COMPARISON OF OPERATING COSTS BETWEEN STREAMING USING ELISA VIIHDE'S CDN AND GOOGLE'S MEDIA CDN

Lappeenranta-Lahti University of Technology LUT

School of Engineering Science

Software Engineering

2023

Hesham Ahmed

Examiners: Professor Jari Porras (LUT University)

Professor Henry Muccini (University of L'Aquila),

Professor Patricia Lago (Vrije Universiteit Amsterdam)

SE4GD
Software Engineers for Green Deal

With the support of the Erasmus+ Programme of the European Union

This thesis has been accepted by partner institutions of the consortium (619839-EPP-1-2020-1-FI-EPPKA1-JMD-MOB).

Successful defence of this thesis is obligatory for graduation with the following national diplomas:

- Master of Computer Science (University of L'Aquila)
- Master of Science in Technology (LUT University)
- Master of Computer Science (Vrije Universiteit Amsterdam)

ABSTRACT

Lappeenranta-Lahti University of Technology LUT

School of Engineering Science

Software Engineering

Hesham Ahmed

COMPARISON OF OPERATING COSTS BETWEEN STREAMING USING ELISA VIIHDE'S CDN AND GOOGLE'S MEDIA CDN

Master's thesis

2023

66 pages, 8 figures, 16 tables and 2 appendices

Examiners: Professor Jari Porras (LUT University),
Professor Henry Muccini (L'Aquila University),
Professor Patricia Lago (Vrije Universiteit Amsterdam)

Keywords: Video streaming, CDN, Total Cost of Ownership

This study **aims** to compare the operating costs of Elisa Viihde's existing on-premises streaming infrastructure against an on-cloud streaming solution that utilizes Google's Media CDN. The **rationale** behind this work stems from the need to optimize the streaming process and reduce unnecessary costs. The **Research process** followed to reach this goal started with studying relevant literature and studying the system of Elisa Viihde to identify the key components and cost-driving factors. Then, Elisa Viihde's system was simulated on Google Cloud. The cost comparison is based on the cost associated with utilizing both streaming solutions under a specific data traffic. On-cloud simulation costs were calculated using Google cloud forecasting mechanisms, while on-premises costs were estimated based on hardware, electricity, real estate, and labor. Total cost of ownership (TCO) over 3 years was calculated to be 14.5 million EUR for streaming on Google Cloud versus 1.2 million EUR for the current on-premises system, with Media CDN and hardware ownership being the largest contributors. The **results** suggest that on-premises is more economical. This initial quantitative cost comparison provides a baseline to inform decision-making, but more research accounting for dynamic pricing, improving simulations with real usage data, and weighing qualitative factors will enable a more holistic evaluation.

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my thesis supervisors Professor Jari Porras, and from Elisa, Santeri Auvinen, and Sami Volotinen, for their guidance, feedback, and support throughout the process of researching and writing this thesis.

I am thankful to my family for their love, prayers, and sacrifices for educating and preparing me for my future. Their unconditional love and support have always been my strength.

I would like to thank the SE4GD program committee for this opportunity and I wish to acknowledge my dear friends and colleagues from SE4GD cohort 1 with whom I studied, lived, and traveled for the duration of this program.

SYMBOLS AND ABBREVIATIONS

Abbreviations

AI Artificial Intelligence

API Application Programming Interface

BGP Border Gateway Protocol

CAGR Compound Annual Growth Rate

CAPEX Capital Expenditure

CDN Content Delivery Network

CLI Command-Line Interface

CSR Corporate Social Responsibility

CUD Committed Use Discount

DRM Digital Rights Management

ECC Edge Cloud Computing

EPG Electronic Programme Guide

GB Gigabyte

GCP Google Cloud Platform

GHG Greenhouse Gases

GKE Google Kubernetes Engine

HLS HTTP Live Streaming

IaaS Infrastructure as a Service

ICT Information and Communication Technology

IoT Internet of Things

IRD Integrated Receiver/Decoder

ML Machine Learning

OPEX Operational Expenditure

OTT Over-the-top

PaaS Platform as a Service

QoE Quality of experience

QoS Quality of Service

RTMP Real-Time Messaging Protocol

RTP Real-Time Transport Protocol

SA Stochastic Approximation

SaaS Software as a Service

SUD Sustained use discount

TB Terabyte

TBO Total Benefits of Ownership

TCO Total cost of ownership

URL Uniform Resource Locator

VM Virtual Machine

VoD Video-on-Demand

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

The exponential growth of internet traffic and the increasing demand for high-quality content has led to the widespread adoption of CDN. CDN is a geographically distributed network of proxy servers and their data centers. CDN Consists of the following [1]:

