COMP 535 Computer Networks
September 2003

General Information

Instructor: Muthucumaru Maheswaran
Tel: 398-1465
Office: Room 213B, McConnell Engineering Building
Office hours: TBD. Appointments can be made for meetings at other times.
Email: Please use WebCT for all course-related mail. I will try to respond to mails within 24.
Class: MW 16:30-18:00 Macdonald Engineering Building 276
Tutorial: TBA
Prerequisites: CS 310
Class web page: WebCT will be used to disseminate all class material.
TAs: Carlton Davies, Matt Garden, and David Xu.
TA office hours: TBA

“McGill University values academic integrity. Therefore, all students must understand the meaning and consequences of cheating, plagiarism, and other academic offences under the Code of Student Conduct and Disciplinary Procedures (see www.mcgill.ca/integrity for more information).”

Brief Course Description

This is a senior undergraduate/first-year graduate course in computer networks. We will examine computer networks within the context of the Internet. It will build on prior knowledge in operating systems, basic algorithms, and C programming. We will study the fundamental principles, elements, and protocols of computer networks. We will investigate how the different protocols work, why they work that way, and their performance trade-offs. Using this knowledge, we will try to examine the way applications are deployed on the Internet and their performance trade-offs. In particular, we will try to examine some strategies that are commonly used to accelerate application-level performance in the context of the operation of the Internet.

By the end of the course, you should be able to: (i) explain the operation of a range computer networking applications such as email, web, and peer-to-peer file-sharing; (ii) relate the architecture of the Internet to the underlying design principles; (iii) illustrate the operation of common routing protocols, queuing mechanisms, and congestion control mechanisms; (iv) develop elements of a network such as gateways and routers that conform to IETF standards with acceptable levels of simplification and (v) explain the performance of a given set of routing protocols, queuing mechanisms, and congestion control mechanisms on an example network.
Course Content

Concept map

Applications

Performance

Networks

Implementation

Algorithms/Principles

- micro-level performance
- macro-level performance
- benchmarks
- performance implications

- design principles
- routing algorithms
- congestion control
- queuing algorithms

Syllabus and detailed learning outcomes

*Items marked with an asterisk (*) may be omitted depending on available time. The learning outcomes mentioned here are tentative. The actual learning outcomes may vary slightly from these and can be found on the first slide of the corresponding course notes.*

Module I: Foundation [Kurose, Ch. 1]

1.1 Application requirements
1. Introduce the requirements placed on a computer network in the context of the Internet by applications.
2. Realize what applications require and what can be cost-effectively provided by the underlying network.

*By the end of this section you should be able to:*
- List the major application requirements.
- Explain the anticipated importance of considering the requirements in designing and analyzing networks.

1.2 Network architecture and elements
1. Introduce the basic elements and terminology.
2. What is the Internet architecture?
3. What are the major design principles of the Internet?
4. Protocols and layered architecture

*By the end of this section you should be able to:*
- List the major elements of the Internet.
- Explain the Internet architecture.
• Relate the Internet design decisions with the architecture.

1.3 Performance: Delay, loss, and bandwidth
1. Introduce different performance factors such as delay, loss, and bandwidth.
2. Delay X bandwidth product
3. Application performance requirements
By the end of this section you should be able to:
• Explain the causes of delay and loss in networks.
• Describe the importance of the delay and bandwidth product.
• Correlate the application performance requirements with network characteristics.

1.4 Brief history: How it all started?
1. Discuss the evolution of the networking technology.
2. List the major developments and a look forward.
By the end of this section you should be able to:
• List the major developments in networking

Module II: Application Layer [Kurose, Ch. 2]

2.1 Name service
1. Domain hierarchy
2. Name servers
3. Name resolution process
By the end of this section you should be able to:
• Explain the issues involved in designing scalable name services
• Explain the operation of DNS
• Assess alternate name service architectures

2.2 Traditional applications
1. Electronic Mail (SMTP, MIME, IMAP)
2. World Wide Web (HTTP)
3. File transfer (FTP)
By the end of this section you should be able to:
• Identify the requirements of traditional applications
• Explain how traditional applications interact with network services

2.3 Multimedia applications
1. Examples of multimedia networking applications
2. How should they be supported?
3. Real-time transport protocol
4. Session control and call protocol
By the end of this section you should be able to:
• Identify the requirements of multimedia applications
• Explain how multimedia applications interact with network services

2.4 Overlay networks
1. Routing overlays
2. Peer-to-peer networks
3. Content distribution networks
By the end of this section you should be able to:
• Explain the reason for constructing overlay networks
• Identify the key techniques for constructing overlays
• Identify the key performance parameters in overlays
Module III: Network Layer and Routing [Kurose, Ch. 4]

3.1 Internetworking
1. What is an Internetwork, service model
2. IP addressing
3. Datagram forwarding in IP and address resolution
4. Host configuration (DHCP) and error reporting (ICMP)

