

# Distributed Systems

600.437

Introduction

Department of Computer Science  
The Johns Hopkins University

Yair Amir

Fall 16 / Lecture 1

1

## Course Information



- Lecture/Tutorial: Tuesday 3pm – 4:15pm, Shaffer 300
  - Lecture/Tutorial: Thursday 3pm – 4:15pm, Shaffer 300
  - Instructor: Yair Amir
    - Office hours: Malone 209/207 Thursday 1pm – 2pm
  - TA : Tom Tantillo
    - Office hours: Malone 209/207 Monday 1pm – 2pm
  - Special help: Emily Wagner, Amy Babay
    - DSN lab – Malone 207
  - E-mail contact to all of us: [cs437-help@dsn.jhu.edu](mailto:cs437-help@dsn.jhu.edu)
  - Course mailing list: [www.dsn.jhu.edu/mailman/listinfo/cs437-2016](http://www.dsn.jhu.edu/mailman/listinfo/cs437-2016)
  - Course web page: [www.dsn.jhu.edu/courses/cs437/](http://www.dsn.jhu.edu/courses/cs437/)
- E-mail is best. Next, come to office or lab.

Yair Amir

Fall 16 / Lecture 1

2

## This Week and Next

- Thursday
  - Getting to know each other
  - Introduction to the course
- Next Tuesday
  - Introduction to the course (cont)
  - Basic network protocols
- Next Thursday
  - Tutorial, first practical exercise
- Goal
  - By the end of next week, you have the information you need to decide if you want to take the course

## Lets go around the room

- Name
- Department
- Degree (BS, BS/MS, MS, PhD)
- Year in degree (1, 2, 3, 4)
- Programming experience
  - (C, C++, Java, etc. / school - outside of school)
- Other relevant experience (networking, systems)
- ☺ Why are you here ☺  
(what do you expect from the course ?)

## Grading Policy

- Two written assignments
- Three programming assignments
- One final project, presentation date: **12/21/2016**
- Attendance!
- No exam 😊

$$18\% + 42\% + 30\% + 10\% = 100\%$$

The difference between 337 and 437 :

Depth of the assignments and the project

There is no difference in grading

- Ethics code: standard CS code [www.cs.jhu.edu](http://www.cs.jhu.edu)
- Zero tolerance for ethics problems
  - We invest a lot and expect a lot in return

Programming language: C or C++

Testing environment: the undergrad lab - ugrad1-20

Need to get an account!!

Yair Amir

Fall 16 / Lecture 1

5

## Course Overview

### Lecture 1

Recommended Book:

Guide to Reliable Distributed Systems

Kenneth P. Birman

Springer 2012

ISBN: 978-1447124153

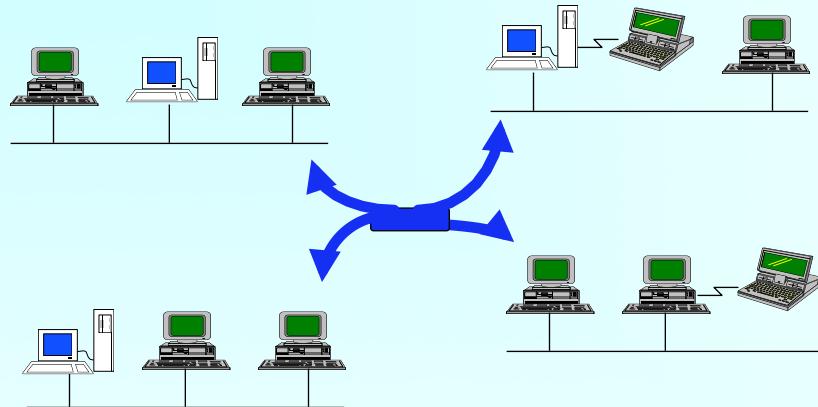
<http://www.dsn.jhu.edu/courses/cs437/>

Yair Amir

Fall 16 / Lecture 1

6

# A Distributed System



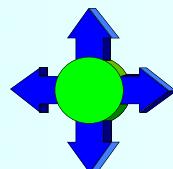
Yair Amir

Fall 16 / Lecture 1

7

## Why Distribute?

- Bridge geographic distances.
- Improve performance.
- Improve availability.
- Maintain autonomy.
- Lower the cost.
- Specialization (e.g GPUs)
- Allow for interaction.



Yair Amir

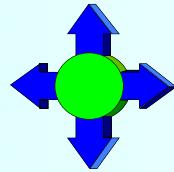
Fall 16 / Lecture 1

8



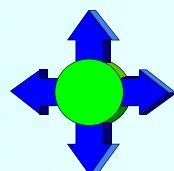
## Why Distribute?