- Origin server, which is the original server where the content is stored and maintained by the service providers.
- Edge/Non-origin/Surrogate servers are servers that cache the origin servers' content.
- Request Routers deliver the client's content request to a suitable surrogate server.
- Load balancing elements to distribute the requested content from the origin to the surrogate servers.
- An accounting mechanism provides logs and accounting information to the origin servers.

Refer to Appendix A for an illustration of the path of a user request in a CDN.

CDNs have been suggested as a means of optimizing bandwidth usage, enhancing accessibility, and ensuring accuracy through content replication [2]. They were first developed in the late 1990s to address the Internet's performance bottlenecks[3]. Currently, CDNs are responsible for delivering a significant amount of online content, which includes various types of web objects (such as text, graphics, and scripts), downloadable objects (like media files, software, and documents), applications (such as e-commerce and portals), live streaming media, on-demand streaming media, as well as social media platforms [4]. The Data volume of global content delivery network internet traffic from 2017 to 2022 has seen almost a 5 times increase from 54 Exabytes per month to 252 Exabytes [5].

Streaming services can benefit from CDN. When a user requests content, the CDN selects the server that is closest to the user and delivers the content from that server. This reduces the distance that the content needs to travel, which can improve the speed and quality of the streaming experience. CDNs can also help to reduce buffering and ensure a smooth playback experience. By caching content in servers located closer to the users, CDNs can reduce the load on the origin server and improve the reliability of the streaming service. Moreover, CDNs can help to improve scalability and handle sudden spikes in traffic. This is especially important for streaming services that experience a high volume of traffic during popular events or new releases.

With the rampant increase in traffic of CDN, an awareness of its side effects is gaining attention. One major area of concern is the impact on sustainability, as increased CDN usage can lead to higher energy consumption and electronic waste. Sustainability in Information and Communication Technology (ICT) is a term that describes a new direction in the development of technology which emphasizes the importance of taking into account the different ramifications of the usage of ICT economically, environmentally, and socially [6]. Freitag et. al. [7], in Figure 1, showed different projections to the increase in ICT Greenhouse Gases (GHG) from 2020 onwards, some predictions displaying exponential increase and some are more moderate and one showed predictions that takes into account a wide mitigating actions. This direction requires long-term thinking at the designing phase of ICT and it would require huge efforts to change the current systems to abide by the new principles.

In light of these sustainability concerns, concepts like Corporate Social Responsibility (CSR) became increasingly relevant. The year 2010 saw the publication of ISO 26000 standard [8] that outlines CSR guidelines and actions that executives should take to display social responsibility. The standard emphasizes the need for businesses to adopt precautionary measures to protect the environment, and promotes the adoption of eco-friendly information technologies to promote greater environmental responsibility in business practices. Today, companies are held to higher expectations than ever before. Investors and other stakeholders evaluate companies based on the "triple bottom line," which considers financial performance, CSR , and environmental practices. The present-day prevalence of the concept of CSR implies that firms voluntarily integrate social and environmental concerns into their operations and their interactions with stakeholders [9].

There are multiple ways that ICT companies may consider to be more sustainable. They can use renewable energy sources, optimize data centers, electricity usage, and utilize energy-efficient hardware. They can reduce waste by adopting the circular economy mindset. They can also set criteria that should be met all the way down the supply chain. They can also demonstrate social responsibility by promoting social equity and creating programs that support community development. In general, they may aim to develop a sustainable business model that focuses on long-term sustainability rather than short-term profits, by prioritizing social and environmental objectives in addition to financial goals.

For sustainability, environmental health, technical innovation, and economic viability need to be addressed together. Economic activity inevitably impacts the environment, while environmental degradation can hurt economic prospects. Advances in technology can drive new business opportunities and efficiencies that support sustainability across sectors. And sustainable economic practices are needed to invest in beneficial technologies. A balanced approach is required, where short and long-term interests are aligned.

There is an established connection between costs saving and being sustainable [10]. For our case, saving streaming costs by finding a more efficient way to stream could be through an energy-optimized flow for storing, transferring, transcoding, and applying process like encryption on the content. Saving energy have direct impact on, for example, omitting CO2 emissions.

