Article

XDN Based Network Framework Design to Communicate Interaction in Virtual Concert with Metaverse Platform

Sangwon Oh , Kwangmoo Chung , Ibrahim Aliyu , Tai-Won Um , Minsoo Hahn , Jinsul Kim [*]

Posted Date: 28 June 2023

doi: 10.20944/preprints202306.1941.v1

Keywords: metaverse; multiuser ivrtual environments; experience delivery network; content delivery network; cloud edge

Article

XDN Based Network Framework Design to Communicate Interaction in Virtual Concert with Metaverse Platform

Sangwon Oh [1], Kwangmoo Chung [1], Ibrahim Aliyu [1], Tai-Won Um [2], Minsoo Hahn [3] and Jinsul Kim [1,*]

[1] Department of ICT Convergence System Engineering, Chonnam National University, Gwangju 61186, Korea; osw0788@gmail.com (S.O.); chungkm1250@gmail.com (K.C.); ibal2010@yahoo.com (I.A);
[2] Graduate School of Data Science, Chonnam National University, Gwangju 61186, Korea; stwum@jnu.ac.kr
[3] Astana IT University, Astana, Kazakhstan; m.hahn@astanait.edu.kz
* Correspondence: jsworld@jnu.ac.kr; Tel.: +82-62-530-1808

Abstract: In the trend of transforming existing systems and services into non-face-to-face models, the concert industry is also showing movements toward transitioning to virtual formats. Physical concerts in the real world require venues that can accommodate hundreds to tens of thousands of spectators, but non-face-to-face methods that can accommodate large audiences face various limitations. Moreover, to elevate the satisfaction level of virtual concert attendees to that of real-world concerts, it's important to implement interaction between performers and audiences. Modern metaverse platforms apply cutting-edge network technologies to accommodate numerous users within a single channel. Many researchers are adopting network technologies such as SDN (Software Defined Networking) and CDN (Content Delivery Network) to set up a virtual concert that can accommodate large audiences. In this paper, we propose a network framework to be designed for the composition of virtual concerts. In particular, we separate a channel dedicated to interaction in order to provide an immersive experience of exchanging interactions between performers and audiences. As massive audiences transmitting interaction data to the performer in a 1:N format can lead to problems with acceptance and latency, this study introduces a concept of a channel form called 'Zone' and proposes an interaction data channel network framework that does not compromise immersion. The proposed framework supports tasks for effectively transmitting interaction data using network technologies for metaverse platforms like XDN and clustering algorithms like fuzzy c-means. We also suggest a CDN-based architecture that can ensure low latency for performers to transmit interaction data to the audience.

Keywords: metaverse; multiuser ivrtual environments; experience delivery network; content delivery network; cloud edge

1. Introduction

As global pandemics limit outdoor activities, various arts and cultural services and systems are attempting to operate non-face-to-face. In particular, metaverse platforms, which offer immersive environments to overcome the lack of physical presence, a limitation of online contexts, are garnering significant attention. To provide immersive experiences, 3D metaverse games like Roblox and Minecraft are utilized [1–4]. Recently, with the development of various metaverse games and platforms, many planners and researchers are actively conducting studies and efforts to implement the metaverse. Likewise in the concert industry, musicians are starting to host virtual concert events using various metaverse platforms. This trend of major artists participating in virtual concert services is accelerating [5–7]. For instance, startups providing virtual concert platforms, such as WAVE, where avatars perform live concerts, have attracted investments of up to 30 million dollars. Backed by enormous capital, these industries are expanding their content, producing concerts with globally renowned artists like Justin Bieber.

The metaverse platform industry is researching and integrating various technologies, including volumetric technology, to provide users with immersive experiences [8]. Volumetric technology captures human motion in a green screen studio equipped with hundreds of 4K-level cameras, creating a 360-degree stereoscopic image. Human objects in a virtual space created using volumetric technology are typically referred to as digital humans. While volumetric content has the advantage of enabling a lifelike avatar representation of performers compared to traditional virtual concert content, interaction between performers and users has been limited to a few dozen simultaneous connections within the metaverse channel due to real-time concurrent user restrictions. Interaction in concerts is a means for performers and audiences to communicate bi-directionally. Implementing an interactive environment in the virtual concert world of the metaverse platform can enhance QoS for both parties. However, when the audience and performers exchange interaction data during a virtual concert, the amount of traffic that needs to be processed in parallel increases exponentially with the number of audience members.

To construct a metaverse network for virtual concerts, many providers create concert venue worlds and environments using game engines. To efficiently create a metaverse world, world creators use basic network functions provided by the game engine. However, building a virtual concert with a single function results in handling various types of data occurring in the concert venue world within a single network framework. In particular, interaction data exchanged between performers and users require an end-to-end network, potentially escalating the overall network framework's requirements. Moreover, the performer values the visual experience of receiving interactions from the audience more than the specificity of the transmitting party, so it's not necessary to implement 1:N type communication depending on the number of audience members. For instance, if 1,000 audience members enter and all 1,000 interaction data they send must be processed, real-time concert can be compromised due to exponentially increased network traffic.

