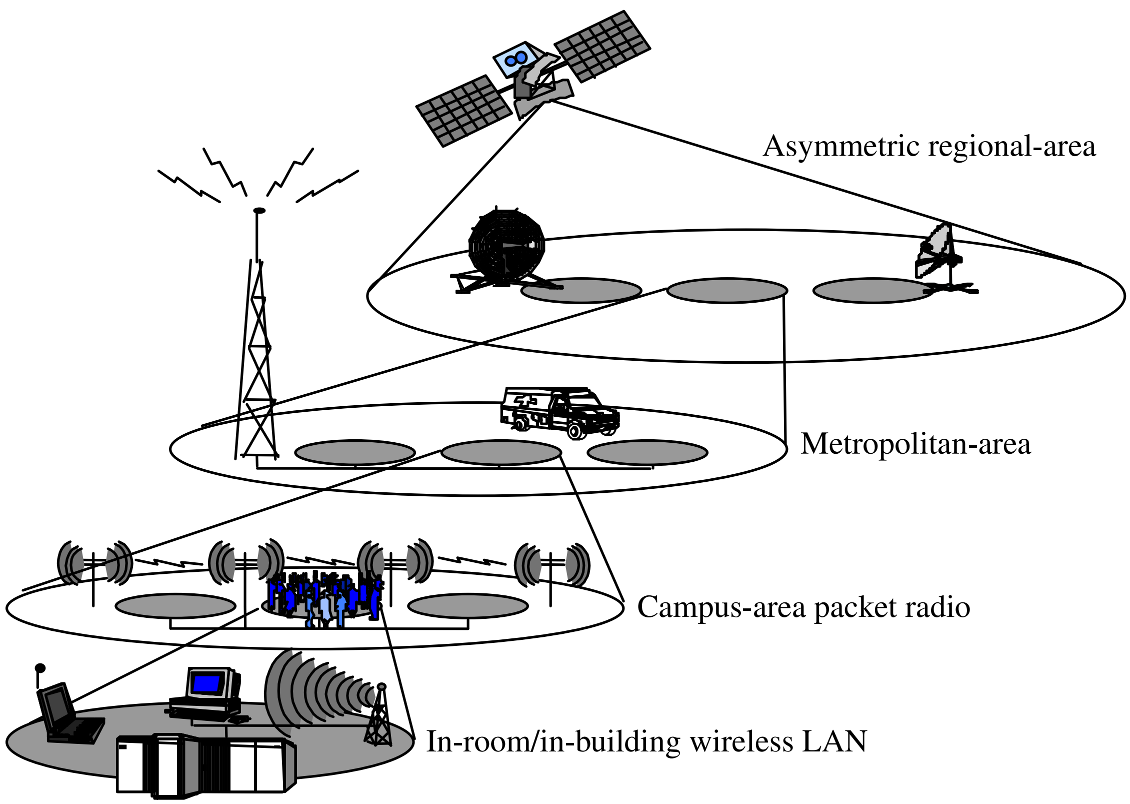

Figure 1.3 Wireless overlay network architecture.

One way of visualizing this diversity in access technologies is by using a tiered, overlay architecture as shown in Figure 1.3. At the bottom of this hierarchy are in-room and in-building networks that provide maximum bandwidths between 1 and 10 Mbps over a relatively small range. Exam-

Network	Max. raw bandwidth	1-way latency	Typical range	Salient characteristics
Room/Building wireless LAN (IR/RF)	> 1 Mbps	< 10 ms	< 300 meters	Pedestrian speed Shared medium
Campus-area packet radio (RF)	< 100 Kbps	100-500 ms (highly variable)	< 1 mile (via repeaters)	Slow vehicular Variable latency
Metro-area (Cellular, PCS, GSM)	< 20 Kbps	> 100 ms (highly variable)	Few miles	Vehicular Variable latency
Regional-area (Wireless cable)	10-30 Mbps (shared)	< 10 ms	Few miles (point-to-point)	Stationary Bandwidth asymmetry Relocatable mobility
Regional-area satellite networks	10-100 Mbps	>> 100 ms	Several hundred miles	Stationary Large latency Bandwidth asymmetry Relocatable mobility

Table 1.1 Characteristics of a variety of wireless data networks. These technology characteristics are current as of mid-1998.