This study aims to calculate the streaming costs associated with two video streaming flows and then compare them. The first streaming flow is the one of Elisa Viihde. The second streaming flow is on-the-cloud alternative using Google Cloud. The process of the study includes studying the process of streaming through Elisa Viihde and simulating it on the cloud with no reduction of the important processes. The objective behind this comparison is decide the cost-effective option. It could also help identify areas where resources can be optimized or scaled based on the specific requirements of the streaming service. Finally, it could build a case for specific future directions by the company.

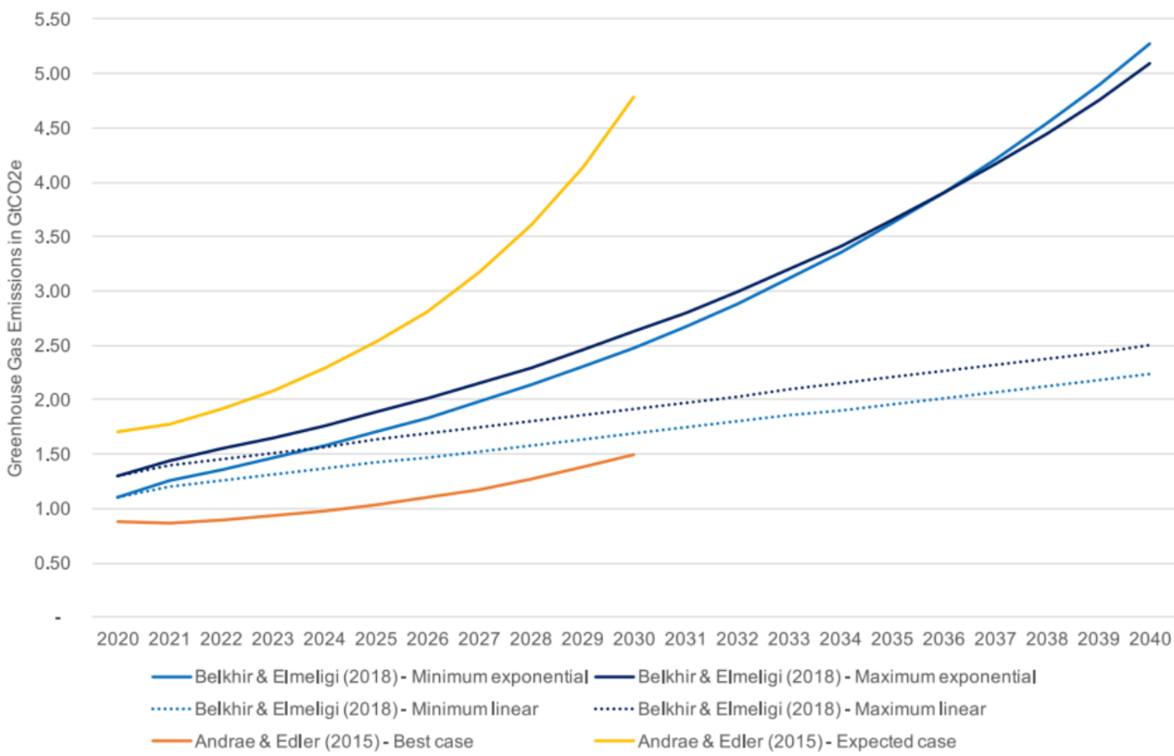

Figure 1: Different projections to the increase in ICT GHG emissions from 2020 on-wards

Structure of the thesis

The remainder of the manuscript is organized as follows:

- Chapter 2 explores the previous related works to the subject of the thesis.
- Chapter 3 states the Goal of the thesis and the process the research follows.
- Chapter 4 includes background information regarding the case study Elisa Viihde as well as its architecture.
- Chapter 5 shows the simulation set up and its modules and the technologies used.
- Chapter 6 shows the process to calculate the costs.
- Chapter 7 shows the results, an interpretation of the results, and limitation of the study.
- Chapter 8 states the conclusion of the study.

2 RELATED WORK

In this chapter, we review key literature related to Video streaming and its development over the last two decades and scalable and adaptable video streaming. Additionally, we view work on Cloud-Based video streaming rise, challenges, and future work. Finally, we review work on CDN and streaming operating costs.