Hence, in this paper, we employ bidirectional network channels to segregate the network channel transmitting interactions. This study proposes a framework where the audience in the virtual space is divided into n zones to receive interaction data from a massive audience, with each zone summarizing its interaction data for transmission to the performer. In Chapter 2, we explain related technologies, and Chapter 3 outlines the CDN-based network framework that we aim to design in this paper. Chapter 4 concludes this research, detailing future experimental plans and issues to address.

2. Related Researches

In this section, we explain the underlying approach for constructing the network framework, and the background and algorithms of the approach used in this paper to process interaction data from the audience.

2.1. SDN(Software Defiend Networking)

Software Defined Networking (SDN) is a concept of defining and controlling networks through software, a framework designed and built to manage traffic handling methods and various functions in a centrally controlled system by separating data and control domains. This concept was devised to overcome the complexity and flexibility limits of traditional hardware-centric network structures and to implement user-centric networks. It started with a paper published in August 2007 by Stanford University, proposing a way to facilitate network management and enhance security [9]. SDN provides an open interface that enables the development of software capable of controlling connectivity provided by a set of network resources and network traffic flow, along with potential inspection and modification of traffic performed in the network. As illustrated in Figure 1 below, SDN is composed of three layers, emphasizing four features: the central separation of the Infrastructure layer and Control layer, a central controller, an open interface between the Control layer devices and Infrastructure layer devices, and network programming by external applications [10].

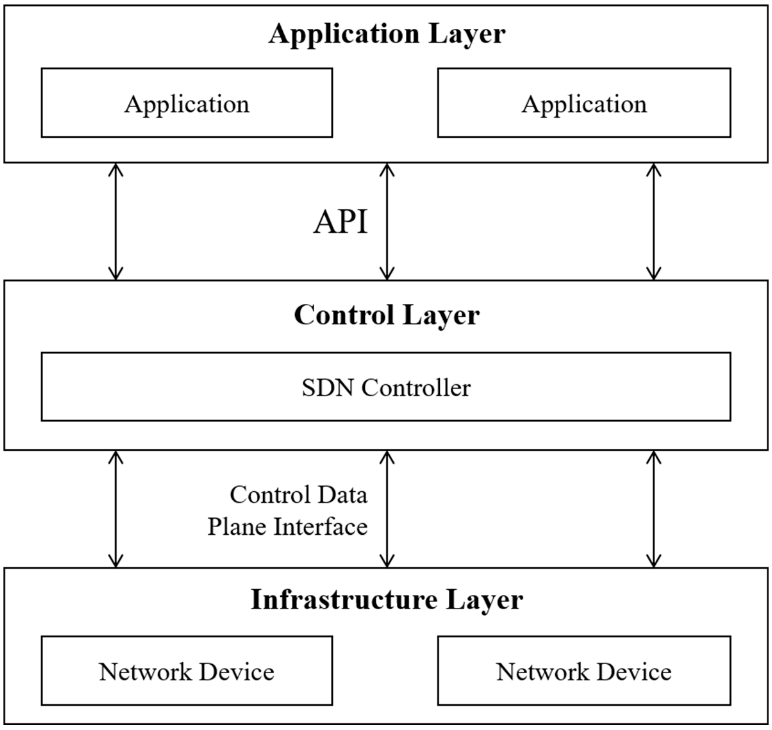


Figure 1. Architecture of SDN(Software Defined Networking).

The Application Layer is where various programs utilizing SDN exist. It enables the use of various functions across the network through the APIs provided by the control layer. These APIs are used for interaction between the Application Layer and the Control Layer and can be used by the Application Layer to receive information related to the Control Layer. The Control Layer is the layer that centrally handles the flow of network data, responsible for determining the routing and forwarding methods for specific packets. It controls the equipment of the data layer through the Control Data Plane Interface and provides an interface abstracting the network's functions to the Application Layer through APIs. The Infrastructure Layer is a layer composed of network devices capable of packet forwarding and processing. It houses the flow table, which is responsible for the actual transmission of data. It operates by receiving routing information from the Control Layer through the Control Data Plane Interface. These layers are controlled through interfaces. Since traditional network devices are single-box, there has been difficulty in changing or manipulating the equipment. Therefore, SDN is designed so that control is possible as long as one knows the standard interface, enabling escape from hardware vendors.

2.2. CDN(Contents Delivery Network)

The Content Delivery Network (CDN) was proposed to enhance the traditional content delivery method, which relied on a single large-scale server and could not guarantee quality while providing content to all users via the internet [11,12]. CDN provides multiple servers responsible for numerous users to prevent them from receiving content through unreliable internet bandwidth. Figure 2 below illustrates the servers constituting a CDN. Among the components of CDN, the Staging Server receives content from the Content Provider and stores content to be distributed to other servers. All content distribution begins at the Staging Server, and eventually arrives at the Edge Server, which is the server that delivers content to users and serves as the target for content distribution.