ples of these include IBM's Infrared and Lucent's WaveLAN [145] technologies. The next tier is composed of campus-area packet radio networks like Metricom's Ricochet network [98]. These provide maximum link bandwidths of about 100 Kbps, within a range of a mile or less. Above these are metropolitan-area cellular data networks like the Cellular Digital Packet Data (CDPD) network [49] and the GSM data network [99]. These provide low bandwidths of less than 20 Kbps today, but their range is several miles long. Finally, at the top of this hierarchy are an intriguing set of regional-area networks, including wireless cable and direct broadcast satellites. These networks provide high bandwidth connectivity over a large shared area, but mainly in one direction alone. The opposite direction, used to request data and acknowledge successful data delivery, is over a much lower bandwidth path, often using a different technology like a telephone dialup line or a packet radio channel. Thus, these access technologies display bandwidth asymmetry, with different available bandwidths in the two directions. Table 1.1 shows the salient features of these overlay network technologies.

The result of this technology diversity is a variety of problems that adversely affect reliable transport performance in these different networks, as explained below.

1.1.2 Poor Transport Performance

The performance of reliable transport protocols over most wireless networks today is unacceptably poor. Table 1.2 shows the throughput results of several experiments over four different wireless networks for a 1 MB bulk transfer. While there is a wide range in the measured performance depending on external conditions, the median throughputs are much closer to the lower end of the spectrum in every case. Thus, there is often an order of magnitude (or more) difference between what is rated for each technology and what is perceived by the user during actual usage. Our goal is to identify the reasons for this gap between perceived and rated performance and to bridge it.

Technology	Rated link bandwidth	Measured throughput
IBM Infrared	1 Mbps	100—800 Kbps
Lucent WaveLAN	2 Mbps	50 Kbps—1.5 Mbps
Metricom Ricochet	100 Kbps	10—35 Kbps
Hybrid wireless cable with dialup return	10 Mbps	500 Kbps—3 Mbps

Table 1.2 Rated link bandwidth and measured throughput for four different wireless technologies. In each case, median performance is close to the lower end of the spectrum than the higher. There is thus a wide gap between perceived TCP throughput and vendor-rated link performance.

Our research uncovers three fundamental challenges that significantly degrade the performance of reliable data transport protocols when the wired Internet is augmented with wireless access technologies:

- The preponderance of wireless bit-errors.
- Asymmetric effects and latency variation.

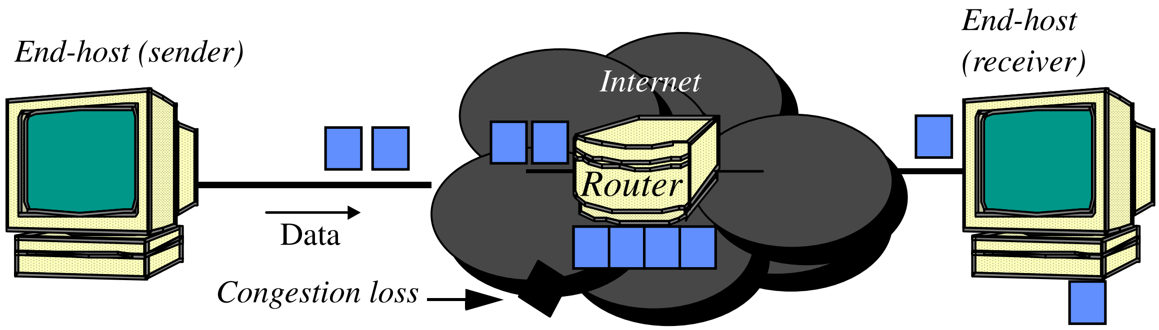

Figure 1.4 Packets between end-hosts on the Internet are routed through a set of routers. Router queues could overflow and cause packet loss during times of congestion.

- Small transmission windows due to low channel bandwidths.

Before discussing these challenges in detail (Section 1.3), we need to understand the protocol architecture in the Internet today and how reliable data transmission is achieved.