2.1 Video streaming

Li et al. [11] provide a retrospective review of research progress on Internet video streaming over the past two decades. They identify three key stages in the evolution of video streaming technology:

- Client-Server Streaming (1990s-early 2000s): Early research focused on developing protocols tailored for streaming media delivery from servers to clients. Transport protocols were designed to provide QoS over best-effort networks.
- Peer-to-Peer Streaming (early 2000s-late 2000s): P2P streaming emerged to provide better scalability by having peers contribute bandwidth. However, P2P streaming faced challenges like instability and lack of incentives.
- HTTP Streaming and Cloud Computing (late 2000s-present): HTTP streaming overcomes issues of previous approaches and sees wide adoption using cloud platforms like YouTube and Netflix.

The rapid expansion of the Internet and the rising need for multimedia content has led to significant interest in streaming video over the Internet from both academia and industry. Transmitting real-time video comes with specific demands for bandwidth, delay, and loss. However, the existing best-effort Internet lacks Quality of Service (QoS) assurances for streaming video. Additionally, efficiently and flexibly managing video multicast poses challenges. These challenges include [12]:

- Video compression: Recent efforts have focused on scalable video coding to achieve high compression efficiency, flexibility, and low complexity. Combining video coding techniques can address QoS fluctuations.
- Application-layer QoS control: Includes congestion control and error control. Rate control can be source-based, receiver-based, or hybrid. Error-control mechanisms involve re-transmission, error-resilient encoding, and error concealment. Combining

- Continuous media distribution services: Research aims to build scalable and efficient infrastructures for continuous media distribution to achieve QoS and efficiency in streaming video.
- Streaming servers: Designing efficient streaming servers involves storage and retrieval of multimedia objects, scalability, and fault tolerance.
- Media synchronization: Research focuses on achieving synchronization in multimedia applications, especially addressing multi-cast video synchronization while supporting interactive functions.
- Protocols for streaming media: Standardized protocols exist for communication between clients and streaming servers. Future research topics include considering caches, supporting pause/resume operations, and enhancing protocol security.

Sun et. al [13] discussed recent topics in video streaming which is the main components of a scalable video streaming system that includes video server with storage, video encoder, video transcoder or bitstream truncator, and network protocols which enable the transport of video data to end-users.

De Cicco and Mascolo [14] presented a model of the automatic bitrate adaptation algorithm used by a major commercial adaptive streaming system. They model the control system consisting of two interacting feedback loops - one controlling the client buffer length by regulating server sending rate, and another implementing stream switching logic to select the video bitrate. They empirically deduce the control laws governing these two loops through experiments on a testbed using traffic shaping.

2.2 Cloud-Based Video Streaming

As video-streaming services continue to expand and enhance, cloud-based video has become an essential component for businesses to effectively connect with both internal and external audiences. Over-the-top (OTT) video streaming platforms like Netflix and YouTube have witnessed significant growth and have emerged as dominant players in global IP traffic. More than 50% of OTT video traffic is currently being delivered through CDNs. [15].

The global cloud based video streaming market size was USD 14.42 billion in 2022 and estimated to reach USD 37.80 billion by 2028, exhibiting a Compound Annual Growth Rate (CAGR) [1] of 17.4% from 2022 to 2028. The sudden spike in CAGR is attributable to demand returning to pre-pandemic levels once the pandemic is over [16].

[1] The compound annual growth rate (CAGR) is the annualized average rate of revenue growth between two given years, assuming growth takes place at an exponentially compounded rate.

Live streaming service providers increasingly rely on cloud services, such as Amazon AWS, to support their operations. In this context, a cloud-based architecture was introduced to enable transcoding of video content into various resolutions for live streaming. Additionally, a cost-efficient scheduling method was proposed to meet viewers' QoS requirements. [17]

Li et. al. [18], highlight open issues that face researchers and practitioners in cloud-based video streaming such as supporting interactivity, leveraging heterogeneity, building specialized video clouds, unifying Video-on-Demand (VoD) and live streaming, and improving reliability. Moreover, they Identify future research directions which include interactive streaming engines, harnessing resource heterogeneity, geo-distributed edge federations, blockchain-based security.