- Bridge geographic distances.
- Improve performance.
- Improve availability.
- Maintain autonomy.
- Lower the cost.
- Specialization (e.g GPUs)
- Allow for interaction.



## Why Distribute?

- Bridge geographic distances.
- Improve performance.
- Improve availability.
- Maintain autonomy.
- Lower the cost.
- Specialization (e.g GPUs)
- Allow for interaction.



**Software solutions are needed!**

# Shaffer 300

## Course Content



- Introduction. [Sep 1](#)
- Course intro + Basic Network Protocols [Sep 6](#)
- Synchronous models in distributed environments  
[Sep 13-15](#)
- Messaging Systems (benign model) [Sep 20-22-27-29](#)
  - Multicast, group communication, overlay networks
- Asynchronous models in distributed environments. [Oct 4 – Oct 6](#)
- Consistent State (benign model) [Oct 11-13-18-25](#)
  - Distributed transactions, replication, Paxos, Congruity

Yair Amir

Fall 16 / Lecture 1

11

# Shaffer 300

## Course Content



- Intrusion-Tolerant Replication [Oct 27 – Nov 1 – 3](#)
  - BFT, Prime, Proactive Recovery, applications of intrusion tolerant replication
- Intrusion-Tolerant Messaging [Nov 8 – 10](#)
- Final project discussions [Nov 15 – 17](#)
- Large-scale Data Stores & Probabilistic Protocols [Nov 29 – Dec 1](#)
- Knowledge in Distributed Systems (game) [Dec 6](#)
- Course summary [Dec 8](#)

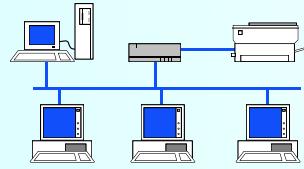
Yair Amir

Fall 16 / Lecture 1

12

# Shaffer 300

## Tutorial



- Presenting and discussing exercises
- Communication using Unix sockets:
  - Reliable point-to-point communication (TCP/IP)
  - Non-reliable p-to-p communication (UDP/IP)
  - Non-reliable Broadcast, Multicast (UDP/IP)
- The Spread toolkit
- The Spines overlay messaging system

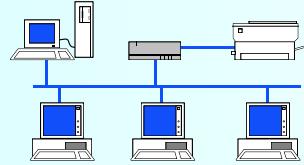
Yair Amir

Fall 16 / Lecture 1

13

# Shaffer 300

## Tutorial



- Thursday **Sep 8** – point to point tutorial
- First practical exercise – Thursday **Sep 8**
- Submission of first exercise: Monday **Sep 19**, 11pm
- Tuesday **Sep 27** – Multicast tutorial
- Second practical exercise –Tuesday **Sep 27**

Yair Amir

Fall 16 / Lecture 1

14

## Synchronous and asynchronous models.

Some Theory :)



Yair Amir

Fall 16 / Lecture 1

15

## Messaging Systems: Multicast Protocols



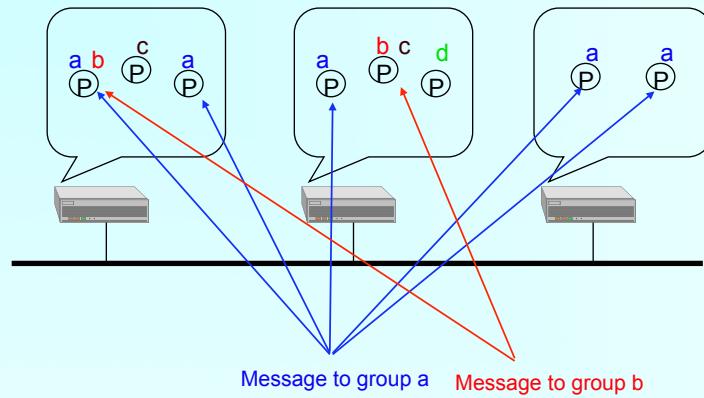
- Reliability
- Ordering
- Membership
- Network span
- Routing
- Quality of Service

Yair Amir

Fall 16 / Lecture 1

16

## Messaging Systems: Group Communication



Yair Amir

Fall 16 / Lecture 1

17

## Spread: A Group Communication Toolkit

Spread provides:

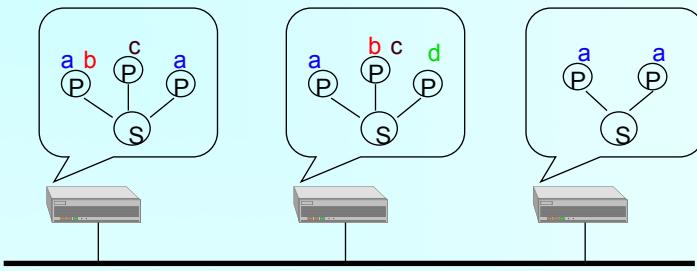
- Process groups across local and wide area networks (Internet).
- Consistent service semantics:
  - Various levels of reliability
  - Various levels of message ordering
  - Membership services.
- High performance.