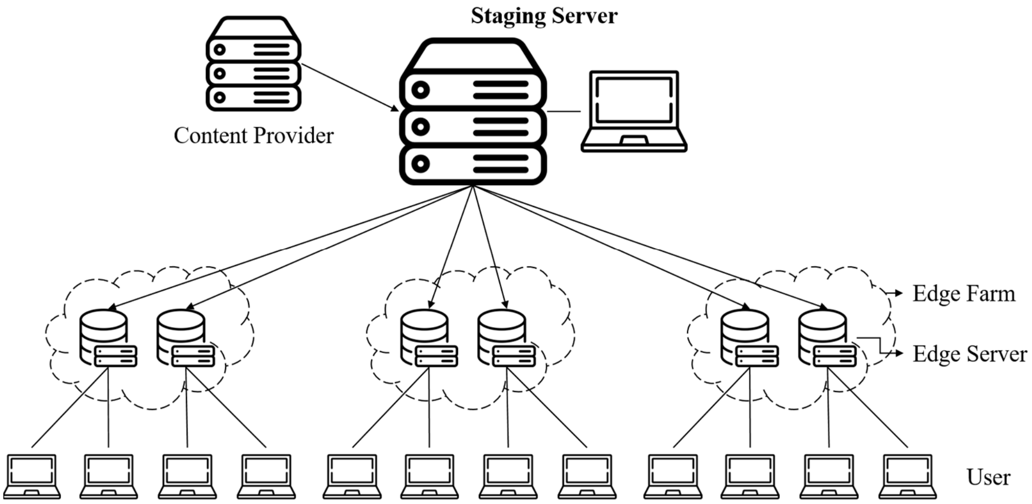


Figure 2. Architecture of CDN Servers.

A typical CDN is a distributed network of servers that can efficiently deliver content to users. It minimizes latency by storing cached content on edge servers located near the point-of-presence (POP) of the end user. Through the application of edge computing, it enables proactive responses based on the situational awareness of the end users, depending on the type of service. In metaverse virtual concerts, interactions between the performer and user, as well as between users, are crucial for creating a sense of immersion and realism. Consequently, securing a bidirectional data channel is essential. However, there are limitations with the current CDN structure to facilitate this [13].

2.3. XDN(eXperience Delivery Network)

Services that support mixed reality (XR), such as virtual concerts, provide immersive content to users. Delivering immersive content in a CDN environment faces the following challenges:

- Expansion to accommodate massive user-generated content is required.
- Interactive experiences, real-time delivery of content composition, and content synchronization are necessary.
- Creation of a 360-degree view, addition of text and image overlays, and real-time collection, processing, and distribution of content via wireless networks are required.

Thus, to address these issues, the Experience Delivery Network (XDN), which overcomes the limitations of the conventional CDN, has been newly proposed. XDN, an evolution of CDN, enables immersive content experiences including XR and metaverse through the application of 5G and edge computing [14]. Various microservices for collecting, controlling, and distributing media in XDN can be seen in Figure 3 below. The Media Control Function (MCF) provides functions to control authentication, session management, and clustering of virtual concert audiences and Zones, processed at the edge close to content providers and consumers. The Media Distribution Function (MDF) handles the distribution of Zone-specific media based on OMAF tile-based viewport-dependent streaming. The Media Ingestion Function (MIF) deals with the ingestion of various types of media depending on the environment of the performer and audience.

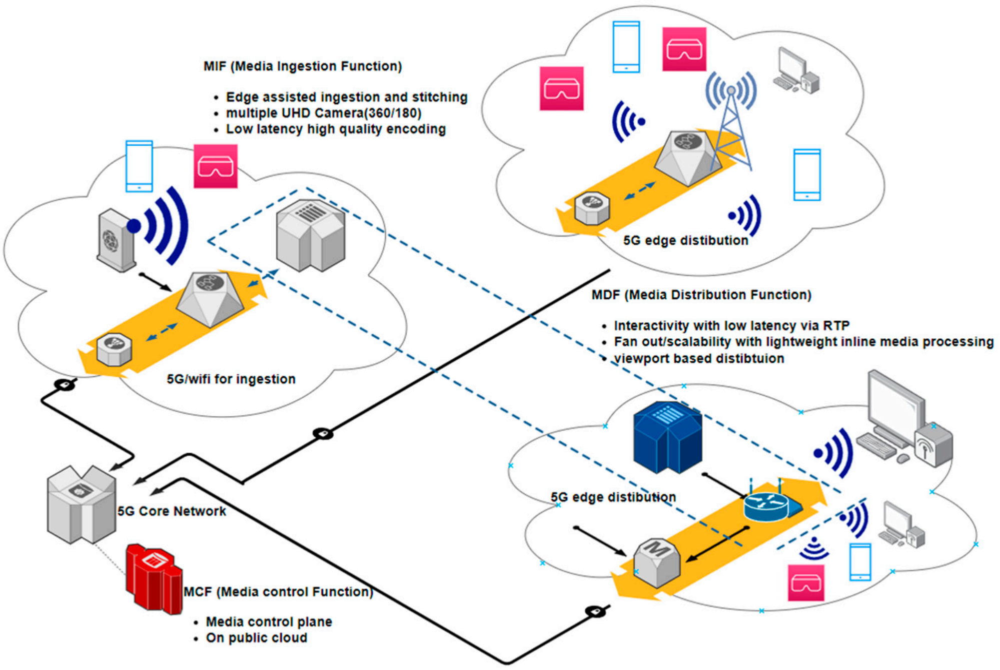
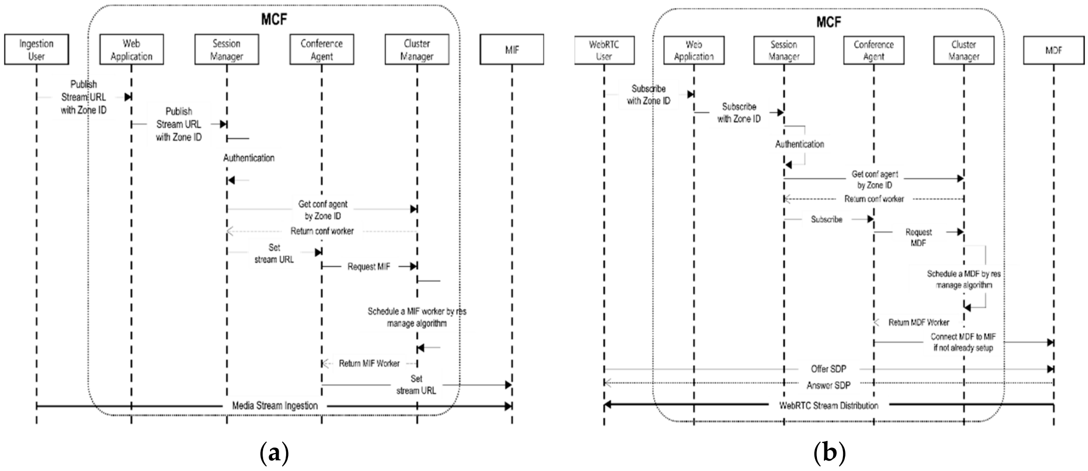