2.3 CDN

The research on CDNs has gained significant attention since its infancy. It can be categorized into four different but related directions [19]:

- Research on Developing theoretical models to efficiently solve the resource allocation and management problems in a CDN. For example, request routing mechanisms [20].
- Research exploring new policies for CDN infrastructure: considering there are many choices about CDN framework setup, content distribution and management, and request management approaches– there is a domain that develops new policies that manages these aspects.
- Research on developing academic CDNs: Rather than relying on a commercial CDN provider to deliver content, Web content servers are involved in an academic CDN that charges low fees. Academic CDNs are practical systems that operate in a wide-area environment, utilizing the actual Internet topology. One such academic CDN is Globule [21].
- Developing simulation test bed systems: This category involves the development of a simulation test bed system for a CDN. The purpose is to simulate a set of machines dedicated to distributing content to clients from the origin server in a reliable and efficient manner. Unlike academic CDNs, this test bed runs on a single machine in a simulated environment.

2.4 Operating costs

Numerous efforts in [22], [23], and [24] have been directed towards reducing the operating expenses of CDNs by optimizing energy consumption in clusters or data centers. The concept of "rightsizing" service capacities is central to these initiatives, aiming to conserve energy costs by scheduling idle servers into power-saving mode during periods of lower workload.

He et. al. [25] investigates the cost-Quality of experience (QoE) trade-off in cloud-based video streaming under Amazon EC2's multiple Virtual Machine (VM) pricing models. The goal is to minimize the weighted sum of VM procurement [2] cost and negative QoE through dynamic VM provisioning and procurement. The paper proposes an online algorithm called that can intelligently utilize different VM instance types based on demand and prices to achieve near optimal cost-QoE trade-off.

In the paper "Stochastic total cost of ownership optimization for video streaming services " Goudarzi [26] addresses the problem of minimizing the Total cost of ownership (TCO) for video streaming services over IP networks. The TCO includes Capital Expenditure (CAPEX) [3] and Operational Expenditure (OPEX) [4] . The author developed mathematical models to estimate the CAPEX and OPEX components associated with various elements of a video streaming system, including set top boxes, modems, servers, etc. The models account for how costs vary over time and with the number of subscribers. The overall problem is formulated as a constrained nonlinear stochastic optimization problem, due to the uncertainties in the cost models. The author proposes using a Stochastic Approximation (SA) method to find the optimal number of edge servers over time that minimizes the TCO. The results show the proposed technique can help video streaming providers improve return on investment by optimally selecting the number of deployed edge servers in the presence of uncertain ownership costs. Sensitivity analysis is performed to study the effects of key parameters on the performance of the proposed SA method.

In another work [27], Goudarzi formulated the TCO optimization problem of IPTV service as a nonlinear programming one. The solution of the proposed optimization problem can track the dynamic changes of the TCO and lead to a time-varying optimal solution.

Ciccarella et al. [28] developed modeling and simulations to quantify the performance and cost benefits of deploying Edge Cloud Computing (ECC) platforms in telecommunication networks. The goal was to evaluate if ECC can provide the high quality of service while

[2]Virtual Machine (VM) procurement refers to the process of acquiring or renting virtual machine instances from cloud service providers. VM procurement involves: 1. Deciding how many VM instances of each type (small, medium, large etc.) to rent. 2. Choosing which pricing model - on-demand, reserved or spot pricing - to use for acquiring the VMs. 3. Dynamically adjusting the number and types of VMs rented over time based on user demand, VM prices, and other factors.

[3]CAPEX pertains to the capital invested in acquiring, upgrading, or expanding physical assets with enduring value, benefiting the company over multiple accounting periods. These assets are commonly utilized in production or to facilitate business growth and operations.

[4]OPEX signifies the daily expenses a business incurs to sustain its regular operations and generate revenue. These costs are of a short-term nature and are fully deductible in the accounting period they occur.

also reducing network costs. Their results show that distributing content and applications to ECC platforms at network edges can substantially improve throughput, latency and other key performance metrics. The modeling also showed ECC can make TCO reduction, with potential savings of 25-40%, by localizing traffic and enabling caching efficiencies.

Mazrekaj et al. [29] provides an overview of pricing schemes used by cloud computing providers, analyzing factors like quality of service, usage, and depreciation that affect pricing. It outlines fixed (pay-per-use, subscription, menu), dynamic, and market-dependent (auction, bargaining) models, and compares examples like Amazon and Microsoft's use of pay-per-use and subscription pricing. Key factors affecting pricing are examined, including infrastructure and market dynamics.

