CS 464/564: Networks Programming [and Distributed Algorithms]

Stefan D. Bruda

Fall 2023

Network Programming

- Coordinates:
  - **Course Web page:**

    http://cs.ubishops.ca/home/cs464
    http://cs.ubishops.ca/home/cs564
  - Instructor: Stefan Bruda (stefan@bruda.ca, Johnson 114b, ext. 2374).
  - Email rules.
  - Office hours?
- Textbook:
  - Douglas E. Comer and David L. Stevens, Internetworking with TCP/IP, Vol. III: Client-Server Programming and Applications, Linux/Posix Sockets Version, Prentice Hall, 2001.
  - Course pack available at Doolittle's Co-Op (printed and electronic)
- **Must have a working account on linux.ubishops.ca or obtain one asap**

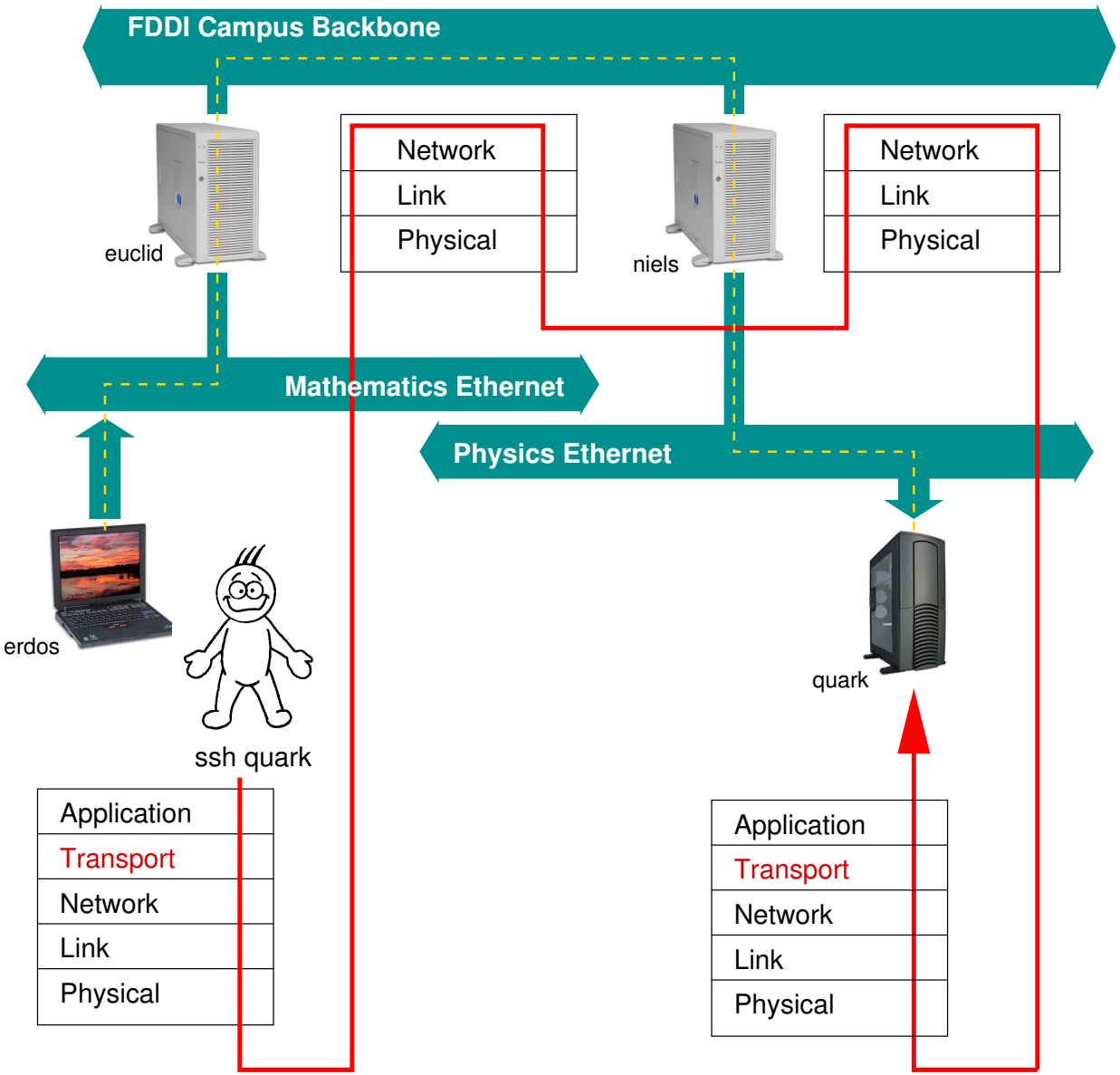
What is it all about

- A bit of computer networks, a lot of programming
- Or how one can use TCP/IP to communicate across internets
  - We learn how to build distributed applications (using the TCP and UDP protocols, and the client-server paradigm)
  - We will then see what algorithmic features allow us to do so (the IP protocol)
- We use
  - C/C++
  - The TCP/IP protocols
  - The POSIX (Portable Operating System Interface for uniX) standard
  - Berkeley sockets
  - Linux (programming details are largely unchanged in other flavours of Unix, principles are the same for any OS)

Networks

- Networks are all about transmitting messages between individuals
- As old as mankind
- Consider Stone Age: *A* wants to invite *B* to his place, uses a drum, but *B* is too far away to hear. Then *A* can
  1. get a bigger drum
  2. walk to *B*'s place
  3. ask *C* (who lives halfway) to forward the invitation $\rightarrow$ networking!
- Of course, we now use computers, fiber optics, satellites, etc. and we send each other emails

NETWORKS (CONT'D)



Application	C/C++	CS 464
Transport	TCP	CS 464
Network	IP	CS 464
Link	Ethernet	CS 464
Physical	Twisted pair	

Protocols

- Networking is all about communication
- In order for two programs to understand each other, they must use a common language i.e., a protocol
  - standard if they are part of the “official” protocol suite, or
  - nonstandard
- Standard protocols exist for many services including remote login, file transfer, and sending and receiving email
- Distributed applications involve communication between a client and a server through some protocol understood by both

Where do we go from here

- We basically learn how to design, implement, and optimize distributed (i.e., client-server) programs
  - Once this is understood, we will go a bit deeper and talk about the underlying algorithms
- We thus discuss implementations of application protocols
- In doing this, we must introduce and then use concurrency
- During the class, we work in C, or C++ with C constructs when the API being used requires it