

Multiplayer Game Synchronization

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Lecturer: Dr. Joseph Doyle

Introduction

- Massive Multitplayer Online Games (MMOGs) produce billions of pounds of revenue and have several interesting design challenges
- Can be considered a distributed system
 - The state of the game should be consistent to all users
 - Updates to the state need to be communicated to all users in a robust fashion

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

MMOG Design Principles

- There are several concepts specifically related to MMOGs namely: **Assignment Project Exam Help**
 - Object Types
 - Player Interactions **<https://tutorcs.com>**
 - Object Replication
 - Latency Tolerance **WeChat: cstutorcs**
 - Bucket Synchronization and Frame Rate
 - Bandwidth Requirements
 - Interest Management
 - Consistency Control

Object Types

- Immutable Objects
 - Landscape or terrain usually designed offline and never changed during the game
- Characters or Avatars
 - Controlled by the player using an input device and can interact with some other object types
- Mutable Objects
 - Examples include food and tools. Players can interact with these objects
- Non player characters (NPCs)
 - Similar to player characters but controlled by an AI
- Each of these objects need to be handled differently
- For example information on Immutable objects does not need to be sent as it never changes

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Player Interactions

- Player updates
 - Updates that only affect the player such as position updates
- Player object interactions
 - A player interacting with a mutable object such as food is an example of this
- Player player interactions
 - A player interacting with another player such as waving to them (and the various less friendly alternatives)
- Again these need to be handled differently
- Do we need to send information on a player update if it is outside the Area of Interest (AOI) (more on this later) of another player?

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Object Replication

- When a player joins an MMOG they receive an instance of a game world
- This instance is usually a *secondary* copy of the game world which contains objects relevant to the player
- The primary copy is a definitive version of the game world which is sometimes held on a special game server
- When players update objects the update must be sent to the primary copy which will authorise the update (it may not do this if it believes the update is related to a cheat for example)
- This update is then sent to all secondary copies of the game world
- Similar to publish-subscribe systems (RSS, MQTT(IoT), JMS etc)

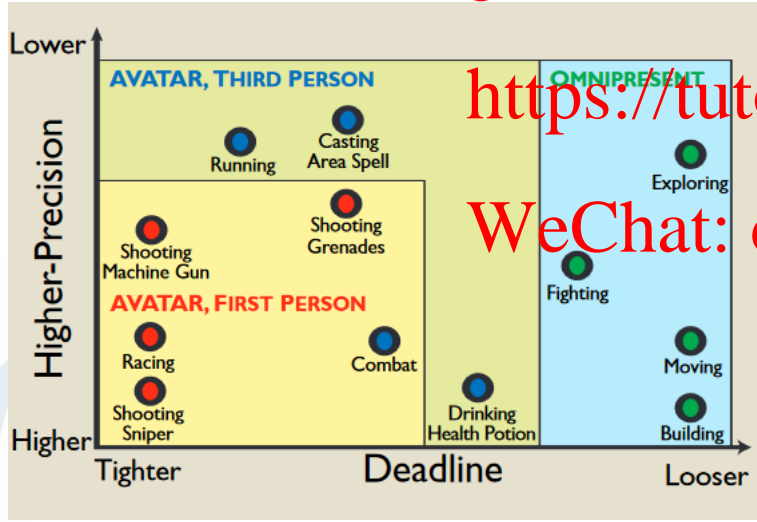
Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Latency Tolerances

- There is a point where network delays or latency make the game frustrating or impossible to play but this varies significantly depending on the game type
- This can dramatically affect the architecture



<https://tutorcs.com>

WeChat: cstutorcs

<https://web.cs.wpi.edu/~claypool/papers/precision-deadline/final.pdf>

Bucket Synchronization and Frame Rate

- Bucket Synchronization is a techniques for handling latency in MMOGs **Assignment Project Exam Help**
- Each client will have a different latency and the primary copy of a game state will receive multiple updates concurrently from different clients **<https://tutorcs.com>**
WeChat: cstutorcs
- Many games deliberately lag behind the execution of events to improve the fairness despite latency issues
- This is similar to Nagle's algorithm in IP networking