2.5 Summary

The research encompasses core topics relevant to video streaming delivery, including the evolution of streaming protocols, cloud-based streaming growth, CDN optimization, and TCO reduction. Across these areas, we see common goals and challenges emerge. A primary focus is developing methods to provide reliable, high-quality streaming on best-effort networks through enhancements to compression, quality control, distribution infrastructure, and synchronization. Cloud-based services have rapidly gained prominence, enabled by HTTP streaming and CDNs, but face issues like supporting interactivity. CDN research continues to target improved resource allocation and infrastructure policies to handle growing demands.

This study aims to showcase the differences in cost, if any, between video streaming through different layouts rather than seeking to make optimization to the current layout of Elisa Viihde. Moreover, the parameters used to calculate the cost of TCO of Elisa Viihde are different from the ones presented in previous work.

3 METHODOLOGY

In this chapter, the goal and research questions of this study are stated as well as the research process that will be followed to answer the question.

3.1 Goal and Research Questions

The primary goal of this thesis is the comparison of operating costs between streaming using Elisa Viihde's CDN and Google's Media CDN. As the demand for streaming services continues to grow, service providers like Elisa Viihde must evaluate their options to optimize their infrastructure to meet their technical requirements while minimizing costs. The objective is to provide an analysis of the differences in costs between the current streaming solution of Elisa and an on-the-cloud alternative. The exact parameters of this cost will be stated in 3.2.7, specifically in the ”Calculating Elisa Viihde's cost” subsection. Table 1 displays the goal of the study.

Table 1: Goal of this study

Statement	**Description**
Analyze	Operating costs
For the purpose of	Cost evaluation
With respect to	Streaming services
From the point of view of	Streaming content providers
In the context of	CDNs used by Elisa Viihde and Google's Media CDN

Providing Elisa Viihde with a data-driven comparison of the operating costs associated with a streaming solution utilizing their CDN and another streaming solution that is using Google's Media CDN will support decision-making within the company about future direction. The rationale for this work stems from the need to optimize the streaming process and reduce unnecessary costs. In the current setup, there may be repetitive processes such as multiple transcodes of content received from the satellite. These repetitive processes not only increase the operating costs but also affect the efficiency of the streaming process. Moreover, the company can evaluate which option can scale efficiently without sacrificing performance or incurring significant additional costs.

Main RQ:

- What are the differences in operating costs between Elisa Viihde's CDN and Google's Media CDN?

Prerequisites required for the study

In order to effectively observe and analyze the existing processes of Elisa Viihde's streaming infrastructure, it will be critical to first gain permission and the necessary access to information from relevant stakeholders within the organization. This should include access to usage data and metrics from the existing platform, such as data transfer rates, as well as details on the hardware used and maintained. Additionally, account credentials for the cloud billing console will need to be obtained in order to accurately estimate simulation costs based on usage forecasts. Finally, the simulation would require access to Media CDN, as its access it restricted by Google. This information will be obtained through collaboration with relevant stakeholders within the organization.

3.2 Research Design

In this section, the steps followed by this research is stated in Table 2 and elaborated on:

Table 2: An overview the steps this study follows

Step	Procedure
Study relevant literature	1. Identify relevant search terms and databases. 2. Conduct a systematic search. 3. Obtain and review papers. 4. Synthesize and organize findings.
Study the system of Elisa Viihde	1. Conduct interviews with experts at Elisa Viihde to understand their current streaming workflow and system components. 2. Observe and analyze the streaming process and data flow. 3. Identify the key aspects and cost-driving factors of the system.
Define the settings of the simulation	1. Determine the type of content, codecs, and streaming protocols to be used. 2. Specify the simulation conditions such as geographical location of VMs. 3. Define the metrics to be monitored during the simulation, such as data transfer.
Simulate Elisa Viihde's process on the cloud	1. Select appropriate cloud infrastructure and tools. 2. Set up VMs and containers to represent system components. 3. Accurately replicate ingestion, transcoding, encryption, delivery, and caching behaviors. 4. Validate simulation through interviews with Elisa Viihde experts.
Compare both streaming solutions	1. Estimate Elisa Viihde's on-premises costs based on hardware, labor, real estate, electricity. 2. Use tools to calculate on-cloud costs based on usage of storage, Media CDN, and transcoding.