Yair Amir

Fall 16 / Lecture 1

18

## Process groups in Spread



- One Spread daemon in each machine
- Multiple destination groups per message

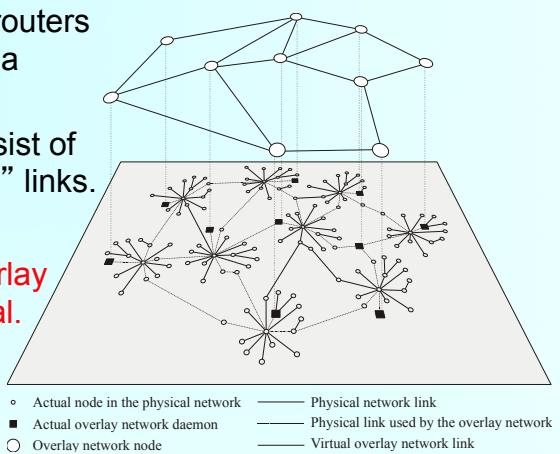
Yair Amir

Fall 16 / Lecture 1

19

## Messaging Systems: Overlay Networks

- Application-level routers working on top of a physical network.
- Overlay links consist of multiple “physical” links.
- **Incurs overhead.**
- **Placement of overlay routers not optimal.**
- **Flexible use of peer-protocols.**
- **Provides added value.**

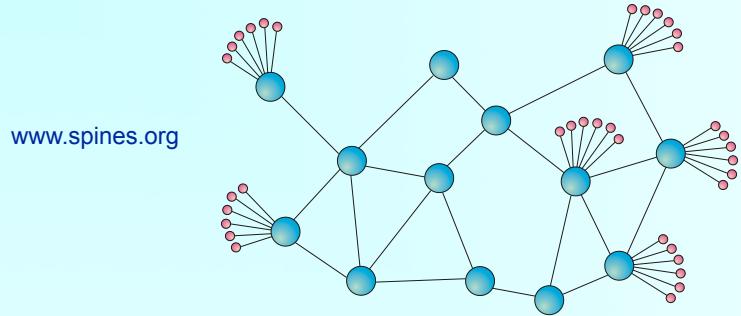


Yair Amir

Fall 16 / Lecture 1

20

## The Spines Overlay Messaging System



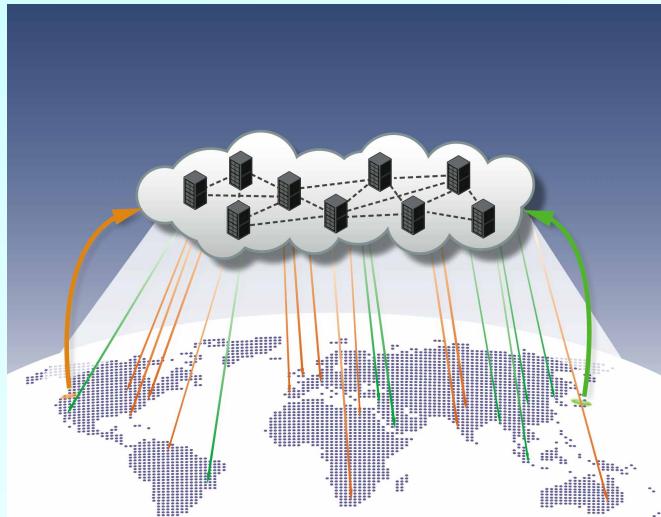
- Daemons create an overlay network on the fly
- Clients are identified by logical addresses identical to Internet addresses (IP address and a port ID)
- Clients feel they work with standard Internet protocols
- Protocols designed to support up to 1000 daemons (locations), each daemon can handle up to about 1000 clients