Bucket Synchronization and Frame Rate (cont)

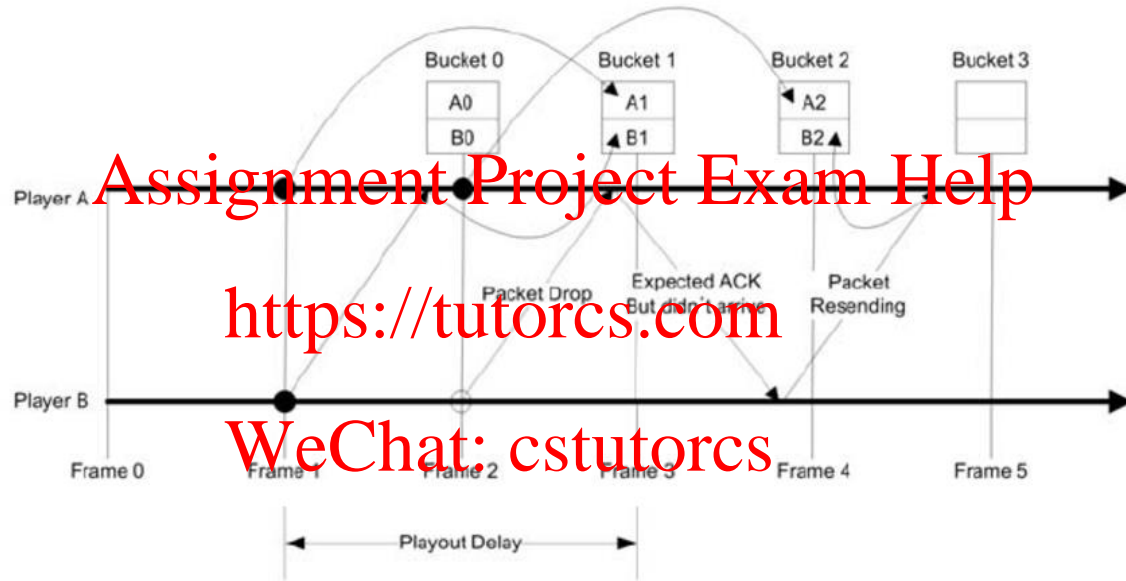
- Games implement a discrete event loop (also known as a frame) in which all action which have occurred since the last loop are received buffered and executed
- Updates are sent at the end of the loop
- NPCs also execute a think function which determines their actions during the loop
- This loop is executed 10 to 20 times a second which is known as the frame rate
- Note that this is different to the display frame rate

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Bucket Synchronization and Frame Rate