3.2.1 Literature Review

This process include conducting a literature review on CDN technologies, streaming processes, and their associated costs. As well as, identify relevant studies or industry reports that compare CDN providers and their associated costs. Table 3 shows this process.

Table 3: Literature Review Process

Action	Description	Result
Identify relevant search terms and databases	Determine the key concepts and theories to search for and select appropriate academic databases and search engines.	The main database used is IEEE Xplore. The search terms are: Video streaming, CDN, Total Cost of Ownership, Cloud Video Streaming
Conduct a systematic search	Use the identified terms to thoroughly search the selected databases and engines. Scan titles, abstracts, and keywords to identify potentially relevant papers.	After finding relevant works, backward and forward snowballing were used to find more relevant literature.
Obtain and review papers	Acquire the full text of papers that seem useful based on the search. Read through them to determine their true relevance and contribution to the topic. Take notes and keep track of citations.	19 related works were selected to represent the related work.
Synthesize and organize findings	Group reviewed papers by concepts, methods, chronology, etc. Look for common themes, major developments, contrasting perspectives, etc. that emerge.	The works were divided into 4 sections: Video streaming, Cloud Video Streaming, CDN, and Operating costs.

The results of this step are presented in Chapter 2.

3.2.2 Study the system of Elisa Viihde

This step includes conducting interviews with experts and make observations within Elisa Viihde's streaming workflow. The purpose is to understand the current streaming solution and its main processes. This is necessary to able to replicate the workflow on the cloud without making any faulty reductions to it. The process include understanding the data flow between the components and how the data changes with each phase. Gaining understanding of the system gives us the ability to spot the cost-intensive processes and therefore is important to make a valid comparison between both solutions. This step is presented in Chapter 4.

3.2.3 Define the settings of the simulation

This process include:

- Type of content: This task involves deciding on the specific type of content that will be ingested into the streaming solution. This includes the codecs, the dimensions, and protocol over which the content will be transmitted.

- Conditions: This task involves specifying the conditions under which the experiment will be conducted. For example, the geographical position of VMs instances.
- The monitored metric: the metric needs to be defined that will be recorded during the simulation. Using Cloud Monitoring, the data transferred between the CDN and the user will be tracked in gigabytes (GB) and terabytes (TB).

Clearly defining the settings of a simulation is crucial to ensuring the domain and usefulness of the study. Precisely specified settings reduce confounding variables, enable replication by others, provide context for analyzing results, and clarify the scope and bounds of the simulation. This step is presented in Chapter 5.

3.2.4 Simulate Elisa Viihde's process on the cloud

This process include:

1. Selecting appropriate cloud infrastructure: This task involves choosing the cloud platform or provider that best suits the requirements of simulating Elisa Viihde's process.
2. Tools selection: Choosing appropriate tools and technologies to represent the different stages of the process.
3. Setting up virtual machines or containers: Setting up virtual machines to represent different components of Elisa Viihde's streaming system such as the storage system and encryption of content.
4. Replicating the behavior: This includes content ingestion, transcoding, encryption, content delivery, and caching. Each step should be accurately simulated to reflect the actual behavior of the system.
5. Validating the simulation: The simulation needs to be validated to ensure that it accurately represents the behavior of Elisa Viihde's streaming system. This will be done through interviewing Elisa Viihde's experts.

This step is presented in Chapter 5.

3.2.5 Compare streaming solutions

In 1987, Gartner, an American technological research and consulting firm located in Stamford, approved the concept of TCO. This financial estimation was designed to assist companies in assessing the economic impact of projects throughout their entire life cycle.

Enterprise managers and purchasing decision-makers conduct a thorough cost analysis of various alternatives, subsequently evaluating the TCO to find the overall expenses and, in the end, identify the most economical choice in the long run.

Migration to the cloud is a trend among companies as they assess the cost-saving benefits provided by a pay-as-you-go system. It enables them to replace their on-premises equipment, reduce expenses, or transition to an agile business model by developing cloud-native applications.

Multiple methodologies and software tools exist for calculating the TCO, yet the process is imperfect. Numerous enterprises struggle to establish a unified methodology, which is insufficient as it prevents them from making purchasing decisions based on consistent information. Generally, The reliability of the analysis depends on the quality and completeness of the information used to calculate the costs.