1.2 The Internet Protocol Architecture

The Internet is a *best-effort* network. Today's Internet infrastructure provides no guarantees on bandwidths or latencies and has no mechanisms to avoid packet losses. Packets between end-hosts are routed through a set of routers, as shown in Figure 1.4. Typically, in conventional wired networks, network congestion due to overload and router resource contention causes packet loss. This service model leads to a simple internal architecture that scales to handle a large number of communicating entities, but requires higher level protocols and applications to *adapt* to packet loss, varying available bandwidths and latencies.

1.2.1 Heterogeneity

The Internet is characterized by an enormous degree of heterogeneity. End-hosts range from powerful servers and desktop computers to so-called "thin clients" [113] and hand-held devices; network links range from multi-gigabit wired links to low-bandwidth dialup and wireless links; protocols range from unreliable unicast protocols (e.g., UDP [114]) to reliable multicast protocols [125]; and applications range from bandwidth-intensive video conferencing (e.g., [95]) to latency-sensitive remote terminal applications (e.g., [118). Despite this diversity, the Internet enables hosts to communicate with each other independent of the specific characteristics of the machines or

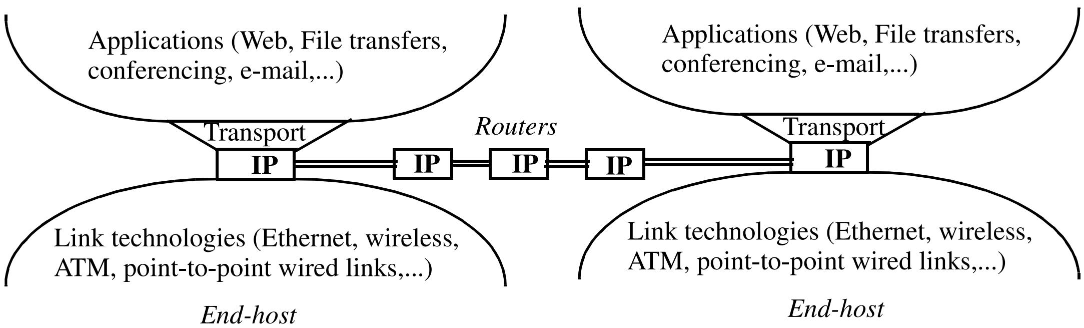

Figure 1.5 A "stovepipe" diagram showing a variety of different applications running on end-hosts communicating over a variety of different link technologies. The IP layer is the lingua franca for communication on the Internet, setting the standard for addressing nodes and routing packets between them.

access technology. This is achieved by using the Internet Protocol (IP) [115], which provides a common vocabulary for communication between hosts. That is, end-hosts can communicate with one another as long as they use IP for addressing and communication, via a network of routers that run IP. Thus, IP enables universal connectivity independent of host and link technology, making it the lingua franca of the Internet today (Figure 1.5).

1.2.2 Reliability

By bringing together a range of disparate technologies, IP solves one aspect of the heterogeneity problem: that of enabling basic connectivity between hosts. However, because the IP service model provides the abstraction of a best-effort network, packet losses can and do happen. There are numerous applications, such as World Wide Web access (e.g., using HTTP [44, 136]), file transfers (e.g., `ftp` [119]), electronic mail [117], interactive logins (e.g., `telnet` [118]), etc. that require *reliable, ordered* data delivery from the network. This is provided by a reliable transport protocol that runs over IP, that such applications can use via a well-defined socket interface [134].

To work well in the global Internet, a good reliable transport protocol must perform two critical functions: *loss recovery* and *congestion control.* Loss recovery is the ability of the sender to deduce that certain packets did not reach the receiver and retransmit them, either when requested by the receiver or without any explicit request from the receiver. Congestion control is the ability of the sender to recognize situations when the network is overloaded and react to it by reducing its transmission rate.

The *Transmission Control Protocol,* or *TCP* [116, 20, 135], is the de facto standard for reliable unicast data transport in the Internet today. It is used by virtually all unicast applications that require reliability. TCP achieves reliability by using *cumulative acknowledgments (ACKs)* that are sent by the receiver to the sender for all received data on a regular basis. A cumulative ACK acknowledges the successful receipt of all data received *in-order* so far. These ACKs are used by the sender to determine the successful delivery or loss of data. The TCP sender performs loss recovery by retransmitting packets that it considers missing, based on a lack of an ACK for it. Thus, TCP achieves reliability using an Acknowledgment-Repeat-reQuest (ARQ) [83] scheme with cumulative ACKs.