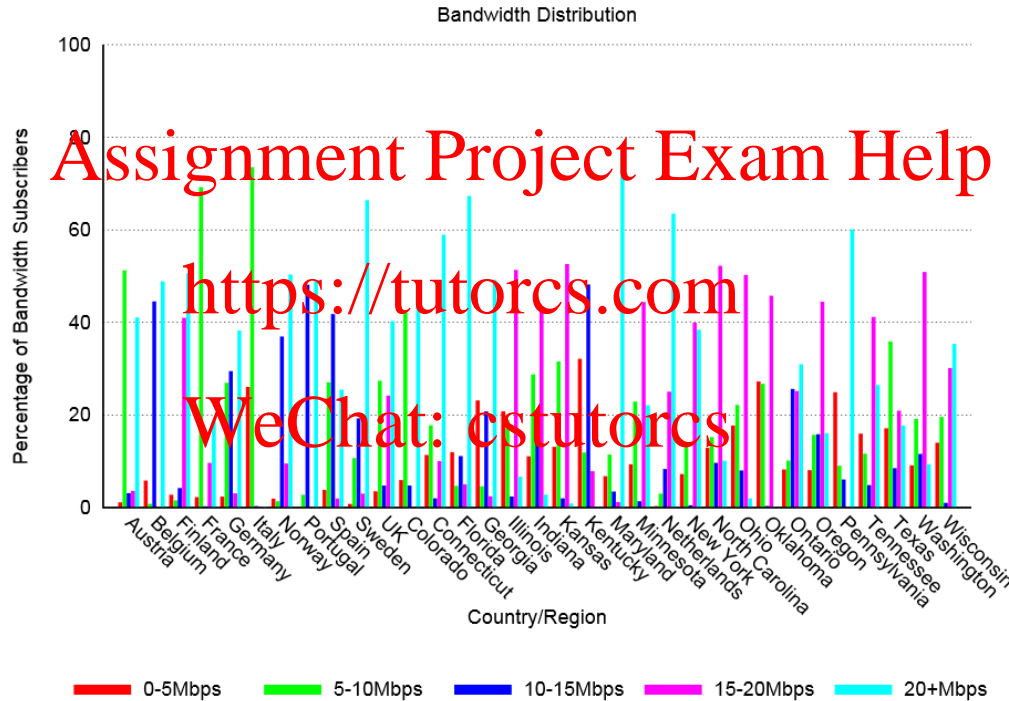


https://www.researchgate.net/figure/The-Locked-Bucket-Synchronization-algorithm_fig1_29461970

Bandwidth Requirements

- Bandwidth requirements are based upon the number of active users, message size and update rate
<https://tutorcs.com>
[Assignment Project Exam Help](#)
- Games must also accommodate bursts in traffic due to changes in environments or activity
[WeChat: cstutorcs](#)
- This is typically dealt with via over provision

Bandwidth Requirements



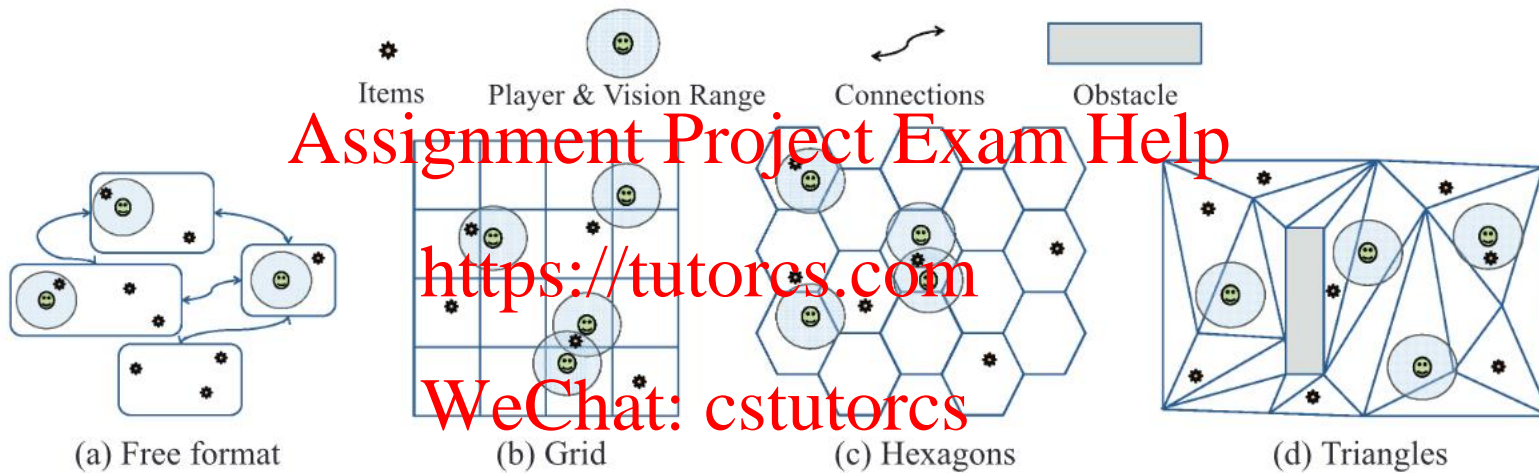
Interest Management

- Important mechanism used in games for scalability reasons
- Limits the players movement and vision capabilities
- Thus, players can only move a small distance in the game in a given time interval
- This allows data access which shows spatial and temporal locality (This is very useful when it comes to caching data)
- Players only receive the game state that is relevant to them based upon their position and vision which improve scalability