MIF (Media Ingestion Function)
Edge assisted ingestion and stitching
multiple UHD Camera(360/180)
Low latency high quality encoding
5G/wifi for ingestion
5G edge distibution
MDF (Media Distribution Function)
Interactivity with low latency via RTP
Fan out/scalability with lightweight inline media processing
viewport based distibtuion
5G edge distibution
5G Core Network
MCF (Media control Function)
Media control plane
On public cloud

Figure 3. Schematic of XDN(Experience Delivery Network).

Figures 4 illustrate the sequence diagrams of microservices typically required when constructing an XDN. These examples represent operations performed in a zone network architecture within a WebXR-based system that supports XDN. WebXR is a framework that provides XR (Extended Reality) experiences based on WebRTC.



(**a**) (**b**)

Figure 4. (a) Sequence diagram of MCF and MIF in XDN system based on WebXR (WebRTC), (b) Sequence diagram of MCF and MDF in XDN system based on WebXR (WebRTC).

CDN architectures can reduce bandwidth consumption and latency, and provide scalability needed for handling abnormal traffic loads, improving server performance. Therefore, in order to design a real-time adaptive XDN with ultra-low latency, it is necessary to design a robust network architecture for the CDN.

2.4. Fuzzy C-means Algorithm

In this paper, we employ clustering algorithms to segment a large audience within a virtual concert based on certain criteria. Specifically, we adopt the Fuzzy c-means algorithm to assign an appropriate number of Zones based on the positional coordinates of the audience. Dunn has devised a method using fuzzy segmentation by expanding the k-means al-gorithm and presenting the objective function as shown in Equation (1) below [15,16].

$$\begin{gathered}\min J_2(U,V) = \sum_{i=1}^{c}\sum_{k=1}^{n}(\mu_{ik})^2\|x_k - v_i\|^2 \\ u_{ik} \in \{0,1\}\forall i,k, \sum\nolimits_{i=1,c} u_{ik} = 1\,, \forall k. \\ 0 < \sum\nolimits_{k=1,n} \mu_{ik} < n, \forall i.\end{gathered} \tag{1}$$

$U = [\mu_{ik}]$ represents the matrix of a fuzzy c-partition, and Equation 2 indicates the objective function representing the constraint of the fuzzy c-partition. Since each x_k can have a rational number between 0 and 1, it can belong to more than one cluster along with its membership. Bezdek generalized $J_2(U,V)$ without any restrictions.

$$\min J_m(U,V) = \sum_{i=1}^{c}\sum_{k=1}^{n}(\mu_{ik})^m\|x_k - v_i\|^2 \tag{2}$$

Since the fuzzy c-means algorithm specifies the number of clusters and weighting coefficients and has a different optimization procedure from the k-means algorithm, it offers superior results in various experiments compared to the k-means algorithm [17–19]. This algorithm has been employed for clustering in a wide range of industries, and in this study, we utilized it to cluster audiences in the virtual concert environment [20–22].

3. Proposed Framework

To design the network structure constituting the virtual concert, an analysis of requirements specialized for various content and services must be first conducted. In particular, for the communication of interaction data between performers and the audience, network designers need to perform tasks to derive requirements for interactions and network functions. We assume an environment capable of constituting a virtual concert world where thousands can participate simultaneously, and ensuring free interaction between performers and users within the concert hall. As shown in Table 1, network functions for interactions can be derived.

