Communication Architectures

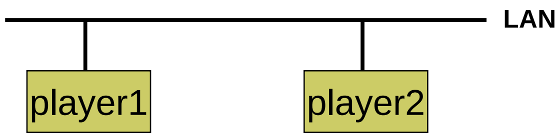
448430
Spring 2009
3/30/2009
Kyoung Shin Park
Multimedia Engineering
Dankook University

Outline

- 2 Players on a LAN
- Network Models
  - Centralized Server-Client
  - Distributed Peer-to-Peer
  - Hybrid
- Communication
  - Broadcast
  - Multicast

Two Players on a LAN

- Logical architecture
  - How the messages flow
- Physical architecture
  - How is it really wired
  - How do our messages really flow



Two Players on a LAN

- Latency
  - 100 milliseconds, 200 milliseconds, Or what?
  - How long before we can't control or react to the other players across the net?
  - This the THE BIG DEAL, IN FACT THE WHOLE DEAL with NVEs.

Two Players on a LAN

- DIS Protocol PDU
  - 144 bytes in length and contains the information necessary to manage entity state for the various players.
- NPSNET-IV Performance with DIS
  - Assume graphics performance at 30 frames per second
  - Aircraft 12 PDUs per second
  - Ground vehicles 5 PDUs per second
  - Weapon firing 3 PDUs per second
  - Fully articulated humans 30 PDUs per second (frame rate)
  - Heartbeat PDU every 5 seconds.

Two Players on a LAN

- Some network parameters
  - 56 Kbps for V.90 modems
  - 10 Mbps for our Ethernet LAN (saturates at 70% utilization)
  - 1.5 Mbps for T-1 lines
- Assumptions
  - Sufficient processor power
  - No other network usage
  - A mix of player types

Two Players on a LAN

- How many players can we support?
  - Ethernet LAN
  - V.90 modems

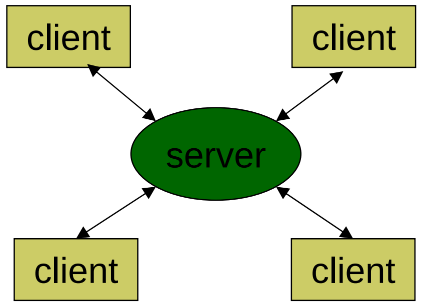
Two Players on a LAN

- Ethernet LAN
  - 7 Mbps of available bandwidth
  - A PDU is 8 bits/byte times 144 bytes or 1152 bits.
  - Meaning we can send about 6,076 PDUs per second on our Ethernet LAN.
  - If we assume 'mayhem', all players firing 1 per second, we can make some statements about the outer bounds of what we can support.
    - Assume fully articulated humans (30 PDUs/second) firing once per second (3 PDUs/second), we can have 184 humans on such a LAN.
    - Assume all aircraft (12 PDUs/second) firing once per second (3 PDUs), we can have 405 aircrafts.
    - Assume all ground vehicles (5 PDUs/second) firing once per second (3 PDUs), we can have 759 ground vehicles.
    - So we can support between 184 to 759 players on our LAN, assuming away all other usage.
  - NPSNET-IV maxed out at about 300 players.

Two Players on a LAN

- V.90 Modems
  - 56 Kbps of available bandwidth implies about 48 PDUs/second.
    - 1 human, 3 aircraft or 6 ground vehicles.
  - If we do better than that and make our packets only 22% of that size, we can have 218 PDUs/second.
    - 6 humans, 14 aircraft or 27 ground vehicles.
  - In a two-player NVE on a LAN, the protocol selection (TCP, UDP, broadcast, ...) hardly matters
  - As the number of live or autonomous players increase an efficient architecture becomes more important

Network Model: Centralized

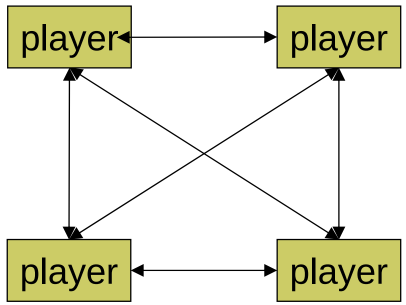


- Client-server model
- Each player sends packets to other players via a server
- One computer (server) collects all data and sends updates to the users (clients)
- Simple structure, easy to maintain database (useful for compression & admin tasks)
- Not scalable, the central server is the bottleneck

Network Model: Centralized

- Benefits of Client-Server Architecture
  - No need to send all packets to all players
  - Compress multiple packets to a single packet
  - Smooth out the packet flow
  - Reliable communication without the overhead of a fully connected NVE
  - Administration

Network Model: Distributed



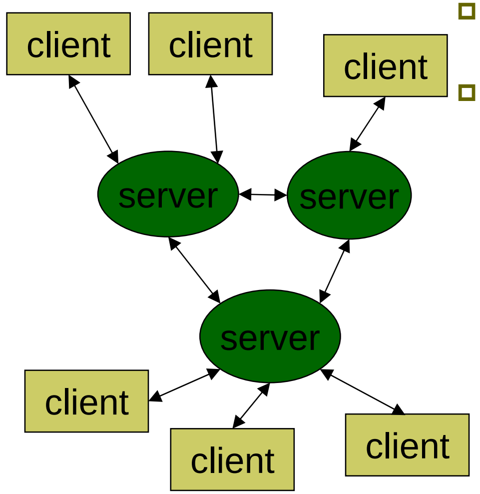
Broadcast or multicast?

- Peer-to-peer model
- Each user maintains its own copy of the database
- Updates are send to other users
- Difficult to manage the number of connections
- Not scalable, the network is the bottleneck

Network Model: Distributed

- In the ideal large-scale NVE design, avoid having servers at all
  - Eventually we cannot scale out
  - A finite number of players
- Design goal
  - Peer-to-peer communication
  - Scalable within resources
- Peer-to-peer communication goes directly from the sending player to the receiving player (or a set of them)

Network Model: Hybrid



- Multi-players client-server model with multiple servers
- If we are using a multiplayer videogame service company

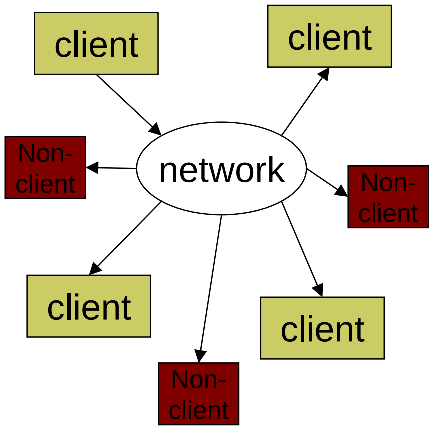
Network Model: Hybrid

- Players can locate in the same place in the NVE, but reside on different servers
  - Real World != Virtual World
- Server-to-server connections transmit the world state information
  - WAN, LAN
- Each server serves a number of client players
  - LAN, modem, cable modem
- Scalability

How to avoid bottlenecks?

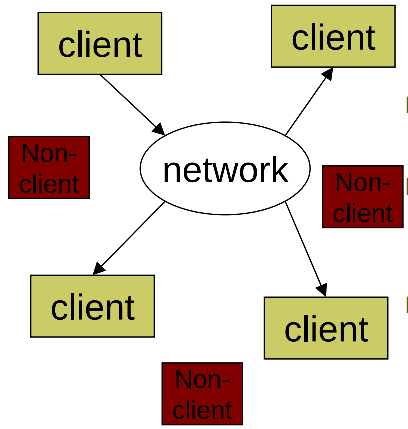
- Better communication models
  - Reduce number of connections and messages
- Better database models
  - Distributed databases
- Better decision making
  - Make it distributed but any given decision is made in only one place

Broadcast Communication



- The message is sent to all users (and non-users)
- Not selective
- Floods network with packets
- All packets must be brought up through the kernel of the operating systems of all users
- Even if the packet is not for that machine! Thereby, wasting CPU time.

Multicast Communication



- The message is sent to the multicast group (and therefore to all group members)
- Non-users (non-group users) do not receive messages
- To utilize multicast, assign packets to proper multicast groups.
- Multicast services allow arbitrarily sized groups to communicate on a network via a single transmission by the source.
- Can inter-network (route over the network layer) with multicast.