Interest Management (cont)

- The area around a player in which they can perceive game objects is known as the Area of Interest (Aoi)
- One common mechanism for managing Aoi is zoning where the game world is divided into smaller sections or zones
- In the most simple version of this the player can interact with all game objects in the zone
- In more complicated version players can interact with multiple zones to allow a continuous game world

Interest Management (cont)



<https://dl.acm.org/doi/10.1145/2522968.2522977>

Interest Management (cont)

- Also possible to use the concept of mini-world which are zones which have a free format and portals connecting them
<https://tutorcs.com>
- This does not offer the illusion of a continuous world
[WeChat: cstutorcs](#)
- Determining the right size of a zone is challenging
- Trade-off between network workload and performance

Interest Management (cont)

- There are some limitations to interest management as flocking in MMoGs is common
- Flocking is movement of players to one specific area in the game world
- Possible to reduce the problems associated with flocking via dynamic zones
- Limitations to this as two many dynamic zones can reduce performance and there is also the possibility of inter zone communication becoming a bottleneck

Assignment Project Exam Help

<https://tutor.com>

WeChat: estutours

Consistency Control

- Like other distributed systems concurrent and possibly conflicting updates are possible
- Inconsistency can be caused by latency or loss of updates
- Games tend to use UDP for performance reasons which can exacerbate this
- TCP or commit protocols can be used to try and improve consistency

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Consistency Definitions

- Several types of consistency for games have been defined
 - Very strong consistency is defined as every interaction with a game object being treated as a transaction
 - Eventual consistency is defined as a game in which all copies of game objects would reach the same value if updates stopped
- Brewer's CAP theorem suggest that only two out of three properties of consistency, availability, and handling network partitions can be achieved in a distributed system.

Consistency Control

- MMoGs are slightly different as they tend to aim for inconsistency resolution rather than inconsistency prevention as interactivity is a key component
- Different levels of consistency, however, can be performed on different mechanisms
- For example, strong consistency may be used in transactions involving virtual currency as this frequently has real world value

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Stale Views

- When a secondary copy of a game state has sent an update to the primary copy but has not yet received an update from the primary the view can be defined as stale
- Invalid update requests are therefore possible
- In 50% of cases this inconsistency lasts less than a second
- There are several techniques which games use to hide inconsistency and message loss

Consistency Techniques

- Consistency Techniques broadly fall into two categories namely:
 - Predictive Contract Mechanism (PCMs)
 - Dead Reckoning
 - Smooth Corrections
 - Multiresolution Simulation

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Dead Reckoning

- Dead reckoning is based upon the assumption that game objects rarely change speed or direction
- The previous movement is therefore an accurate predictor of the current movement
- An extrapolation algorithm can be used to estimate the position of a game object and reduce the number of updates to the game object
- Updates are only transmitted when an error threshold is reached
- It is possible to implement dead reckoning with lag awareness