Table 1. Network function fo interaction data based on requirements [13].

Requirements	Network Function
User motion information (joint variations)	Selection of interactions at each step and definition of data structures are necessary.
Emotional expressions through emoticons	
Text message (chatting)	Joint variation information in 3D vector from is delivered through data streaming.
Voice	

3.1. Metaverse Virtual Concert Network Design

Before designing a network for the communication of interaction data, the network for the overall virtual concert must first be designed. The metaverse virtual concert platform allows subscribers to access the platform and participate in their desired virtual concert service channel according to role distinction, managed through subscriber management and concert information management. To provide scalable server operations and variable load balancing features, the virtual concert platform is loaded onto a cloud-based server. This allows for an increase in the number of physical subscriber servers as the number of simultaneous users increases. As shown in Figure 5, pre-

doi:10.20944/preprints202306.1941.v1

defined fixed stage-related assets in the planning process for the virtual concert hall and stage equipment are downloaded before the real-time concert, and during the real-time concert, only real-time data provided from the performers are streamed.

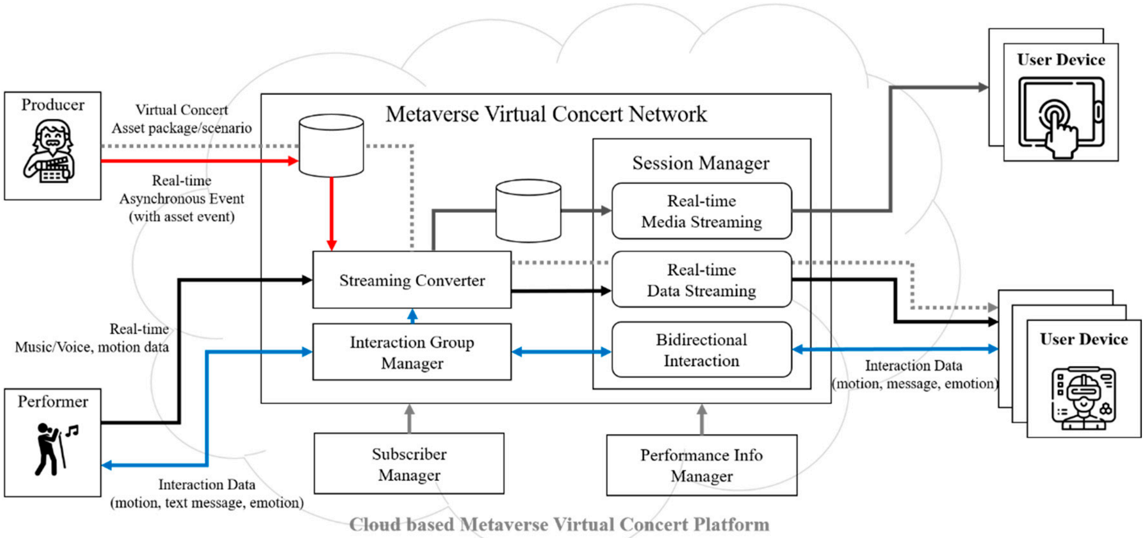


Figure 5. Metaverse Virtual Concert Network Design [13].

A virtual concert takes the form of users connecting to a virtual channel, and this virtual channel can vary dynamically. The session manager manages and controls the physical channels within the virtual channel, and the producer takes on various roles according to events that occur asynchronously during the concert. The interaction data discussed in this paper is created and delivered bidirectionally between the performer and the audience. The virtual concert network can be configured to selectively provide only real-time media streaming for user devices that have difficulty generating motion data.

3.2. Network Framework Design to Communicate Interaction in Virtual Concert

The network framework proposed in this paper represents the biggest difference between the network configuration of the existing metaverse-based virtual concert system and the network configuration described in 3-1. To solve the problem of exponentially increasing the number of users accessing the virtual concert channel as the size of the existing virtual concert users is expanded to a large scale, we applied virtual channelization technology. In an environment where the number of channels changes variably, the bidirectional interaction data channel was separated to ensure that interaction data can be well synchronized and represented in the virtual concert channel. Figure 6 below proposes a method for sending data, including information to identify the interaction target, for the audience to transmit interaction data to the performer.

