MUXER

At-a-Glance

Muxer connects vast numbers of disparate clients across a distributed cloud and edge network. Providing bidirectional data flow from a single rendezvous point, it comprises multiple instances or “nodes”, locally clustered to ensure the most important information is processed and returned to the client as soon as possible. As more connections join the network, new nodes are automatically created, scaling to enable real-time data streaming, regardless of computational intensity.

FEATURES

- Bidirectional data flow
- Long running processes
- Low latency data streaming
- Fully customisable
- Dynamic scaling
- Asynchronous architecture

Muxer and Aether Engine

When used in combination with Aether Engine, Muxer ensures low latency, globally distributed simulations are made available to hundreds of thousands of connected clients to monitor, view and interact with in real-time. Muxer also enables client authentication, and other protections such as failing to accept updates or sending too much event data back to the simulation.

About Hadean

The Hadean Platform implements a unique process model that transforms the performance, reliability and scalability of cloud computing. It underpins a number of libraries that solve intractable distributed problems, instilling a set of core properties that allows them to run at massive scale. These libraries act as an interface for developers looking build out complex, high-performance applications across a distributed cloud and edge network.

Real-time streaming and event handling across a distributed cloud and edge network

Creating long running processes between client and server is an architectural problem that has existed since the inception of distributed computing. Traditional communication methods require a new client-initiated connection flow to be established each time an I/O exchange is requested. It’s inefficient and renders it difficult to push messages out; giving rise to a host of environment-specific workaround solutions, such as websockets and SSE. However, these technologies have limited performance and adaptability, constrained by inflexible netcode and a high-level API.

Muxer by contrast includes a low-level API, allowing it to integrate with protocols such as TCP, UDP or even MQTT, as well as existing web solutions such as socket.io. It can be used in any scenario that requires vast numbers of connections and/or involves the flow of vast quantities of data.

The Muxer Node

Located close to the client, the Muxer Node comprises a library for event forwarding and a netcode implementation that handles the interest management (network relevancy.) It uses a binary space partitioning tree to prioritise which data gets sent to a client and reduce bandwidth.

The Muxer Client

The Muxer Client provides a deep integration into the client engine. The plugin replicates (for example Aether Engine entities) directly into the client as actors, removing the need to manually replicate code.

Key Benefits

Vast numbers of client connections

Current networking models struggle to manage vast numbers of connections and provide the requisite processing power to computational spikes. Muxer by contrast is designed to handle near limitless numbers of connections through a single rendezvous point, across a distributed cloud and edge network.

Dynamic Scale

As the number of concurrent connections, netcode CPU consumption or state size (or any combination of all three) increases, the Muxer will automatically and dynamically scale to meet demand. It is able to spin up new instances or “nodes” as required, without any preprovisioning.

Bidirectional flow

Muxer enables servers to push information and messages out to clients, as well as receive them. Once the connection is established, communication no longer needs to be client initiated.

Low Latency, Real-time Streaming

Muxer’s long running processes remove the need for repeated inefficient connection requests, reducing the overall time it takes for data to be sent back and forth. At the same time, by elevating interest management to a first class concern, Muxer optimises bandwidth usage and prioritises sending key information to the client in real-time.

Technologically Agnostic

The muxer has both a low level API for use on TCP as well as easily integrating with high level tools such as socket.io. This enables clients to communicate with either the server or one another through a range of channels.

HADEAN

How it Works

Interest Management

The interest management (network relevancy) decides which data gets sent to a client in order to optimise bandwidth usage. In a game for instance, objects that are further away from the player are less important proportional to the square of distance, meaning items that are twice as far may be sent four times less frequently. In practice, this gives a **97.5%** reduction in bandwidth, with a constant gameplay experience.

Binary Space Partitioning Tree

The simulation state is reconstructed on Muxer and stored in a binary space partitioning tree in order to efficiently query which information is relevant to each client.

Customisable Net Relevancy

Given the optimal trade-off between speed and fidelity will vary from simulation to simulation, Muxer's net relevancy implementation is designed to be easily customisable.

Asynchronous I/O

Muxer makes extensive use of asynchronous I/O, ensuring thousands of connections are handled simultaneously without needing a single thread per client.

Logging, Distributed Debugging and Performance Analysis Tools

Muxer allows for the logging, analysis and debugging of any and all processes within the network. A network visualiser provides enormous amounts of information about the connections between them. Not just latency and bandwidth, but also detailed statistics, such as the lost packets, the window sizes or the time since the last send or receive.

Practical Applications

MMOs

Muxer allows for hundreds and thousands of players to join a single-sharded game from anywhere in the world. Games become truly social networks, can even facilitate player run worlds and economies.

Single Synthetic Environments

Muxer can integrate with any number of data types and modelling techniques to produce a high-fidelity, realistic view of the world. It can stream vast quantities of external data in near real-time, including geospatial information, 3D environments and real-time weather systems.

Virtual Events

Muxer provides a framework for easily deploying virtual events with a greater deal of scale and immersion, creating a truly global experience that transforms the way we communicate, learn and interact.

Smart Cities

By connecting with thousands of IoT devices, Muxer enables digital twins to replicate real-life cities. It enables urban planners and transport modellers to optimise and understanding the implications of their projects, factoring in an array of data sources.

www.hadean.com | info@hadean.com

**8th Floor, 9 Appold Street,
London, EC2A 2AP
0020 3514 1170**