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Smooth Corrections

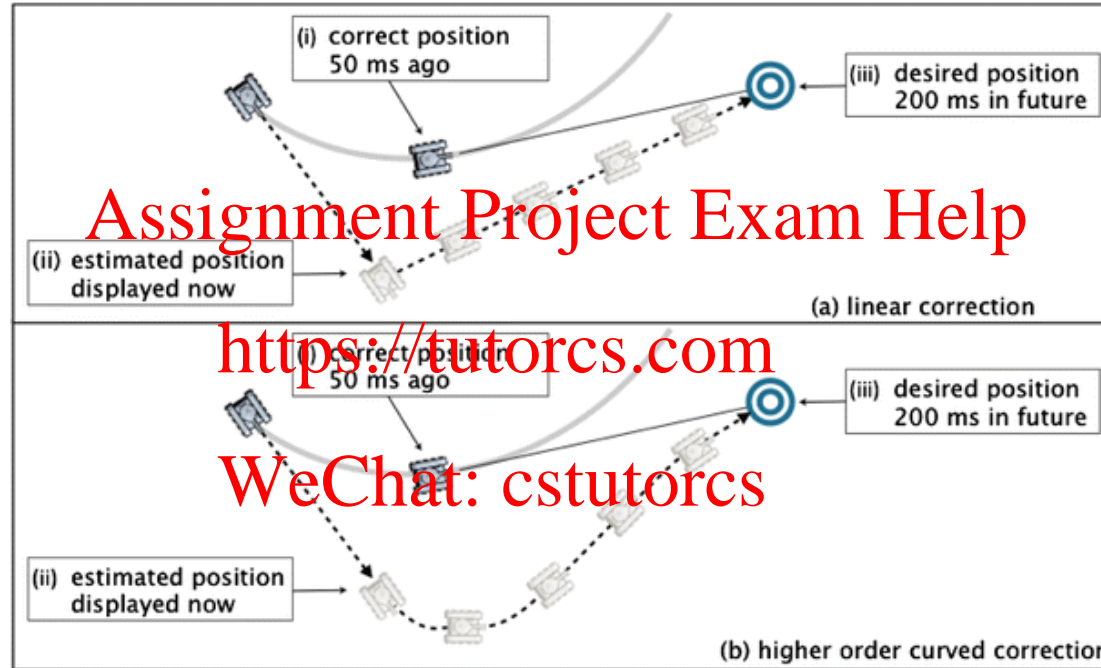
- When a dead reckoning update arrives which is significantly different from the current predicted location the simplest solution is move the play to the new location
- This result in jerky animation and can be visually jarring for the player
- Convergence techniques can be used to correct these errors in a smooth less surprising manner
- By estimating where the player will be in the future it can be moved progressively to that location

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Smooth Corrections



<https://link.springer.com/article/10.1007/s00530-012-0271-3>

Measuring Consistency

- Can be simply calculated as the difference between the state at each client and the virtual perfect site that receives and executes all interactions with no delay and in the right order
- Also possible to evaluate consistency based upon three metrics namely
 - **Responsiveness:** The time the system takes to respond to an event
 - **Precision:** The degree of accuracy required to complete an action successfully
 - **Fairness:** The degree of difference among all players' gaming environments
- Could also consider **inconsistency**, **interactivity**, and **discovery latency**

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

MMOG Architecture

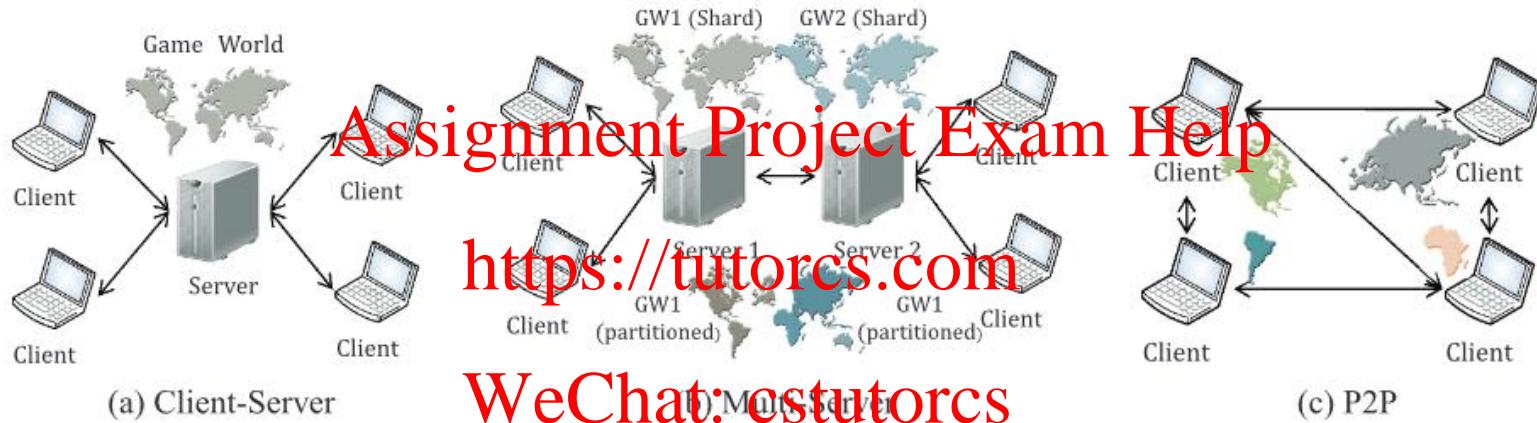
- There are a number of different architectures which can be used to implement MMOG namely
 - Client Server
 - Multi Server (MS)
 - Peer-to-peer (P2P)