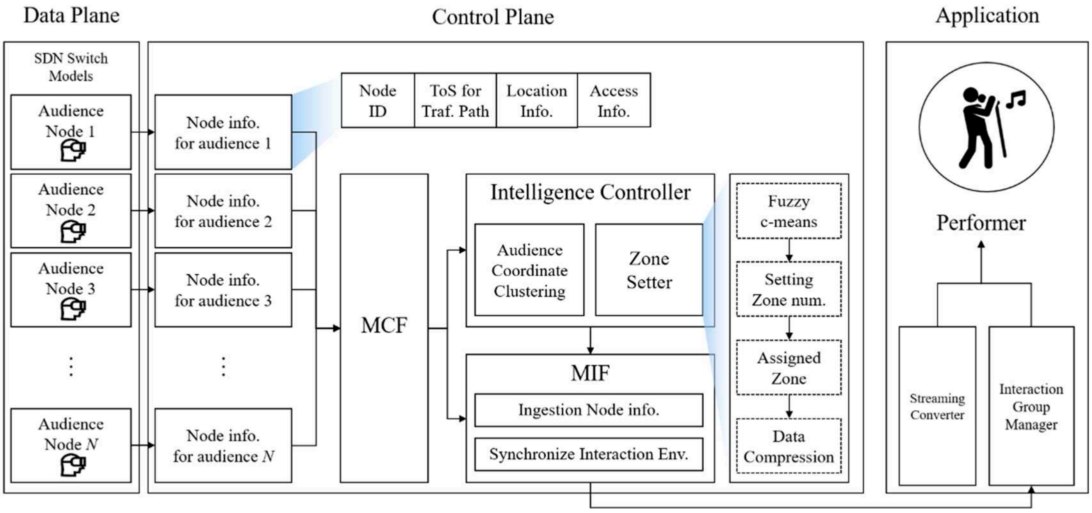


Figure 6. Schematic of XDN(Experience Delivery Network).

Considering the virtual channel and the dynamically changing network environment, we adopted an SDN (Software Defined Networking) type of framework. The Data Plane is composed of node models that receive information from the audience in the virtual concert and perform SDN switch functions. The node and the process models inside the node send their information to the Control Plane, which is then delivered to the MCF (Media Control Function) microservice of the XDN for the preparation of authenticating audience information and assigning Zones. Some studies receive and store the node's information in the form of a pointer using a shared memory interface [23], but this study does not deal with specific packet processing methods for synchronizing other network depths outside of the interaction channel. The main information of the Node received from the Control Plane includes the Node's Id, ToS for Traffic Path information, Location in the virtual concert, and Access environment information. Before the virtual concert begins, if all the audience has entered, the MCF sends the node information to the Intelligence Controller.

Once the MCF receives information about the audience that has entered the virtual concert, it sends the Node's packet information to the Intelligence Controller to create Zones and assign audiences to each Zone. First, 2D (which could also be 3D depending on the nature of the world) coordinates are generated through the audience location coordinates within the virtual concert world, and clustering is carried out in the Zone Setter. Various metrics including the Elbow method are used to determine the appropriate number of clusters c, and the Fuzzy c-means algorithm is performed accordingly to generate the number of Zones and centroids. Once the coordinates of the Centroid within the virtual concert are determined, the Zone Setter performs clustering and assigns Zones to each Node. Then, the MIF ingests the Node information, including Zone information, after compressing it.

After the concert starts and interaction data is sent from the Node, the process mentioned above is followed until the MIF accumulates interaction data within Zones. A specific time interval is set (generally between 500ms to 3000ms), and during this time, the accumulated interaction data types are categorized, and the transmitted volume for each type is measured. This information is then sent entirely to the Interaction Group Manager. The types of interactions can be configured as shown in Table 2, but the virtual concert producer has the discretion to specify more diverse interaction data types.

Table 2. Type of interaction data when use in virtual concert.

Kind of Interaction	Details (0 … N)
Gesture	Jump Hurray Waving arms
Emoticon	Smile Shoot Heart Emoticon Nyah Weeping
Text(Default Message)	“Hello!” “Nice to meet you.” “I love you.” “Very good.” “Wonderful!”

The interaction data from the Node is sent to the Performer, and to avoid affecting the performance provided in the form of streaming, the interaction transmission channel is kept separate. Furthermore, to synchronize the streaming and interaction, the MIF performs the task of synchronizing both channels. If the performer sends an interaction to the audiences, they bypass separate MIF or Intelligence Controller and directly transmit interaction data to all Nodes. Future research will design a task where the performer selects a zone and sends interactions only to that zone. As shown in Figure 7, not only does the performer provide performances through the streaming converter as a central server in the form of a CDN, but it also sends interaction data to each Edge Farm, supporting low-latency communication.