Until now, one of the main reasons the Internet has successfully withstood rapidly increasing growth is the sound congestion control principles on which TCP is based [59, 30]. TCP uses a window-based congestion and flow control mechanism, where data transmissions are governed by a dynamically changing window size that attempts to track available bandwidth and react to congestion. The TCP sender is *ACK-clocked*; it relies on timely ACK feedback from the receiver to transmit new data smoothly. Since TCP has no a priori knowledge of the state of the network, it uses the rate at which ACKs arrive to adjust its window and deduce the sustainable data rate for the connection. In response to any packet loss (at which time it also performs a retransmission), TCP reduces its window and hence its transmission rate by at least a factor of two, reducing the load on the network. An implicit assumption made by TCP's congestion control algorithm is that *all* data loss is caused by network congestion. While this is a valid assumption in conventional wired networks, it is not true in most wireless systems.

Thus, TCP's congestion control and loss recovery mechanisms are tightly coupled to each other, which leads to degraded performance in wireless networks where losses frequently occur for reasons other than congestion. Furthermore, any irregularity in the ACK stream skews TCP's ACK-clocking mechanism and manifests itself in the data stream in the immediate future. This situation arises in bandwidth-asymmetric and packet radio networks.

1.2.3 Incremental Deployment

In addition to enormous heterogeneity, the Internet is characterized by tremendous scale, as shown in Figure 1.1. By virtue of the fact that there are over 16 million hosts in the Internet today [86], "legacy" systems are an integral part of the Internet landscape. It is therefore crucial to recognize the need for an incremental and evolutionary approach towards protocol design and deployment in large-scale heterogeneous networks like the Internet [46]. Otherwise, it is hard to make impact and influence its future evolution.

There are two important aspects to incremental deployment that Internet researchers must address: (i) developing a clear deployment path from where the world is today and where they would like to take it with their protocol enhancements and (ii) ensuring that there are no adverse interactions between their solutions and existing protocols and applications. Not meeting either of these requirements would greatly reduce the widespread acceptance and deployment of any proposed scheme.

We use the TCP/IP protocol architecture as the starting point for our solutions to wireless problems rather than designing a new protocol architecture completely from scratch. There are significant advantages in adopting this approach. The first is the ability to incrementally deploy our changes in the global Internet and realize the vision of making wireless devices first-class citizens in the Internet. The second is being able to operate with currently-deployed legacy protocol implementations at certain hosts in the network, obviating the (impossible-to-achieve) need for a "flag day," a day on which all Internet hosts need to change to a new protocol architecture.

Despite these benefits, it is still justifiable to design a new protocol architecture from scratch, *if* the features in the new protocol cannot be implemented in the TCP/IP framework. Our experience with a wide variety of algorithms and techniques demonstrates the flexible and extensible nature of the TCP/IP protocol architecture. Using a combination of TCP options, local link-layer solutions, and modifications and enhancements to TCP's algorithms, we have been able to comprehensively solve many observed problems to efficient wireless data transport. Our research demonstrates the flexibility of the TCP/IP protocol architecture and strongly discourages the need to design a new protocol framework to combat the problems observed in the heterogeneous networks investigated in this thesis. Thus, our solutions enable a variety of wireless technologies to be successfully and efficiently integrated into the Internet in an incremental fashion.

1.3 Preview: Three Fundamental Challenges

The challenges to efficient reliable data transport over networks composed of wired and wireless links arise primarily because of network heterogeneity and different characteristics of the underlying wireless technologies. These lead to problems that greatly degrade the performance of protocols like TCP in networks with wireless links.

There are three key independent factors that affect the performance of reliable data transport in heterogeneous wireless networks:

1. The preponderance of packet losses due to wireless bit-errors and user mobility.
2. Asymmetric effects and latency variation due to adverse interactions between media-access protocols and TCP.
3. Small transmission windows due to low channel bandwidths.

While TCP adapts well to network congestion, it does not adequately handle the vagaries of wireless media. In this thesis, we analyze these problems in detail and solve them, using a combination of link-layer techniques and modifications and enhancements to TCP at the sender and receiver. The rest of this section summarizes the problems caused by these factors and outlines our solutions to them.