Yair Amir

Fall 16 / Lecture 1

21

## A Global Overlay in Action

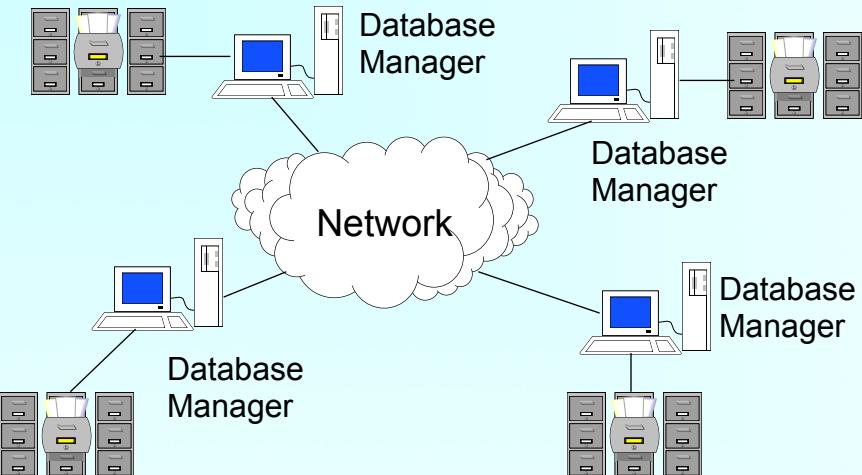


Yair Amir

Fall 16 / Lecture 1

22

## Consistent State: A Distributed Database



Yair Amir

Fall 16 / Lecture 1

23

## Distributed Transactions

When data is spread over several database servers, there should be a way to coordinate transactions so that they will be:

- Atomic - either all effects take place, or none.
- Consistent - correct.
- Isolated - as if there was one serial database.
- Durable - effects are not lost.

**Atomic Commit Protocols** correctly coordinate distributed transactions.

Yair Amir

Fall 16 / Lecture 1

24

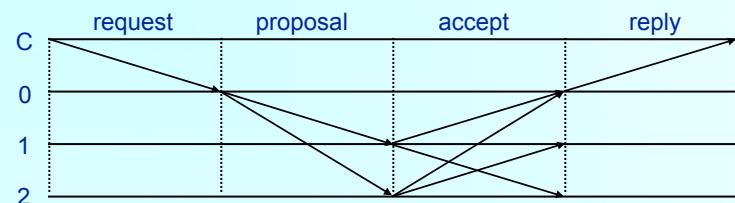
## Consistent State: Replication

### Considerations:

- Improve availability
- Improve performance for queries (higher throughput, lower latency).
- Cost
- Soft state / Persistent state
- Update rate / State size.
- Dynamic instantiation and consolidation.

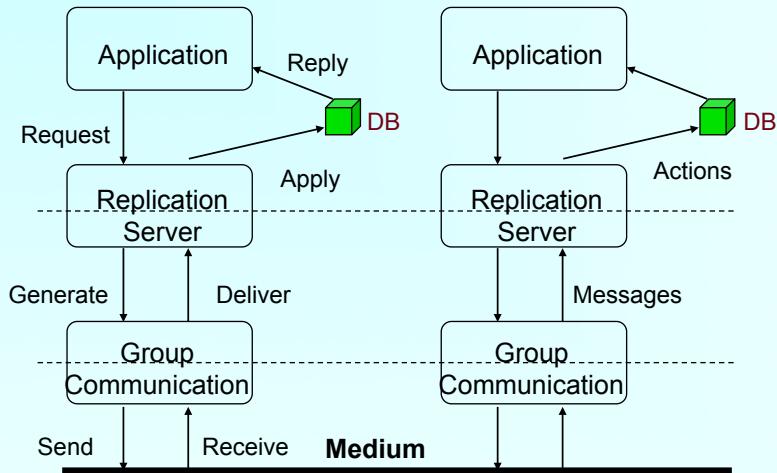
Fault model? Guarantees? Performance?

## Paxos Replication



- Lamport, 1998, 2001
- A very resilient protocol. Only a majority of participants are required to make progress.
- All participants are trusted.
- Works well on unstable networks.