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

MMOG Architecture



<https://dl.acm.org/doi/10.1145/2522968.2522977>

Client Server Architecture

- In this architecture the server holds the master copies of all mutable objects and avatars and maintains global knowledge of the game world
- Clients connect to the server and receive necessary information about the game world
- Updates and interactions are sent to the server for execution and conflict resolution
- Once the updates have been executed the server forwards the updates to players who are affected

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Client Server Architecture Advantages/Disadvantages

- This system tends to get the best performance for low numbers of players as no inter server communication is required
- Performance tends to decrease as the number of players increases as even the best provisioned server can only support a limited number of players
- It is also a single point of failure and fault tolerance can be an issue
- Multiple servers can be added to improve performance

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Multi Server Architecture

- This is the approach normally taken by large game companies
- Server farms are maintained to provide services to clients
- There are two main categories of this architecture namely:
 - Shards
 - Regions

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Multi Server Architecture Shards

- A shard is defined as a complete instances of the game world and several of these exist at the same time
- Each shard is maintained by a separate server
- Each server is responsible for a different set of clients and follow a traditional client server architecture
- As each server contains a complete copy of the game world there is no need for communication between the servers
- Games are usually designed to make it difficult to move between shards (E.g. £17 to transfer character between zones in WoW). Why?
- Users are typically assigned to servers based on their geographical location. Why?

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Multi Server Architecture Regions

- With this architecture only a single game world exists but the game world is divided into several regions
- Each region is maintained by a server
- Players can only interact with players who are in the same region
- Players can move between regions but this requires some mechanism to move the character between servers in a transparent fashion
- For example as a player hears a border necessary information can be sent to the server handling that region so that the player can easily cross the border
- It is also possible to use portals or gateways so that transparent movement between regions does not need to be implemented
- This approach is useful when implementing interest management

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Multi Server Architecture Shards vs Regions

- Shards allow players to communicate with any player in the world without additional network traffic (Regions would require the message to be forwarded to another server if the player was in a different region)
- Regions allow a greater density of players in a particular region
- Both approaches can be used in tandem. E.g. It is possible to have shards of regions but this is only necessary in very large games

Multi Server Architecture Advantages/Disadvantages

- Allows a greater number of players to play the game without performance problems
- Increased complexity. If regions are used the game needs to handle functionality associated with global player interactions (Assuming this is a function of the game) as well as the transition of players between regions
- Using regions allows for a greater number of players to participate in the same game world assuming additional functionality can be implemented
- Fault tolerance is also improved as the loss of a shard will only affect some of the players
- Very expensive. Acquiring servers for 30,000 simultaneous players can cost approximately \$800,000 and the bandwidth costs can reach hundreds of thousands of dollars

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Peer to Peer (P2P) Architecture

- In this architecture each client also acts as a server
- Each node can become responsible for maintaining a master copy of some game objects and disseminating updates to other players
- As more nodes join the game more resources become available for use by the distributed game system

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Peer to Peer (P2P) Architecture Advantages