1.3.1 Challenge #1: Wireless Bit-Errors

TCP performance in many wireless networks suffers because of packet losses induced by wireless bit errors, which occur in bursts because of the nature of the wireless channel. Unfortunately, TCP wrongly attributes these losses to network congestion because of the implicit assumptions made by its congestion control algorithms today. This causes the TCP sender to reduce its transmission window in response and often causes long timeouts during loss recovery that keep the connection idle for long periods of time. The result is degraded end-to-end performance. In addition, packet losses that occur due to user mobility cause the TCP sender to remain idle for long periods of time even after the handoff is completed, resulting in unacceptably low throughput [25].

Traditional approaches to addressing the bit-error problem either operate solely at the link-layer, or split the end-to-end transport connection in two. Both these approaches are fraught with problems.

For example, pure link-layer protocols often interact adversely with TCP's loss recovery and timer mechanisms [38, 14]. Split-connection protocols attempt to shield the sender from the wireless link by explicitly terminating the wired connection at the base station and using a separate transport connection over the wireless link [149, 9, 81]. However, they do not preserve end-to-end semantics because data may be acknowledged to the sender even before it reaches the receiver, complicate handoff procedures because they involve hard state in the network, and do not usually provide the best possible performance.

Our approach to solving this problem, called the *Berkeley Snoop protocol [16],* exploits *cross-layer protocol optimizations* to improve performance. In addition, we design a mechanism called *Explicit Loss Notification (ELN)* that can successfully distinguish between congestion and channel error-induced packet losses to substantially enhance end-to-end performance. The Snoop protocol uses only *soft state* [31] at agents in the network, which is periodically refreshed upon the arrival of data segments and ACKs. It also leverages TCP's reliability messaging, without generating its own independent protocol messages.

For data transfer *to* a mobile host, the protocol deploys a snoop agent at the base station that caches unacknowledged TCP segments destined for the mobile host. The snoop agent uses the loss indications conveyed by duplicate TCP ACKs and locally maintained timers to retransmit lost data from the base station. The agent also *suppresses* duplicate ACKs corresponding to wireless losses from the TCP sender, thereby preventing unnecessary congestion control invocations. Since local retransmissions are based primarily on TCP ACKs and duplicate TCP ACKs are suppressed from the sender, we call this a *transport-aware reliable link protocol* [14]. It is a cross-layer optimization where a link-layer agent is aware of transport messaging and semantics.

For data transfer *from* the mobile host, we use a technique based on ELN to achieve good performance. The base station uses queue length information to distinguish congestion from corruption and passes this information to the mobile host's TCP using ELN as an in-band control signal. Thus, the mobile TCP sender can perform retransmissions decoupled from congestion control, which makes it a *link-aware transport protocol.*

Performance measurements of different workloads show throughput improvements between a factor of two and twenty across a range of bit-error rates for transfers in either direction, measured over a Lucent WaveLAN-based wireless LAN [145]. We present the results of experiments over an indoor installation in the Bay Area Research Wireless Access Network (BARWAN) [36], as well as over an outdoor installation in U.C. Santa Cruz's Reinas wireless network [85]. Chapter 4 presents the details of the Snoop protocol and performance measurements. In Chapter 5, we describe several experiments that precisely explain the reasons for the observed performance, demonstrate the benefits of cross-layer optimizations and compare our approach to several other schemes.

1.3.2 Challenge #2: Asymmetric Effects and Latency Variation

The second challenge arises due to the asymmetric nature of many wireless communication systems. An example of a network that exhibits asymmetry is a wireless cable modem network [24], where available bandwidth in the forward direction is much larger than that in the reverse path (a dialup line or low-bandwidth wireless link). Such *bandwidth-asymmetric* networks are becoming increasingly popular because of economic reasons and predominantly asymmetric Web workloads.

Latency asymmetry occurs when latencies in opposite directions are different and variable, as in many packet-radio networks. A key problem observed in many multi-hop packet radio networks is the large and variable latencies that arise due to the reliable link-layer (for error control) and MAC (for media access) protocols used in these networks. These variable latencies lead to ACK queueing at radio nodes and to highly variable round-trip times at the TCP sender. To ensure that packets currently in transit do not get retransmitted prematurely, TCP takes into account both the smoothed round-trip time estimate and the linear deviation in its timeout algorithm [71]. Thus, timeouts in these networks result in long idle periods on the order of several seconds.