## Congruity Replication Replication over Group Communication

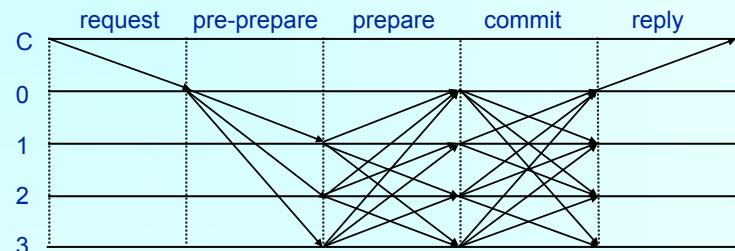


Yair Amir

Fall 16 / Lecture 1

27

## Intrusion Tolerant Replication BFT



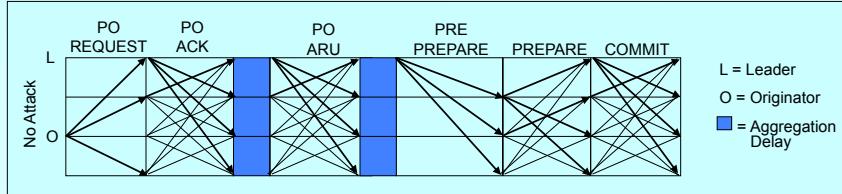
- **Byzantine Fault Tolerance [Castro and Liskov, 99]**
- 2/3 total servers +1 are required to make progress.
- Three rounds of message exchanges.
- Works even if up to (but less than) 1/3 of servers are compromised!

Yair Amir

Fall 16 / Lecture 1

28

## Intrusion Tolerant Replication Prime



- Performance Guarantees under Attack [Amir, Coan, Kirsch, Lane, 2008]
- Works even if up to (but less than) 1/3 of servers are compromised!
- **Bounded-Delay:** There exists a time after which the update latency for any update initiated by a stable server is upper-bounded.

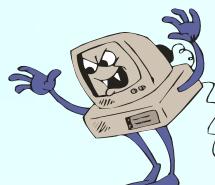
Yair Amir

Fall 16 / Lecture 1

29

## Intrusion Tolerant Replication Application: Critical Infrastructure

- Supervisory Control and Data Acquisition (SCADA) systems form the backbone of critical infrastructure services
- Today's systems tolerate "benign" faults but are not built to survive **intrusion attacks**
  - Assumption of private network no longer holds - systems move to the Internet
  - SCADA is increasingly a target for attackers
- Intrusion tolerant replication serves as an important building block for intrusion-tolerant SCADA



Yair Amir

Fall 16 / Lecture 1

30

## Intrusion Tolerant Messaging

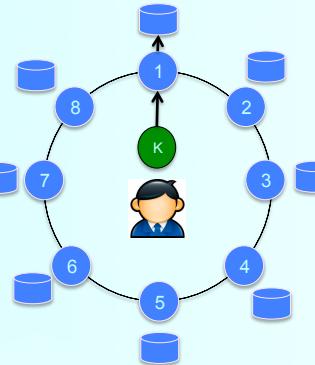
- Distributed systems can span wide area locations and rely on geographically-distributed networks to communicate
- Underlying network must be **intrusion-tolerant** to protect against network intrusions
- Normal routing algorithms are insufficient
  - Nodes are trusted and routing updates determine path calculations
  - **Compromised** nodes can disrupt the routing protocol by lying in their updates

## Intrusion Tolerant Messaging

- Any node can be a source
- Any node can be **compromised**
- **Compromised** nodes may be undetectable
  - Cannot prefer one node's traffic over another's
  - Risk of favoring **compromised** nodes and starving **correct** sources traffic
- Ensure **fairness** and guarantee **performance** for flows even while under attack
- Requires cryptographic mechanisms for authentication and integrity

## Large-scale Data Stores

- The promise:
  - Data stores can be built to scale horizontally (by adding more machines)
- Advantages:
  - Can run on hundreds of machines
  - Can scale up and down elastically as needs change
- Challenges:
  - Stability – the system needs to handle failures
  - Consistency – transactions can require coordination across many machines



Yair Amir

Fall 16 / Lecture 1

33

## Basic Communication Protocols

### Lecture 1

Further reading: Data Networks,  
D. Bertsekas and R. Gallager,  
Prentice-Hall.

Internetworking with TCP/IP  
Volume I  
D. E. Comer

Yair Amir

Fall 16 / Lecture 1

34

# Automatic Repeat reQuest (ARQ) Protocols.

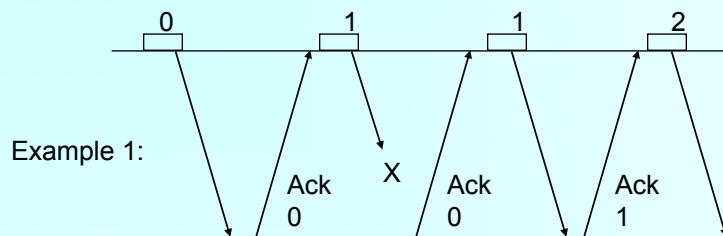
Causes for message omission:

- Buffer spill.
- Error detection in a packet.