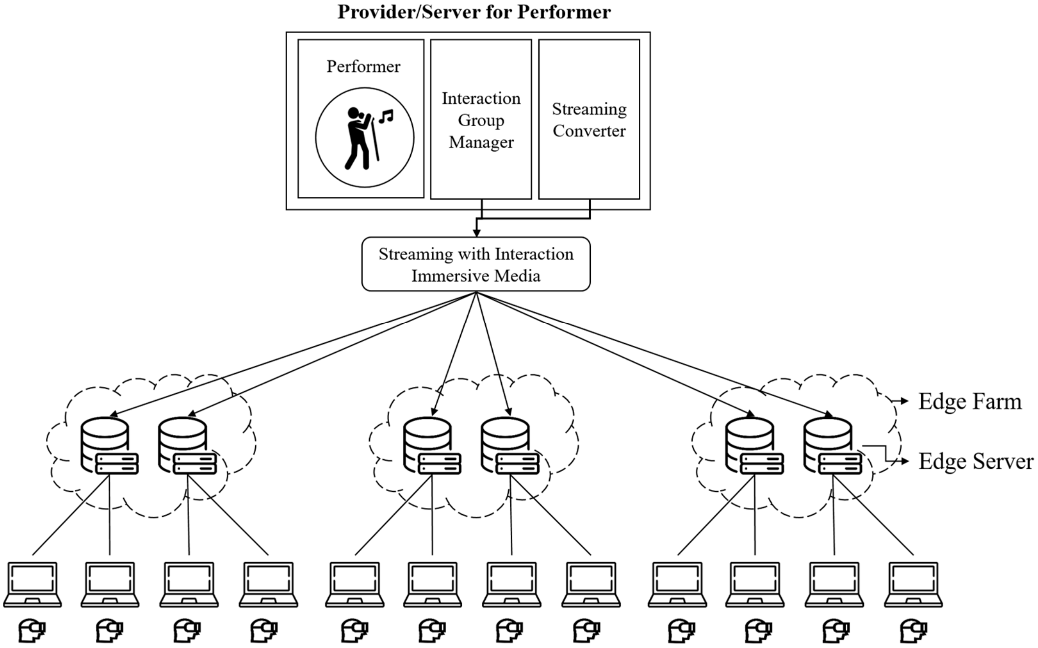

Provider/Server for Performer
Performer
Interaction Group Manager
Streaming Converter
Streaming with Interaction Immersive Media
Edge Farm
Edge Server

Figure 7. Architecture of SDN(Software Defined Networking).

Performers can configure the following CDN-based edge servers to transmit interaction data to large audiences. Edge servers are critical components in the proposed network structure to reduce latency and to manage traffic efficiently. They are strategically placed closer to the nodes (audience) and the performer, which allows data to travel shorter distances resulting in faster data delivery. Here's a brief conceptual overview:

- **Central Server (Performer)**: This is where the live performance happens. The performance data is streamed in real-time from this location. Interaction data from the performer, if any, is also sent from here.
- **Streaming Converter**: The streaming converter takes the raw performance data (motion, audio, voice data) from the central server (Performer) and converts it into a format suitable for transmission over the network. It might involve compression or other methods to ensure efficient data transmission.
- **Edge Servers (Edge Farm)**: These are servers placed closer to the end users (Nodes). They receive the performance stream from the central server via the streaming converter, and then relay this data to the end users. Similarly, interaction data from the users are collected and sent back to the central server through these edge servers.
- **Nodes (Audience)**: These are the end users or audience members in the virtual concert. They send interaction data and receive performance data through the edge servers.

  This setup ensures that data doesn't have to travel long distances all the way from the performer to each individual user and vice versa. Instead, the data travels to nearby edge servers, which can relay it to the users or back to the performer as required, significantly reducing latency and enabling real-time interactions. However, since the above method needs to consider scalability, it is an option worth designing if the virtual performance platform is configured based on the cloud as shown in Figure 4.

4. Conclusions

This paper proposes a network framework that separates data channels from the existing framework to transmit interaction data between performers and audiences in virtual concerts based on the metaverse platform. In order to provide an immersive experience to the audience, we designed a network using XDN (Experience Delivery Network) based microservices. Additionally, we presented a task that can visually experience interactions by zone, enabling performers to efficiently receive interactions from a large audience and immerse themselves in virtual reality. This is done by constructing zones based on the network packet information of the audience and the coordinates in virtual reality. In this process, we performed clustering using the Fuzzy c-means algorithm to divide the zones according to the varying number of audiences. Furthermore, we established the framework of network design at the SDN (Software-Defined Networking) level and proposed an architecture that can communicate interaction data of virtual concerts in each plane.

This research is significant in that it precisely designs a network framework dedicated to interactions, which can be incorporated into the network architecture based on the metaverse platform proposed for virtual concerts. Existing studies have operated virtual concerts using game engines that support a single network or metaverse network architectures that provide various services in an integrated manner. However, by establishing a network architecture specialized for the concert industry as a separate category, we approached dealing with large audiences from a network perspective and proposed a solution for it.

As future research, we plan to conduct simulation experiments to enhance the network architecture for transmitting interaction data in a virtual concert architecture configured in a cloud environment. We will assume the total number of audiences for the simulation, configure edge servers in the form of CDN (Content Delivery Network) accordingly, and design a new algorithm capable of handling scalable multi-interaction data by programming the microservices of XDN (Experience Delivery Network). Furthermore, we will expand the traffic, data volume, and types generated through example virtual concerts, and intelligently upgrade the network framework by calculating cases that can occur during the concert.