- Improved scalability. Each player who joins the game adds more resources to the game
- Reduced cost. As the clients are maintaining the game world there is no increased costs relative to the scale of the game
- Improved failure tolerance. The failure of an individual node should not greatly affect other nodes assuming mechanisms for handling the failure exist
- Low latencies can be achieved by making direct connections between peers properly as updates can be sent to interested peers directly rather than being relayed through a server

Assignment Project Exam Help

<https://tutores.com>

WeChat: ctutores

Peer to Peer (P2P) Architecture Disadvantages

- Security. Cheating is much easier in a P2P architecture
- The systems are much more difficult to manage as the game state is distributed among the peers
- Consistency is a problem as conflicting updates which are executed at different sites may result in an inconsistency
- The complexity and coordination overhead will tend to increase as more clients join the game

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Architecture comparison

Architecture	Advantages	Disadvantages
Client-Server	<ul style="list-style-type: none">+ Simplicity+ Easy Management+ Consistency Control	<ul style="list-style-type: none">- Scalability-- Fault tolerance- Cost
Multi-Server	<ul style="list-style-type: none">+ Scalability+ Fault Tolerance	<ul style="list-style-type: none">- Isolation of players- Complexity-- Cost
Peer-to-Peer	<ul style="list-style-type: none">++ Scalability++ Cost+ Fault Tolerance	<ul style="list-style-type: none">-- Complexity- Consistency Control- Cheating

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Hybrid Architecture Techniques

- There are a number of hybrid architectures which can utilise components of the different architectures discussed namely
 - Cooperative Message Dissemination
 - State Distribution
 - Basic Server Control

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Cooperative Message Dissemination

- In this technique the game state is maintained by one or multiple servers and updates are disseminated using a P2P approach
- Players send their updates directly to the server
- A P2P multicast mechanism is used to update the players
- This reduces the bandwidth requirements of the game

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

State Distribution

- In this technique the game state is distributed among the peers

Assignment Project Exam Help

- Clients hold the primary copies of objects
- All communication between peers is managed by servers
- Servers are also responsible for authenticating players and determining which players are logged in
- Scalability is achieved by distributing the cost of state execution among the clients

<https://tutorcs.com>

WeChat: cstutorcs

Basic Server Control

- Message dissemination and state distribution are handled in a P2P manner
- Servers perform additional functionality such as maintaining sensitive data such as client logins, payment information, and players' progress and state.
- Servers also perform functions such as authentication and admissions control
- Servers can also coordinate some types of interactions between the peers such as those that require the highest consistency.
- P2P can also be used to distribute updates to game clients

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutores

MMOG Communication

- Maintaining a consistent game state with a reasonable Quality of Experience (QoE) for all users requires significant communication between all the nodes in the distributed system
- There are a number of mechanisms which can be used to manage this communication namely:
 - Direct Communication
 - Delta Coding
 - Selective Dissemination
 - Multicast Trees
 - Message Aggregation
 - NAT and Firewall

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Direct Communication

- The primary copy sends all object updates to the nodes that have subscribed to the associated object. (Similar to RSS feed)
- Low latency and quite simple to implement
- Can result in high bandwidth requirements
- In P2P upload capacity can become a bottleneck due to the asymmetry of most broadband connections



/QMUL



@QMUL

Delta Coding

- The message size is reduced by only sending the difference (delta) in update messages
- For instance by only sending attributes of objects which have changed since the last update the message size can be reduced
- Games frequently use UDP for performance reasons which can result in lost packets
- Complete updates can be sent periodically to correct errors in the game state

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Selective Dissemination

- This technique drops some updates when the game is overloaded to maintain interactivity
- Essentially clients attempt to avoid sending events which have already become obsolete
- For example, a position update becomes obsolete once a new move command has been issued

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Multicast Trees

- Frequently used in P2P architectures to reduce communication overhead
- Most commonly, a logical multicast tree is maintained and a single message is sent to the root of the tree which will forward it on to its children
- This lowers the bandwidth requirements but increases latency as the number of hops required to reach the final destination increases
- This, however, can be minimised by attempting to maintain geographical locality when constructing logical multicast trees