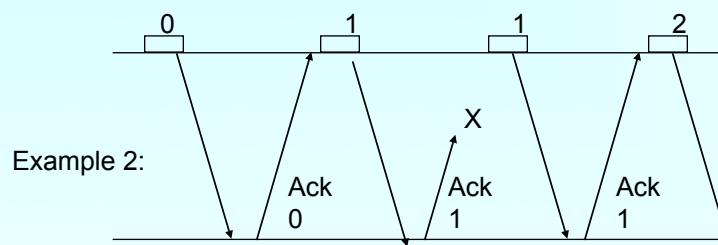
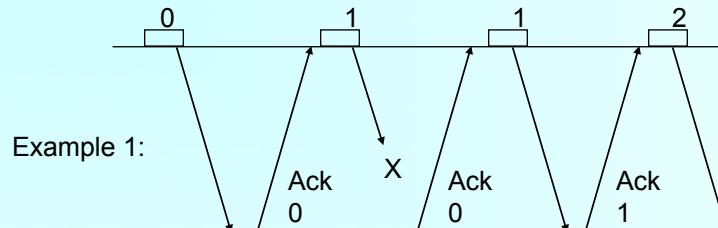
ARQ protocols:

- Send & Wait.
- Arpanet.
- Go back  $n$ .
- Selective Repeat.

## Send & Wait ARQ



## Send & Wait ARQ (cont.)

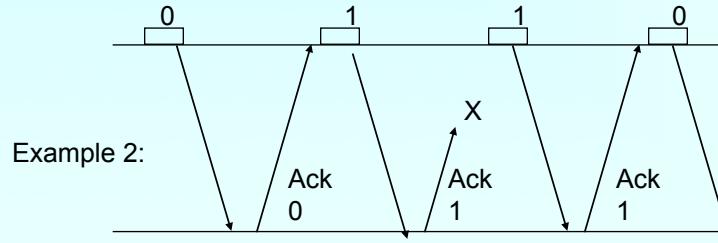
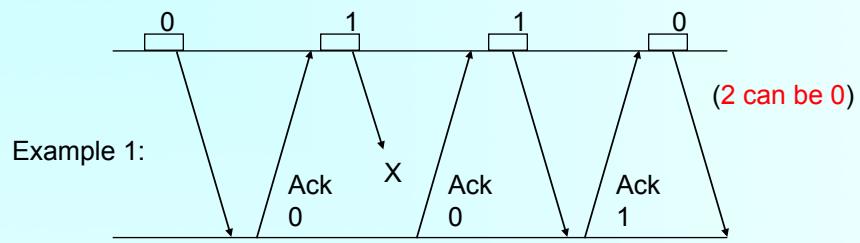


Yair Amir

Fall 16 / Lecture 1

37

## Send & Wait ARQ (cont.)

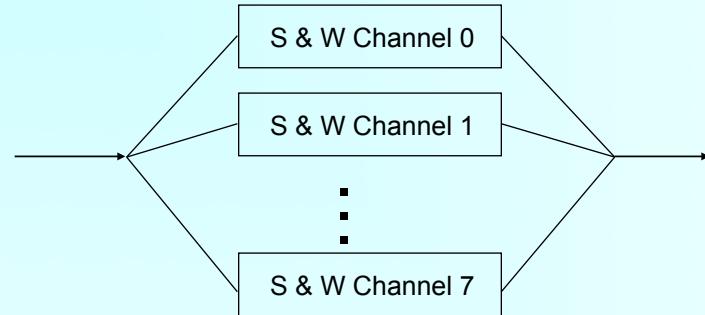


Yair Amir

Fall 16 / Lecture 1

38

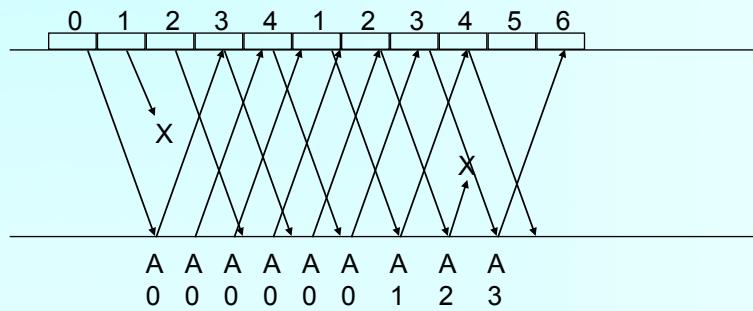
## Arpanet ARQ



- Better line utilization than S & W.
- Unlimited memory required in theory.