By the end of this section you should be able to:
- Identify the main features of the Internetwork service model
- Explain the features of IP addressing: limitations and extensions
- Explain the forwarding process in the Internetwork

3.2 Routing
1. Autonomous systems
2. Intra-AS routing: RIP, OSPF
3. Inter-AS routing: BGP

By the end of this section you should be able to:
- Explain the concept of Autonomous systems and how they could be used by Internetworking applications
- Describe RIP, OSPF
- Describe BGP
- Formulate scenarios that demonstrate the performance trade-offs among the different routing mechanisms

3.3 Router design
1. Overall architecture of traditional IP routers
2. Beyond best effort, high speed routers

By the end of this section you should be able to:
- Describe major components of a traditional IP router
- Identify the performance factors of a router
- Discuss the new demands imposed on routers by beyond best effort traffic

3.4 Multicast routing
1. Internet multicast abstraction and multicast groups
2. IGMP
3. Multicast routing algorithms: distance-vector multicast, protocol independent multicast

By the end of this section you should be able to:
- Describe multicast abstractions
- Explain the operation of different multicast routing algorithms

3.5 Mobility and routing
1. Mobility consideration in network-layer design
2. Mobility management
3. Mobile IP

By the end of this section you should be able to:
- Identify the major issues presented by mobility
- Describe the approaches proposed to handle mobility
- Explain mobile IP and its limitations

Module IV: Link Layer and Switching [Kurose, Ch. 5]

4.1 Hardware building blocks
1. Nodes
2. Links
3. Network adaptors: components, view from memory, bottleneck

*By the end of this section you should be able to:*
- Identify the physical components involved in data communications
- Explain possible performance bottlenecks at the physical system level

### 4.2 Encoding, framing, and error-detection

1. NRZ, NRZI
2. Byte-, bit-oriented framing protocols and clock-based framing protocols
3. Parity checks, checksumming methods, CRC

*By the end of this section you should be able to:*
- Explain the above encoding, framing, and error-detection mechanism

### 4.3 Multiple access protocols

1. Channel partitioning protocols
2. Random access protocols, taking-turn protocols
3. Local area networks (LANs)

*By the end of this section you should be able to:*
- Explain the operation of multiple access protocols

### 4.4 Ethernet, hubs, bridges, switches, and VLANs

1. Ethernet basics
2. Ethernet’s multiple access protocol
3. Hubs, bridges, switches
4. Spanning tree algorithm
5. Virtual LANs

*By the end of this section you should be able to:*
- Explain the operation of Ethernet
- Explain the operation of hubs, bridges, and switches
- Explain the operation of VLANs
- Describe the differences among hubs, bridges, switches, and VLANs

### 4.5 Wireless Links

1. Wireless LANs: 802.11b: Physical properties
2. Collision avoidance, distribution system

*By the end of this section you should be able to:*
- Explain the operation of wireless LANs
- Identify the major differences between wireless and wired LANs

### 4.5 ATM and MPLS

1. Principle characteristics of ATM
2. ATM physical layer
3. ATM Layer
4. ATM adaptation layer
5. IP over ATM
6. MPLS

*By the end of this section you should be able to:*
- Describe the salient characteristics of ATM
- Explain the operation of ATM
- Packet switching versus circuit switching
- MPLS as a hybrid of packet and circuit switching
- Advantages of MPLS over traditional IP packet forward
Module V: Transport Layer and Resource Allocation

5.1 Connectionless transport: UDP
1. Properties of connectionless transport
2. UDP segment structure, checksum
*By the end of this section you should be able to:*
   - Identify conditions that require connectionless transport
   - Explain UDP operation

5.2 Principles of reliable data transfer
1. Reliable data transfer protocols
2. Sliding window
3. Alternate design choices
*By the end of this section you should be able to:*
   - Explain the issues involved in reliable data transfer
   - Trade-offs among the different reliable data transfer approaches

5.3 Connection oriented transport: TCP
1. TCP connection, segment structure
2. Round trip time estimation and timeout
3. Reliable data transfer
4. Flow control
5. Connection management
*By the end of this section you should be able to:*
   - Explain the operation of TCP
   - Identify key parameters that can impact the performance of TCP connections

5.4 Resource allocation and queuing
1. Resource allocation in networks: network model, evaluation criteria
2. Queuing as a resource allocation process
3. FIFO and fair queuing
4. Evaluation criteria
*By the end of this section you should be able to:*
   - Identify the fundamental requirements of resource allocation in networks
   - Interpret queuing as a form resource allocation
   - Explain the differences between different forms of queuing
   - Identify conditions that are best suited for different forms of queuing

5.5 TCP Congestion control and avoidance mechanisms
1. TCP congestion control: additive increase/multiplicative decrease
2. Slow start, fast retransmit and fast recovery
3. TCP delay modeling
4. Congestion avoidance schemes: DECbit, RED, Source-based congestion avoidance
*By the end of this section you should be able to:*
   - Explain the major components of congestion control schemes
   - Explain end system based approaches
   - Explain router assisted approaches
   - Identify the trade-offs among the different approaches