Since protocols like TCP rely on timely ACKs for good performance, the capabilities and characteristics of the reverse path used for ACKs have the potential to seriously affect performance in asymmetric networks [70, 78, 13]. Our contributions to solving this problem include proposing a formal definition of asymmetry with respect to TCP performance in terms of how the characteristics of one direction affect performance in the other, and classifying different types of asymmetry into bandwidth, latency, media-access and loss-rate asymmetries. We then design and implement both end-to-end and link-layer solutions including *ACK filtering, TCP sender adaptation* and *ACK reconstruction* to overcome the adverse effects of asymmetry on TCP performance [13]. We evaluate the effectiveness of these algorithms under uni-directional and Web-like workloads in Hybrid

Networks' bandwidth-asymmetric wireless cable network and in Metricom's Ricochet packet radio network, which displays latency asymmetry.

The fundamental reasons for degraded TCP performance over asymmetric networks are the adverse interactions between TCP data and ACKs. In particular, the reliance on timely ACKs for the sender to smoothly transmit data hampers end-to-end performance. This observation that imperfect and variable ACK feedback degrades TCP performance regardless of the specific network technology motivates our two-pronged solution approach. We first develop techniques to reduce the frequency of ACKs to reduce adverse interactions between data and ACK flows. But minimizing these interactions alone is not enough, as TCP transmissions are clocked by arriving ACKs. Thus, we develop techniques to handle the reduced frequency of ACKs. Together, these techniques comprehensively solve the problems posed by asymmetry.

We use ACK Filtering (AF) to reduce the number of TCP ACKs. AF is a transport-aware link-layer technique that operates at the router feeding the constrained reverse link used by TCP ACKs. Since TCP ACKs are cumulative, later ACKs completely subsume the information contained in earlier ones. When a later ACK is enqueued, all earlier ones belonging to that connection are purged from the queue.

Sender Adaptation (SA) and ACK Reconstruction (AR) are two independent techniques to handle this reduced ACK frequency. The first is an end-to-end scheme where the sender increases its congestion window depending on the number of bytes acknowledged, rather than on the count of the number of ACKs (which is what TCP does today). In addition, the sender also *regulates* the transmission of packets by performing rate control based on its estimate of the current window size and round-trip time. This prevents bursts of packets from being sent out, thereby reducing congestion along the forward path. Together, these techniques reduce the reliance on timely ACK feedback that today's TCP senders rely on.

ACK Reconstruction is a transport-aware link-layer scheme that involves no changes to end-host TCP implementations and accomplishes the same results as SA. This scheme deploys a soft-state agent called the ACK reconstructor at the other end of the constrained reverse link. The reconstructor intersperses TCP ACKs whenever it observes large gaps in the ACK sequence, enabling the sender to obtain consistent and regularly-spaced ACK feedback. It does not spuriously generate any new ACKs, thus maintaining the connection's end-to-end correctness.

1.3.3 Challenge #3: Low Channel Bandwidths

The bit rate commonly available between distant hosts in the Internet today is often limited and time-varying [17]. This is especially true in many wide-area wireless networks where maximum bandwidths (< 500 Kbps) are often orders of magnitude lower than their wired counterparts. In addition, today's dominant TCP application, the World Wide Web, uses short TCP connections because most Web transfers are short. For any connection, the average TCP transmission window size is roughly equal to the product of the available bandwidth and round-trip delay. This is because in the steady state, the sender sends a window's worth of data per round-trip and the ratio of these two is the available bandwidth. When the available bandwidth is low, so is the transmission window. When connections are short, the sender's window obviously does not grow beyond the length of the connection.

Small TCP transmission windows prevent the sender from recovering from losses without incurring expensive timeouts that keep the connection idle for long periods of time. As a result, end-to-end throughputs are significantly lower than the already-low maximum channel bandwidth: TCP transfers in these situations yield throughputs that are only a fraction of what is achievable. Thus, a key challenge is in enhancing TCP's loss recovery algorithms when low available bandwidths lead to small transmission windows.