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Message Aggregation

- Multiple messages can be merged together to form a single message to reduce bandwidth requirements
- This can be particularly useful at the roots of logical multicast trees
- Artificial delay can be added before sending messages in order to allow for more messages to arrive and be aggregated together

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

NAT and Firewall

- Incoming connection can be difficult for clients behind Network Address Translation (NAT) devices and Firewalls
- Clients unable to accept incoming connections can rely on other clients or servers for updates
- Possible to improve the number of servers able to receive incoming connections using techniques such as hole punching and STUN

Cheating

- Cheating can be defined as “Cheating occurs when a player causes updates to game state that defy game rules and result in unfair advantage.”
- There are several categories of cheating namely
 - Interrupting Information Dissemination
 - Illegal Game Actions
 - Unauthorized Information Access

Interrupting Information Dissemination

- These cheats are caused by delaying, dropping, corrupting, or changing the rate of the updates or sending wrong or inaccurate information
- Variations include
 - Fast Rate Cheat: The cheater mimics a rate of game event generation that is faster than the real one
 - Suppress-correct cheat: The cheater purposefully drops a number of consecutive updates and then sends an incorrect update that provides them with some advantage
 - Replay cheat: A cheater resends signed and encrypted updates of a different player that they have previously received
 - Time cheating: The player deliberately delays the updates they send to base their actions on those they receive from others.

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: estutores

Illegal Game Actions

- These cheats are caused by tampering with the game code unduly change their state or circumvent the games physical laws
- Variations include
 - Client-side code tampering: The player modifies the client-side code to get an unfair advantage
 - Aimbots: The player uses an intelligent program to provide them with automatic weapon aiming
 - Spoofing: The cheater sends messages pretending to be a different player

Assignment Project Exam Help

<https://tutores.com>

WeChat: cstutores

Unauthorized Information Access

- These cheats are caused by exploiting information available but not supposed to be disclosed (e.g., position of players behind game objects) to increase their chances to outperform other players or to foresee danger, thus helping them evade.
- Variations include
 - Sniffing: Tools that allow players to log and access all kinds of information sent by the game across the network
 - Maphack: The player, by tampering with the client code and libraries, is able to see through walls and obstacles, gaining an advantage.
 - Rate analysis: Multiresolution techniques, where updates are sent to interested players more often than the ones that are not interested, are prone to cheating by using traffic analysis

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

Cheating Prevention

- There are a number of techniques which can be used to prevent cheating namely
 - Cryptographic measures: Cryptographic measures, such as message encryption, signatures, and checksums, are effective in eliminating message sniffing and illegal message modifications
 - Commitment and agreement protocols: Commitment and agreement protocols are another mechanism to prevent cheating, addressing a range of attacks. For instance, time cheating is addressed by the lockstep protocol.
 - It requires all players to first submit a hash code of their next actions and only after every player's hash code has been received, players send their actions to each other.
 - By comparing the hash code with the action, players can make sure that other players have not changed their actions after receiving input from others

Cheating Detection

- There are a number of techniques which can be used to detect cheating namely
 - Verification: A common technique is that all actions are audited and verified for security breaches
 - Client-side code tampering detection: Several tools exist to detect tampering with the game code or to detect cheating processes running in memory. PunkBuster and Valve Anti-Cheat (VAC) are examples of such systems.

Cheating Prevention

- There are a number of techniques which can be used to prevent cheating namely by punishing players which are identified as cheaters
- For instance Blizzard banned thousands of Overwatch players in 2017
- A reputation system can be used to distinguish players who experience genuine network problems and those who are actually cheating
- Once the number of suspected actions exceeds a threshold for a given time period the player can then be penalised

Assignment Project Exam Help

<https://tutorer.com>

WeChat: oztutors

Further Reading

- <https://dl.acm.org/doi/10.1145/2522968.2522977>
- <https://link.springer.com/article/10.1007/s00530-012-0271-3>

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs