

A decorative graphic on the left side of the slide, consisting of a network of light blue lines and circles that resemble a circuit board or a network diagram. The lines are vertical and horizontal, with some diagonal connections, and the circles are small and white with blue outlines.

GPAT – CHAPTER 12

NETWORKED GAMES

INTRODUCTION

- Networked games allow multiple players to connect over the internet and play together
- Provides a unique player experience to interact cooperatively, competitively, and social with other players





PROTOCOLS

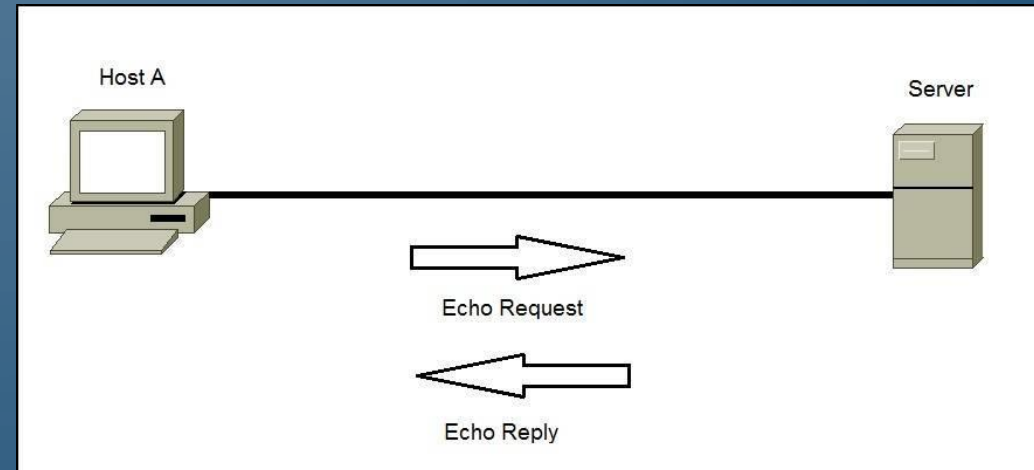
PROTOCOLS



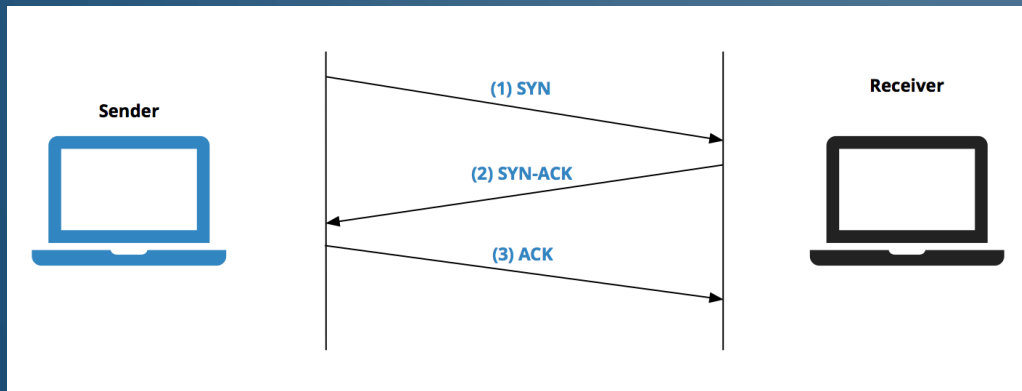
- Information sent over the internet (or network) is like sending a virtual letter. So it knows:
 - Who sent it
 - Where is it going
 - Time it is sent
 - Contents (data)
- This is called a **packet**
 - The logistical information is its **header**
 - The data is its **payload**
- The rules defining how a packet is laid out and what happens when it is sent is called a **protocol**
- **Internet protocol**, IP, is the base protocol that must be followed to send any data over the internet. More complex schemes are built on top of it.

INTERNET CONTROL MESSAGING PROTOCOL (ICMP)

- Not designed for large data transmission, i.e., game data
- Useful for **echoing** to determining connectivity and measure **latency** by **pinging**
- Basically used to send a timestamp back and forth



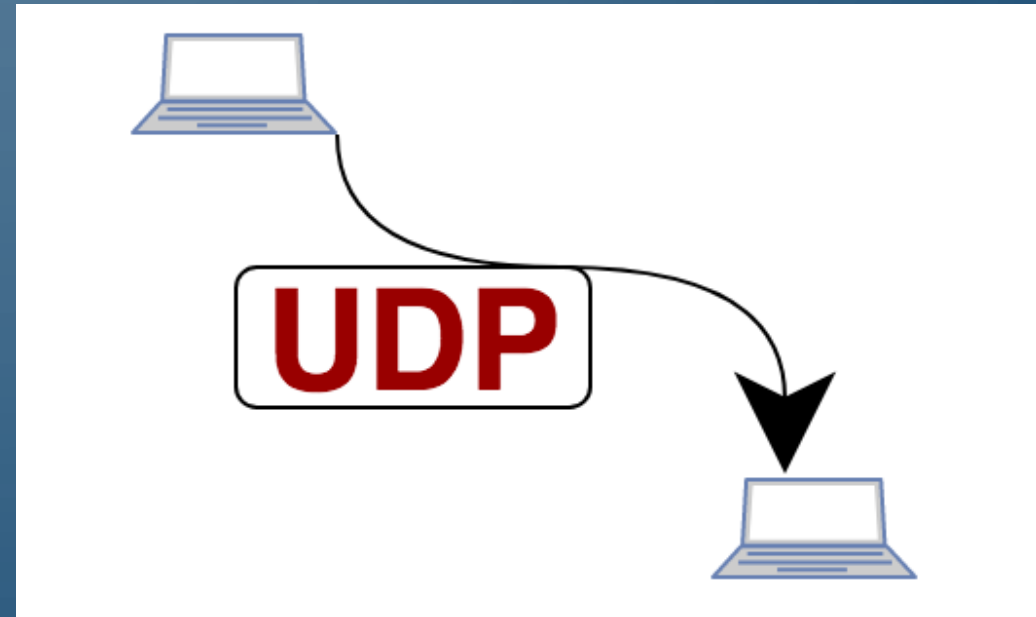
TRANSMISSION CONTROL PROTOCOL (TCP)



- One of two methods to transmit game data
- Connection-based protocol that provides for guaranteed delivery of all packets in the correct order to a specific **port** on a computer
- If an **acknowledgement** is not received in a certain amount of time (**timeout**) a packet is resent
- Which games features is this useful for? Why is it not great for most games?

USER DATAGRAM PROTOCOL (UDP)

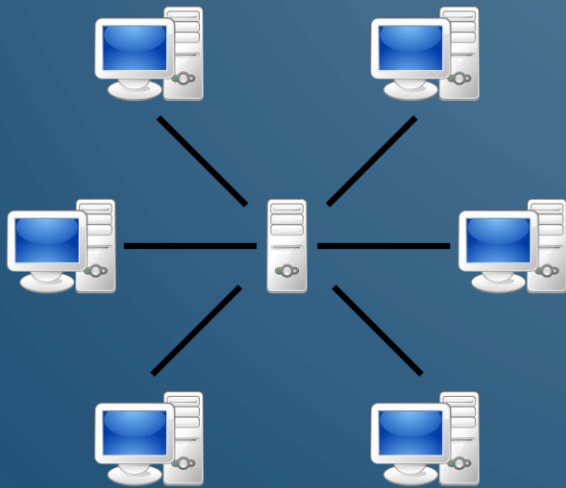
- Connectionless protocol that is "unreliable", i.e., you can send data to a port without actually having a connection
- No guarantee that a packet is received, nor in any particular order
 - Implement your own ordering through sequence numbering
- Most common for use in games. Why?



The image features a dark blue background with white, stylized circuit board traces in the corners. These traces form various geometric shapes and connect to small white circles, resembling nodes or components in a network diagram. The traces are most prominent in the top-left, top-right, bottom-left, and bottom-right corners, framing the central text.

NETWORK TOPOLOGY

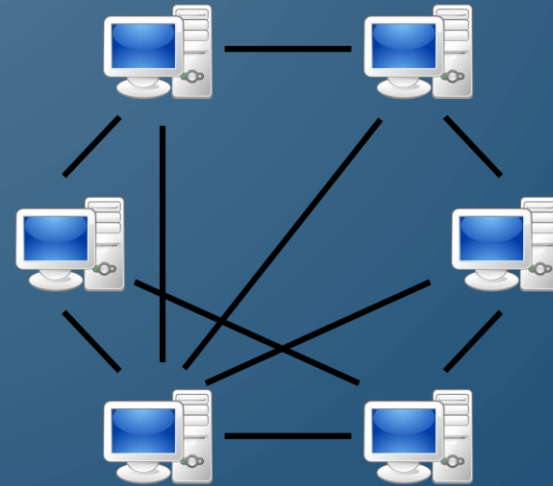
SERVER/CLIENT MODEL



- There is a central computer (**server**) that all other computers (**clients**) connect to
- Most common in games today
- Server is **authoritative** and validates clients actions. Often supported by a **dedicated server**. Why?
- Clients often employ **client prediction**
- Problems?

PEER-TO-PEER MODEL

- Clients connects to all other clients
- Play is often performed in **lockstep**, e.g., real-time strategy games
 - Actions are cued and executed every so often
- Game is simulated on clients individually (means no randomness possible)



The image features a dark blue background with white, stylized circuit board traces in the corners. These traces consist of straight lines of varying lengths and angles, ending in small white circles, resembling electronic components or nodes on a board. The traces are located in the top-left, top-right, bottom-left, and bottom-right corners.

CHEATING

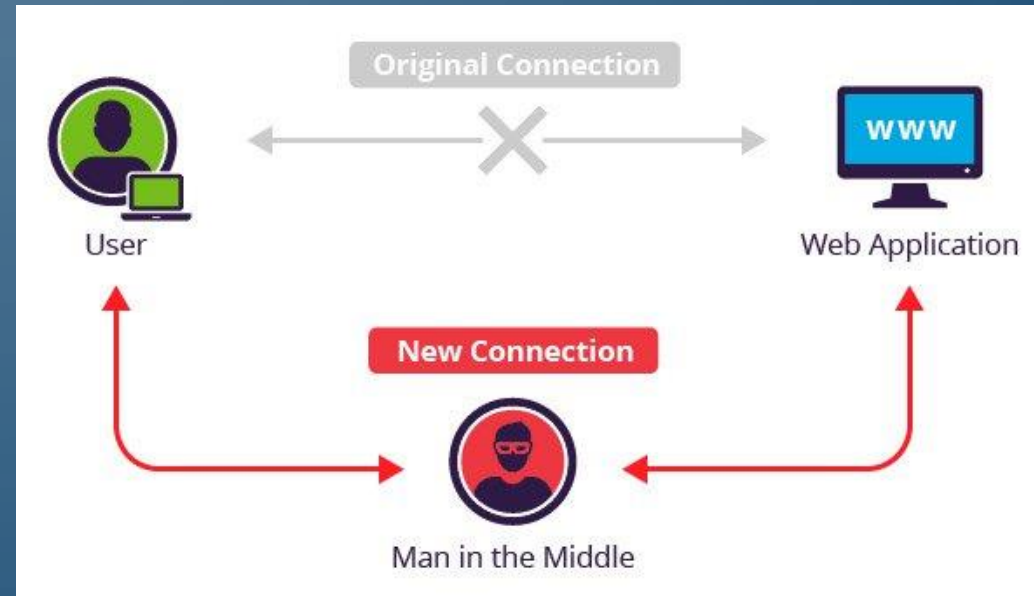
INFORMATION CHEATS



- In **information cheats**, one player is able to get information that players are not normally allowed to have
 - Example – being able to find a stealth character because their position data is still sent by the server
 - Example – seeing all player movement in RTS games (map hack)
- Can be stopped by limiting available information or cheating countermeasures

OTHER CHEATS

- In **game state cheats** a player modifies the state of the game, thus breaking it
 - Example – host server modifies the game
- In a **man-in-the-middle attack** you route all information through another computer that intercepts and modifies packets
 - Can be overcome through encryption of packets



A decorative graphic on the left side of the slide, consisting of a network of light blue lines and circles that resemble a circuit board or a data network. The lines are vertical and horizontal, with some diagonal connections, and the circles are small and white with blue outlines.

FGD – CHAPTER 17

DESIGN ISSUES FOR ONLINE GAMING

ADVANTAGES AND DISADVANTAGES OF ONLINE GAMING

Advantages

- Socializing
- Human intelligence over AI
- Online play vs local multiplayer
 - Why an advantage?

Disadvantages

- Technical issues
 - Communication
 - Latency
 - Dropped/garbled packets
- Harder to suspend disbelief
- Need to produce content
- Customer services

The background is a solid dark blue color. In the four corners, there are decorative white line-art elements that resemble circuit traces or a network diagram. These lines connect to small white circles, some of which are larger than others. The lines are thin and have a clean, minimalist aesthetic.

DESIGN ISSUES

ARRIVING PLAYERS

- Need to decide when players can join
 - Rolling starts (matches)
 - Immediately (requires fast gameplay)
- Get rid of the victory condition – rather aim for achievements
- Discourage competition between experienced and inexperienced players
- Be sure competition is consensual



DISAPPEARING PLAYERS



- Players can leave at any time and so you need to handle properly to ensure minimal disruption to others
 - The vanishing player forfeits
 - Institute a penalty for disconnections
 - Award victory to whomever is ahead at disconnection
 - Record as a tie or disconnected game
 - Abandon the game
 - Use referees

REAL-TIME VS TURN-BASED GAMES

- Considerations of turn-based:
 - Limit number of players in one game
 - Set time limit on players turn
 - Determine default action if player runs out of time
 - Let players do other things while waiting



COLLUSION



- **Collusion** is a form of cheating in which players who are supposed to be opponents work together in violation of the rules
- To reduce consider how players might:
 - Share secret knowledge
 - Pass cards under the table (transfer items)
 - Take a dive (lose deliberately)

ASYNCHRONOUS GAMES

- In asynchronous games, players actions are not synchronized
 - Don't have to be logged on
 - Don't have to wait for others
- Mostly non-competitive
- Other considerations?



TECHNICAL SECURITY

- Use a secure protocol
 - Encrypt data
 - Implement heartbeats for disconnectivity
 - Add timestamp and unique serial number to packets
- Don't store sensitive data on the players computer
- Don't send the player data they aren't supposed to have
- Don't let the client perform sensitive operations

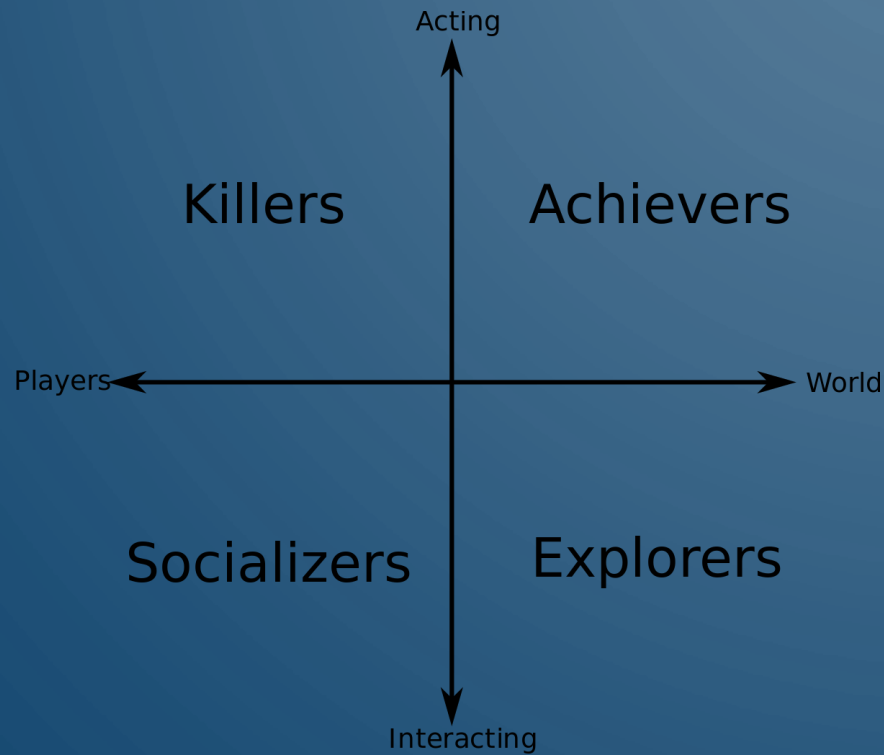
The image features a dark blue background with white, stylized circuit board traces in the corners. These traces consist of straight lines that turn at right angles and terminate in small circles, resembling electronic components or nodes in a network. The traces are located in the top-left, top-right, bottom-left, and bottom-right corners, framing the central text.

PERSISTENT WORLDS

HOW PERSISTENT WORLDS DIFFER FROM OTHER GAMES

- Persistent worlds constitute permanent environments in which players can play, retaining the state of their avatar
- Traditional narrative is difficult to implement because of the number of players, story can unfold by quests at varying scales
- Players can fill a large number of rich and varied roles
- Without a victory condition, gameplay is different as the player decides for themselves what to do (expressive vs reactive gameplay)

TYPES OF ONLINE PLAYERS



- Model proposed in 1997 to describe various types of players in online games
- Conjectures that a healthy online community requires a certain proportion of each type

CREATING AN AVATAR

Maximize expressiveness. Considerations:

- Unique name/handle
- Physical appearance
- History/experience
- Reputation
- Autobiography



WORLD MODELS



- Classic world models
 - Scavenger model
 - Social model
 - Dungeons & Dragons model
 - Player-versus-player model
 - Builder model

AVATAR DEATH

- Some options
 - Permanent death
 - Resurrection with reduced attributes
 - Resurrection with some property missing



THE NATURE OF TIME



- Game time must proceed at a fixed rate for all players
- Different than single player, how?
- Avoid design of time-consuming activities
- Time is irreversible

ECONOMIES

- Harder to tune
- Avoid being able to create something for nothing
- Maybe avoid fixed number of resources?



The background is a solid dark blue color. In the four corners, there are decorative white line-art patterns resembling circuit traces or neural network connections. These patterns consist of thin lines that branch out and terminate in small circles, creating a sense of connectivity and technology.

SOCIAL PROBLEMS

MANAGING CHAT



The screenshot shows the Battle.net chat interface for Starcraft II: Wings of Liberty. The chat window is titled "Teamliquid (9,042) Public". The chat history includes a welcome message and several messages from users like DayNine, LzGamer, Machine, KawaiiRice, Jinro, InControl, and Trump. The user profile for LzGamer is shown at the top right, with a rank of Diamond and a profile description. The settings menu on the left includes options for "Join Channel", "Custom Games", "Quick Match", "View Ladder", and "Settings".

Consider:

- Limiting content
- Profanity filters
- Complaint and warning systems
- Blocking other players
- Moderated chat spaces

PLAYER-KILLER (PK) PROBLEM

- Should you allow players to kill each other?
- Pros:
 - Human intelligence
 - Interesting loot
 - Social experience
- Cons:
 - Unfair
 - Annoying to many players
- Justice mechanisms
 - No automated regulation
 - Flagging criminals
 - Reputation systems
 - PvP switch
 - Safe games (no PvP)
- Factions are a good solution usually
- Bottom line: you can't please everyone
 - It's a fantasy world
 - People pay to play