Based on an extensive analysis of packet traces collected from the Atlanta Olympic Games Web server run by IBM [104], we show that not only are current loss recovery techniques grossly inadequate at preventing sender timeouts, but that proposed enhancements like Selective Acknowledgments (SACK) [92] are not likely to significantly change this because typical Internet transmission windows are not larger than a few segments. Based on the results of our analysis, we develop an *enhanced loss recovery scheme* for TCP that would have prevented about 25% of the timeouts in the particular packet trace, had it been in use, and thereby significantly improved end-to-end performance [15].

In this scheme, the sender enters an *enhanced recovery* phase when windows are small and transmits new probe segments when duplicate ACKs arrive for earlier segments. Upon successful receipt of these new segments, additional duplicate ACKs are generated and reach the sender. When a threshold number of these arrive at the sender, it can accurately deduce that the original segment was indeed lost (and not temporarily reordered in the network) and go ahead and retransmit the segment *without* incurring a timeout. Because packet reordering is a common occurrence on many Internet paths [111], the sender must not retransmit a segment before enough later segments have reached. By sending new segment probes in accordance with the *conservation of packets* principle [59], the sender elicits additional duplicate ACKs from the receiver to distinguish between genuine loss and reordering. Packet conservation ensures that a new segment is sent only after one has been received, since it is triggered by a duplicate ACK; thus, the soundness of TCP's congestion control is preserved.

Using this scheme, we are able to realize a TCP that times out only under persistent congestion—(i) when an entire window is lost and (ii) when a retransmission or a new probe segment is lost—for which a timeout is indeed the correct action. In addition, the enhanced recovery scheme is orthogonal to the information provided by SACKs and can be used in conjunction with SACKs for better overall performance. The resulting protocol improves TCP performance for Web-like workloads on the Internet and substantially improves performance over wide-area wireless networks. Chapter 7 presents the design and implementation details of these enhancements and the results of performance experiments.

1.4 Contributions and Structure of Thesis

In this thesis, we analyze the problems posed by the above challenges and design, implement and evaluate solutions to them that result in significant improvements in end-to-end performance. Because there is tremendous heterogeneity and diversity in wireless access networks, and because there is enormous variation in their characteristics, there is no single panacea that cures all the observed problems. We recognize this and develop a suite of solutions and techniques to combat these problems, that together comprehensively solves these problems. The components of this solution suite include the following protocols and algorithms:

- **The Berkeley Snoop protocol:** This protocol combines a transport-aware link protocol and a link-aware transport protocol, which improves performance in networks composed of wired and single-hop cellular wireless links. The key idea here is the deployment of a soft-state agent at the base station to perform local loss recovery and shield the TCP sender from the vagaries of the wireless link.

- **Explicit Loss Notification (ELN):** ELN is a general framework by which receivers, base stations, or other elements in the network can inform the TCP sender of losses that occur for reasons other than congestion. When combined with algorithms to distinguish congestion losses from corruption, this framework provides a powerful way by which TCP senders can separate congestion control from loss recovery and recover from non-congestion-related losses without invoking congestion control.

- **Asymmetric TCP enhancements:** We present a suite of end-to-end and transport-aware link-layer algorithms to overcome the adverse effects of asymmetry. These involve techniques to reduce the frequency of ACKs and techniques to handle reduced ACK feedback. They improve performance in both packet radio networks that exhibit variable latencies and ACK queueing and in bandwidth-asymmetric networks with low-bandwidth ACK paths.

- **Enhanced TCP loss recovery:** This is an enhancement to TCP's data-driven loss recovery mechanism that improves performance by reducing the number of timeouts when transmission windows are small.

In addition to solving the specific problems in wireless networks discussed above, we also derive a set of general principles and lessons to design transport protocols for heterogeneous networks. These include:

- **Cross-layer protocol optimizations:** The principle of layering [151] has served us well for years as a good way to design network protocols and reason about them. For over a decade, network researchers and practitioners have recognized that a strict layering hierarchy results in an inefficient end-host implementation of a protocol suite, and in response have developed implementation techniques such as Integrated Layer Processing (ILP) [33]. In our work, we develop and refine the idea of *cross-layer protocol optimizations* as a design methodology by which information is exposed across multiple layers of the conventional protocol stack to optimize end-to-end performance. For example, the Snoop protocol deploys a transport-aware link agent, which takes advantage of the reliability mechanisms of the transport layer and does not generate its own independent messages. It also suppresses certain duplicate TCP ACKs from the sender. Similarly, using ELN, the TCP sender is able to distinguish between congestion and corruption-induced losses. Another example is the design of the ACK filter and ACK reconstructor agents to combat the adverse effects of asymmetry; these agents perform TCP-aware connection-specific actions to achieve better end-to-end performance.