5.6 Quality of service
1. Scheduling and policing mechanisms
2. Leaky bucket regulator
3. Integrated services
4. RSVP
5. Differentiated services
6. Limitations and trade-offs of integrated and differentiated services

By the end of this section you should be able to:
- Explain scheduling and policing
- Describe the major concepts behind integrated services
- Describe the major concepts behind differentiated services
- Analyze the limitations of both approaches
- Suggest the best approach for a given scenario

Module VI: Network Security [Kurose, Ch. 7]

6.1 Cryptographic algorithms
1. Requirements
2. Secret key encryption (DES)
3. Public key encryption (RSA)
4. Message digest algorithms
5. Implementation and performance

By the end of this section you should be able to:
- Explain the operation of secret and public key encryption mechanisms

6.2 Security mechanisms
1. Authentication protocols
2. Message integrity protocols
3. Public key distribution

By the end of this section you should be able to:
- Explain authentication protocols
- Explain message integrity protocols
- Explain public key distribution

6.3 Access control: Firewalls
1. Firewall as access control mechanisms
2. Firewall related issues
3. Distributed firewalls

By the end of this section you should be able to:
- Explain access control process through firewalls

6.4 Attacks and countermeasures*
1. Mapping
2. Packet sniffing
3. Spoofing
4. Denial-of-service and distributed DoS
5. Hijacking

By the end of this section you should be able to:
- Describe different forms of attacks
- Explain mechanisms that can detect some forms of attacks
- Explain prevention or countermeasures for different attacks

Module VII: Network Performance*

7.1 Measurement: techniques and tools
1. Active measurement
2. Passive measurement
3. Bandwidth and delay measurement
4. Iperf tool
5. Cprobe: congestion estimation tool  
6. Web traffic measurement  
*By the end of this section you should be able to:*  
- Explain different measurement strategies  
- Explain the operation of the measurement tools

### 7.2 Workload characterization and benchmarks

1. Web workload characterization  
2. HTTP message characteristics  
3. Applying workload models  
4. Web benchmarks  
*By the end of this section you should be able to:*  
- Explain workload characterization  
- Describe the need for workload characterization

### 7.3 Performance improvement strategies

1. Exploiting and improving locality: caching, improving memory utilization, edge delivery  
2. Load balancing: reducing performance bottlenecks  
3. Data staging  
*By the end of this section you should be able to:*  
- Explain the general principles behind performance improvement strategies  
- Describe how these principles are implemented in specific mechanisms

## Instructional Method

The course will consist of three hours of instructor led classes per week together with a maximum of one hour of tutorial taken by the TAs. The class time will be devoted to the presentation and development of new concepts and the application of these concepts to examples and problems, while the tutorials will discuss solutions to the programming projects and written assignments. The primary focus of the tutorials is to provide sufficient “how-to” knowledge through the discussion of the assignments to help in the development of the programming project series.

To make the class time most effective, students are strongly encouraged to finish the reading assignments before the corresponding classes. WebCT quizzes will be available (for selected sections) to help the students check their understanding of the reading material. These quizzes will not be graded and should be used by the students as a study guide.

Students are strongly encouraged to use the WebCT discussion groups to talk about the programming and written assignments. These discussion groups will be monitored by the instructor and TAs for providing the necessary answers.

## Course Materials

### Textbooks

Supplementary or Reference texts:


Assignments and Evaluation

<table>
<thead>
<tr>
<th>Activity</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ungraded written assignments (3 or 4)</td>
<td>-</td>
</tr>
<tr>
<td>Written assignments (2)</td>
<td>10%</td>
</tr>
<tr>
<td>Programming assignments (3)</td>
<td>30%</td>
</tr>
<tr>
<td>Final project</td>
<td>20%</td>
</tr>
<tr>
<td>Extra-credit on final project</td>
<td>10%</td>
</tr>
<tr>
<td>Midterm</td>
<td>10%</td>
</tr>
<tr>
<td>Final</td>
<td>30%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>110%</strong></td>
</tr>
</tbody>
</table>

1. **Ungraded written assignments**: The purpose of these assignments is to point out the relevant conceptual material for the programming assignments. Also, these assignments will form the basis for the tutorial discussions. The questions in these assignments will be of the descriptive form.

2. **Written assignments**: The purpose of these assignments is to cover material in the course syllabus that is not examined through the programming assignments. Congestion control algorithm is an example of such a topic. An assignment is expected to have 3 to 4 questions of short essay type.

3. **Programming assignments**: These assignments attack a sub problem of the final project topic. Once all the programming assignments are done, a substantial portion of
the final project will be covered. By giving the programming assignments and assessing the work in stages we can avoid the problems of late or no start. It is another way to ensure proper scheduling of your time!

4. **Final project:** The final project will focus on a user-level virtual Internet. Specifications will be given and grading scheme will be announced when the project is assigned.