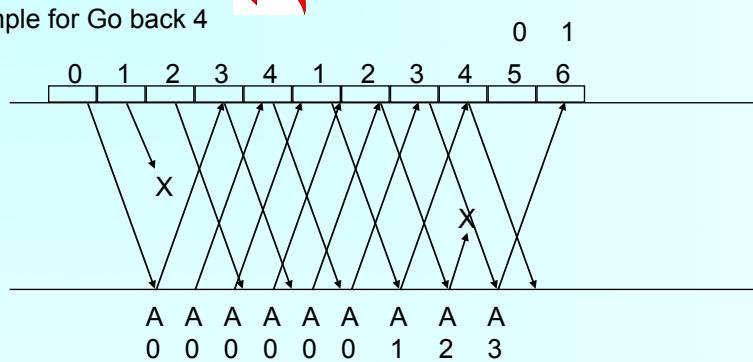
## Go back n ARQ

Example for Go back 4



## Go back n ARQ

Example for Go back 4



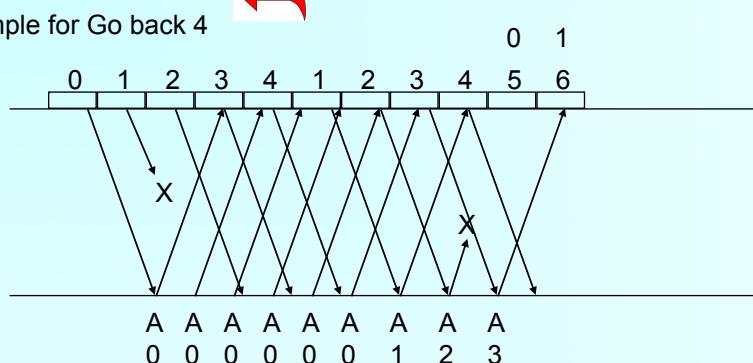
Yair Amir

Fall 16 / Lecture 1

41

## Go back n ARQ (cont.)

Example for Go back 4



- Good utilization
- limited memory required (one packet only).
- Full window is retransmitted in case of (one) error.

Yair Amir

Fall 16 / Lecture 1

42

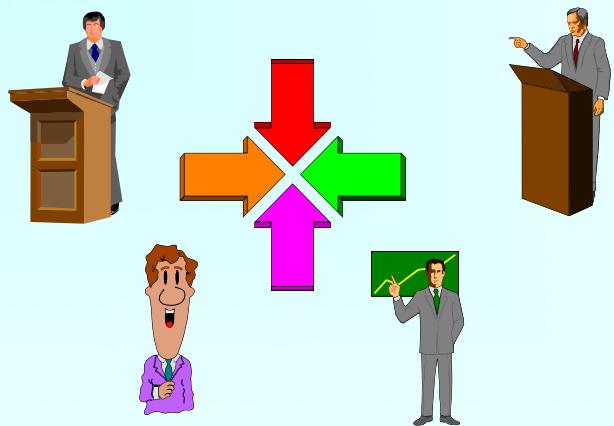
## Selective Repeat ARQ

- Sliding window technique (as Go back  $n$ ).
- Specifically indicating which packet is missing.
- Combines nacks and cumulative acks.
  - Acks acknowledge all messages with index of up to and including the ack value.
  - Nacks (negative acknowledgements) specifically request the messages with the indices in the nacks' values.
- Limited memory required (a full window).

## Question: What if there is no feedback?

- A word about forward error correction (FEC), Internet loss patterns, etc.

# Medium Access Control for Multi-access Communication.

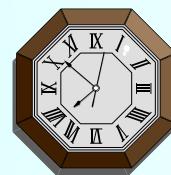


Yair Amir

Fall 16 / Lecture 1

45

# Time Division Multiplexing (TDM)



Slot 1   Slot 2   Slot 3   Slot 4   Slot 1

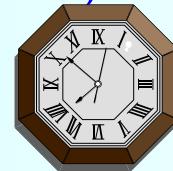
- The best utilization if everyone always has something to send.
- Wastes time if this is not the case.
- Slots can be unevenly assigned.