We believe that this is a powerful technique that can be generalized to many other situations where protocols are being designed with good network performance in mind. This design principle is orthogonal to ILP, which proposes that multiple layers be coalesced to achieve efficient end-host implementations; in contrast, cross-layer optimizations preserve the principle of layering, but expose certain critical parts of different layers to each other for improved network performance. Chapter 5 discusses this principle in greater detail.

- **Soft-state:** A second general design principle that we use in the design of many of the protocols presented here is the notion of soft-state agents in the network. To enhance performance, agents are often deployed in the network and have state associated with them. However, this state is *soft* [31], i.e., it is not essential for the correct end-to-end functioning of the connection. It is also periodically refreshed by the arrival of new packets and ACKs, which implies that even if it gets corrupted, the correctness of the connection is not compromised. This is in direct contrast to protocols that deploy hard-state agents in the network, such as split-connection methods for wireless TCP. Chapters 4, 5 and 6 discuss this principle in more detail in the context of specific network agents, such as the Snoop protocol, ELN-enhanced TCP and TCP-aware link-layer solutions for asymmetric networks.

  In our work, the design of soft-state agents has been guided by the end-to-end argument in systems design [127]. These agents must be viewed as performance optimizers, are not essential for correctness, and do not compromise end-to-end semantics. Another guiding principle has been the conformance to the IP service model; in particular, we use the fact that packets can be dropped or duplicated in an IP network to great advantage, by suppressing certain types of ACKs, generating local retransmissions, etc.

- **Data-driven loss recovery:** This applies in general to the design of network protocols that attempt to achieve reliability using retransmissions. There are two methods by which reliable transport protocols decide that packets are lost and then retransmit them: using timers (*timer-driven retransmissions*) and using information from later data packets that have reached (*data-driven retransmissions*). Because it is important for transport protocols not to retransmit packets that might still be in transit, retransmission timers are necessarily conservative [150]. Thus, to achieve low recovery latencies and avoid long idle periods that timeouts impose, it is important to make loss recovery as data-driven as possible. We achieve this in the Snoop protocol by taking advantage of the information provided by duplicate TCP ACKs (and SACK, when available) as well as in the enhanced TCP loss recovery for small window connections.

The rest of this thesis is organized as follows. In Chapter 2, we discuss background material in the area of reliable transport protocols and describe related work in improving transport performance over wireless networks. In Chapter 3, we describe the details of our wireless testbed and the experimental networks used in this work. We also discuss the details of our research methodology, which includes analysis, design, simulation, implementation and performance evaluation.

Chapters 4 through 7 form the core of this thesis. Chapter 4 describes the Berkeley Snoop protocol, designed to combat the problems of wireless bit-errors and provide good performance in the presence of wireless corruption.

Chapter 5 argues that exploiting cross-layer protocol optimizations and exchanging information across protocol layers is a powerful technique in the design of protocols to optimize performance. In this chapter, we present the results of several experiments to understand the precise reasons behind the good performance provided by the Snoop protocol and quantitatively compare it to various alternatives.

In Chapter 6, we address the challenges posed by asymmetry, starting with bandwidth asymmetry using Hybrid's wireless cable network as the experimental network. We then focus on understanding the problems posed by the interactions between media-access protocols and TCP in packet radio networks, using Metricom's Ricochet network as the testbed. After discussing these problems, we describe the details of our solutions to them.

In Chapter 7, we describe an enhanced loss recovery algorithm that greatly improves loss performance when TCP windows are small, the common case in the Internet today and in many wide-area wireless networks. We motivate this by a detailed analysis of packet traces collected from the Web server of the Atlanta Olympic Games. After describing the enhancements to TCP, we evaluate them over busy Internet and wireless paths.

Finally, in Chapter 8, we present a summary of our work and contributions. We conclude this chapter and this thesis with a discussion of general lessons learned about adaptive protocol design for heterogeneous networks and outline some directions for future research.