Author Contributions: Conceptualization, Ibrahim Aliyu; Project administration, Tai-Won Um; Supervision, Jinsul Kim; Writing – original draft, Sangwon Oh and Kwangmoo Chung; Writing – review & editing, Jinsul Kim and Minsoo Hahn. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT)(NRF-2021R1I1A3060565) and this research was also supported by the MSIT(Ministry of Science and ICT), Korea, under the Innovative Human Resource Development for Local Intellectualization support program(IITP-2022-RS-2022-00156287) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rospigliosi, P. A. Metaverse or Simulacra? Roblox, Minecraft, Meta and the turn to virtual reality for education, socialisation and work. *Interactive Learning Environments* **2022**. *30(1)*, 1-3.
2. Duan, H.; Li, J.; Fan, S.; Lin, Z.; Wu, X.; Cai, W. Metaverse for social good: A university campus prototype. In Proceedings of the 29th ACM international conference on multimedia, New York, United States, 20-24 October 2021; pp. 153-161.
3. Jeon, H.J.; Youn, H.C.; Ko, S.M.; Kim, T. H. Blockchain and AI Meet in the Metaverse. In *Advances in the Convergence of Blockchain and Artificial Intelligence*; IntechOpen: London, UK, 2022; pp. 73-82.
4. Han, J.; Heo, J.; You, E. Analysis of metaverse platform as a new play culture: Focusing on roblox and zepeto. In International Conference on Human-centered Artificial Intelligence, Da-Nang, Vietnam, 28-29 October 2021; pp. 1-10.
5. Koh, B.S.; Kim, M. Metaverse-based immersive content R&D support project trend. *Broadcast Media* **2022**, *27(1)*, 21-26.
6. Kim, K.J. The evolution of the real and virtual world through the case of the metaverse, *Broadcast Media* **2021**, *26(3)*, 10-19.
7. Lim, H.; Cho, J. The Consilience between Metaverse and Performing Arts : Focusing on the Remediation Theory, *Journal of the Korea Entertainment Industry Association* **2022**, *16(3)*, 107-124.
8. Mystakidis, S. Metaverse. *Encyclopedia* **2022**, 2(1), 486-497

9. Park, S.; Jang, I.; Seo, D.; Lee, J. About SDN/NFV, the new paradigm of future networks, *Information and Communications Magazine* **2015**, *32(7)*, 82-92.
10. Sezer, S.; Scott-Hayward, S.; Chouhan, P.K.; Fraser, B.; Lake, D.; Finnegan, J.; Viljoen, N.; Miller, M.; Rao, N. Are we ready for SDN? Implementation challenges for software-defined networks. *IEEE Communications Magazine* **2013**, *51(7)*, 36-43.
11. Buyya, R.; Pathan, M.; Vakali, A. *Content delivery networks*, Springer: Berlin/Heidelberg, Germany, 2008; pp. 4-15.
12. Pathan, A.M.K.; Buyya, R. A Taxonomy and Survey of Content Delivery Networks, Grid Computing and Distributed Systems Laboratory, University of Melbourne, Melbourne, **2007**. Available online: http://www.buyya.com/gridbus/cdn/reports/CDNTaxonomy.pdf (accessed on 10 May 2023).
13. Yu, C.R.; Gil, Y.H.; Jeong, I.K. A Study on the metaverse network design for accommodating large-scale virtual performance users, In Proceedings of Symposium of the Korean Institute of communications and Information Sciences, Gyeongju, Korea, 16-18 November 2022; pp. 749-751.
14. Shen, G.; Dai, J.; Moustafa, H.; Zhai, L. 5g and edge computing enabling experience delivery network (xdn) for immersive media. In Proceedings of the *2021 IEEE 22nd International Conference on High Performance Switching and Routing (HPSR)*, Paris, France, 7-10 June 2021; pp. 1-7.
15. Dunn, J.C. Well-separated clusters and optimal fuzzy partitions. *J. Cybern.* **1974**, *4*, 95–104.
16. Bezdek, J.C.; Dunn, J.C. Optimal fuzzy partitions: A heuristic for estimating the parameters in a mixture of normal distributions. *IEEE Trans. Comput.* **1975**, *100*, 835–838.
17. Hall, L.O.; Ozyurt, I.B.; Bezdek, J.C. Clustering with a genetically optimized approach. *IEEE Trans. Evol. Comput.* **1999**, *3*, 103–112.
18. Krishnapuram, R.; Kim, J. A note on the Gustafson-Kessel and adaptive fuzzy clustering algorithms. *IEEE Trans. Fuzzy Syst.* **1999**, *7*, 453–461.
19. Bezdek, J.C. *Pattern Recognition with Fuzzy Objective Function Algorithms*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013.
20. Ko, S.; Yoon, U.N.; Alikhanov, J.; Jo, G.S. Improved CS-RANSAC Algorithm Using K-Means Clustering. *KIPS Trans. Softw. Data Eng.* **2017**, *6*, 315–320.
21. Chung, J. Parallel k-Modes Algorithm for Spark Framework. *KIPS Trans. Softw. Data Eng.* **2017**, *6*, 487–492.
22. Liu, Y.; Wang, H.; Duan, T.; Chen, J.; Chao, H. Incremental fuzzy clustering based on a fuzzy scatter matrix. *J. Inf. Process. Syst.* **2019**, *15*, 359–373.
23. Lee, C.W.; Lee, G.M.; Lee, H.; Roh, B. Design and Implementation of Riverbed Modeler M&S Framework for Multi-Layered Future Tactical Networks with Intelligent SDN Control Architecture. *The Journal of Korean Institute of Communications and Information Sciences* **2022**, *47(8)*, 1195-1204.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.