Yair Amir

Fall 16 / Lecture 1

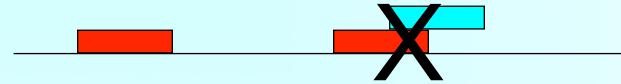
46

## Slotted Aloha (Theoretical)



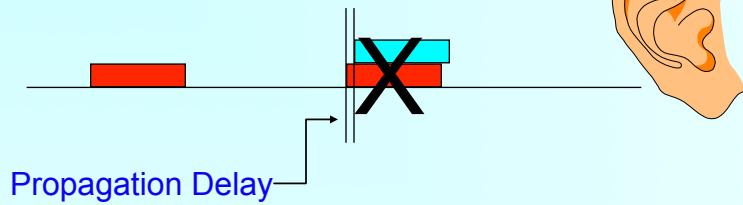
- Send at the next slot.
- If collision occurs - pick a random waiting time and send again at the next slot.
  - Breaks.
  - Maximal utilization is 0.36 (but much less for a desired behavior).

## Aloha



- Send immediately.
- If collision occurs - pick a random waiting time and send again at that time.
  - Breaks.
  - Maximal utilization is 0.18 (but much less for a desired behavior).

## Carrier Sense Multiple Access (CSMA)



- Listen to the line. Send if line is free.
- If collision occurs - pick a random waiting time and **try** again at that time.

## CSMA/CD

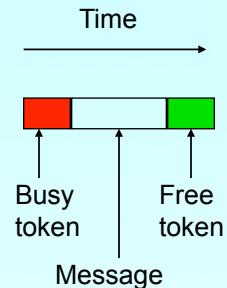
Carrier Sense, Multiple Access with Collision Detection.

Points to clarify:

- Propagation delay.
- X persistent CSMA
- Splitting algorithm for collisions

Ethernet = Persistent CSMA/CD with binary exponential backoff.

## Token Ring



- Token loss.
- Node crash.

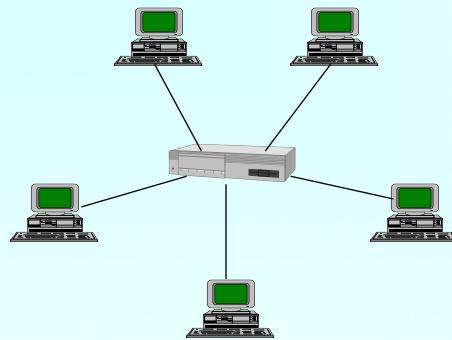
Yair Amir

Fall 16 / Lecture 1

51

## A Star Configuration

Can be used to mimic a bus configuration.  
e.g. for Ethernet, Fast Ethernet, 1Gig Ethernet,  
10Gig Ethernet, or Token Ring.



Yair Amir

Fall 16 / Lecture 1

52

# Routing

- Distance vector routing
- Link state routing
- Inter-network routing

## Distance Vector Routing

- Each router knows the id of every other router in the network.
- Each router maintains a vector with an entry for every destination that contains:
  - The cost to reach the destination from this router.
  - The first link that is on that least-cost path.
- Each router periodically sends its vector to its direct neighbors.
- Upon receiving a vector, a router updates the local vector based on the direct link's cost and the received vector.

## Link State Routing

- Each router knows the id of every other router in the network.
- Each router maintains a topology map of the whole network.
- Each router periodically floods its direct links state (with its direct connectivity information).
- Upon receiving a vector, a router updates the local topology map and re-calculates shortest paths.

## Internet Routing

- Routing Information Protocol:
  - Distance vector protocol.
  - Hop count metric
  - Exchange is done every 30 seconds, fault detection every 180 seconds.
  - Cheap and easy to implement, unstable in the presence of faults.
- Open Shortest Path First:
  - Link state protocol.
  - Internal hierarchy for better scaling.
  - Optimization for broadcast LANs with routers on them. (A designated router represents the whole LAN) - Saves control messages and size.

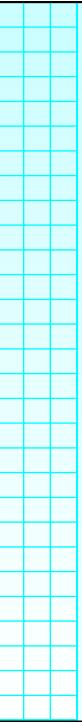
## Internet Routing (cont).

- A hierarchical routing protocol that connects networks, each of which runs an internal routing protocol.
- OSPF or RIP are common internal protocols.
- BGP - Border Gateway Protocol -
  - A path vector protocol with additional policy information for each path. Path vector protocols have the complete path in each entry and not only the next direct member.
  - Generally used as the hierarchical routing protocol.

## Important Issues



- Flow Control.
- Stability.
- Management.
- Security.



## Information Slide

- Code material is available on the CS undergrad lab machines (ugrad1-ugrad20) in the directory ~cs437/tutorials/
- A web page of reference material and programming documentation is available at <http://www.dsn.jhu.edu/courses/cs437/ref.html>
- Lecture slides can be obtained at the course web page <http://www.dsn.jhu.edu/courses/>.
- **Join the course mailing list!!**
- Make sure you have a Linux account for the undergrad lab ugrad1-ugrad20.