How many players can we support with a DIS-like architecture?

Table 4 - 1 Players supportable in a DIS-like networked virtual environment versus various available network tec

Assumptions

1. Infinite compute cycles at each node.
2. Infinite graphics cycles at each node.
3. Network interface with infinite cycles.

	Bytes	Bits	
Packet size for DIS-like VEs =	144	1152	
Packet size for Game-like VEs =	32	256	
Video/audio stream (bytes/second) =	70,000	560000	Origin - Media 100 Natural Video rate.

	Just moving	Mayhem	
Articulated humans (PDUs/second) =	30	33	Mayhem used for Min. Players
Aircraft (PDUs/second) =	12	15	Not used but here for completeness.
Ground vehicles (PDUs/second) =	5	8	Mayhem used for Max. Players

Technology	Min. Speed bps	Max. Speed bps	Min. Players DIS-like	Max. Players DIS-like	Min. Players Game-like	Max. Players Game-like	Min. Players Game & Video
V.90 Modem	28,800	56,000	1	6	7	27	
DSL	384,000	1,500,000	39	163	178	732	111
T-1	1,500,000	1,500,000	39	163	178	732	111
Cable Modems	2,000,000	10,000,000	263	1085	1184	4883	1117
10BT	7,000,000	10,000,000	263	1085	1184	4883	1117
100BT	70,000,000	100,000,000	2630	10851	11837	48828	11771

References

Dawson, Fred "XDSL Market Blooming," Interactive Week, Vol. 5, No. 39, 12 Oct. 1998, pp. 28-29.

Cable Modem FAQ, http://www.cox.com/highspeed/modemfaq.html

Areas of Interest in the VE

- In the real world, which virtual environments emulate, entities have a limited "areas of interest".
- We can exploit this with multicast communications.
- Interactions are mediated by an Area-of-Interest Management (AOIM) software layer.
  - Assign outgoing packets to the right groups
  - Receive incoming packets to the appropriate multicast groups
  - Keep track of available groups
  - Even out stream information

Area of Interest Management (AOIM) Software Layer

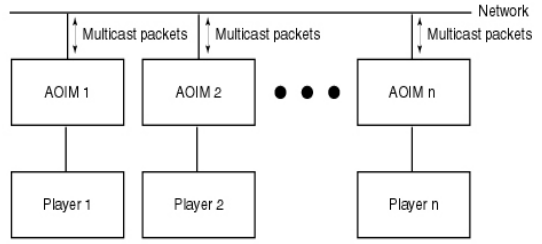


Figure 4-7 Area of interest management (AOIM) software layer

Area of Interest Management Partitioning Classes

- Spatial
- Temporal
- Functional

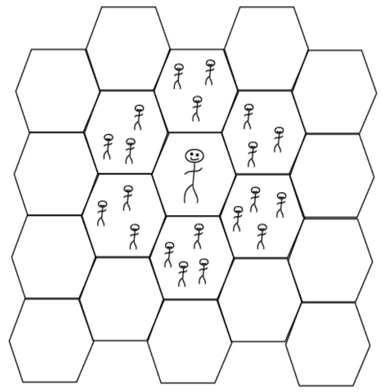


Figure 4-8 Spatial partitioning of the virtual environment using AOIM

Area of Interest Management is Distributed

- Partition the simulation into workable chunks to reduce computational load on hosts and minimize communications on tail links.
- Distribute partitioning algorithms among hosts.

Simulation

- Does partitioning using our architecture reduce bandwidth and computational requirements for large-scale virtual environments?
- Does the architecture scale and if so how well?

How we did the simulation ...

- We took real world data from the National Training Center from a battle scenario.
- We used a constructive model on the real world data to fill in the player positions and actions.
- We ran a simulation of our Area of Interest Manager and kept records on the distribution of players among the hexes.
- Peak multicast traffic around 2Mbits/second, just over T-1 speed (flat for the hex sizes).
- Peak at 1,500 players in an area of interest for Multicast.
  - Apparently unbounded number of players (packets) per machine for broadcast, with the app having to deal with these packets.