It is a common practice to put into consideration the Total Benefits of Ownership (TBO) [1]. Consider this scenario: While one server may have a lower initial purchase price or license fees compared to a competing model, decision-makers can recognize that projected upgrades and annual service contracts would significantly diminish any apparent cost savings. In turn, although the TCO for one model might be marginally higher than that of another model, the TBO associated with it far surpasses those offered by the competing alternative.

Calculating the TCO: On-premise infrastructure

There are many factors to consider when calculating the costs associated with deciding to use on-premise infrastructure. Factors such as:

- Purchase of servers
- Software licenses
- Storage
- Security devices
- Network and IP
- Colocation in one or more data centers

These components have associated costs:

[1] The Total Benefits of Ownership (TBO) is a comprehensive assessment aimed at encapsulating the positive impacts resulting from the acquisition of a plan. It serves as an estimate encompassing all the values that can influence a business. TBO is a financial evaluation designed to assist buyers and owners in gauging the direct and indirect advantages derived from a particular product or system.

- Infrastructure design costs and time
- Renting/paying for building space and insurance
- Energy consumption and maintaining constant temperature of data centers
- Providing maintenance and technical support
- Labor costs: this includes the process of evaluating, recruiting, and employing an IT team. This team would be responsible for carrying out responsibilities such as maintenance, implementing security patches, configuring networks, upgrading software, and troubleshooting issues.
- Expenses for updates and improvements during usage
- Disposal at the end of the product's life cycle
- Inefficiency losses related to the overestimation of both the hardware and physical space required

The expenses encompass both one-time costs, such as acquiring physical equipment, as well as recurring costs, such as maintenance personnel, colocation fees, and energy consumption, which will persist throughout the whole period of utilizing the environment. Therefore, it is recommended to calculate the TCO of the duration of the lifespan of physical equipment that contribute the most to the total cost, which in our case are the servers. This means, calculating the TCO of 3-5 years. In this study, the parameters that will be considered to calculate the TCO of on-premise infrastructure are: labor, real estate, power, hardware. This will be explained in 3.2.7.

Calculating the TCO: On-Cloud infrastructure

Customers usually benefit from Software as a Service (SaaS) [2] pricing models provided by cloud computing service providers. These providers take care of the physical aspects involved in operating an on-premises computing facility, which includes procuring equipment, upgrading software, performing system maintenance and monitoring, and handling various other tasks related to managing a data center.
Referred to as "cloud tenants," customers pay a monthly subscription fee based on their chosen pricing tier and usage to utilize the cloud service technology, essentially renting it. The availability of cloud technology has enabled video service providers to explore more adapt-

[2] Software as a Service (SaaS) enables users to access and utilize cloud-based applications through the Internet. This approach offers convenient access to various applications such as email, calendaring, and office tools like Microsoft Office 365. With SaaS, users can obtain a comprehensive software solution from a cloud service provider on a flexible pay-as-you-go basis.

able revenue models.
Cloud TCO is a method employed to calculate the cumulative expenses associated with hosting, operating, integrating, securing, and managing workloads in the cloud throughout their lifespan. This encompasses charges related to the utilization of resources, including compute power, data transfer, and storage. Furthermore, it incorporates the costs associated with integrating with other cloud services, spanning from security and management tools to machine learning and AI capabilities.

Running workloads in the cloud incurs various types of expenses. These encompass, among others, the following aspects:

- Application migration (rehosting, refactoring, or redesigning).
- Utilization of infrastructure-based resources
- Costs associated with data transit between different cloud services.
- Data duplication across multiple regions or availability zones.
- Anticipated growth in future usage and workload over time.

With cloud, users don't need to forecast the possible usage in the future, they just scale as needed. Where for on-premises they need to forecast audiences and traffic three to five years in advance due to the logistics involved in server procurement and storage, installation and operation. This step is presented in Chapter 6.

3.2.6 Calculating the costs of both streaming solutions

Calculating On-the-cloud costs: In this study, this cost will be referred to as **TCO [On Cloud]**. The process includes the following:

1. Simulate Elisa Viihde's on the cloud.
2. Simulate the user traffic.
3. Billing Account Management Console, alongside Google Cloud Pricing Calculator, will be used to forecast and estimate the costs of running the simulation for a specific period of time, specifically, 1 year.
4. Record estimated expected costs.

So, **TCO [On Cloud]** is essentially the cost of using the required services (Storage, CDN, transcoding, etc...) for a specific amount of data transfer for 1 year. The user traffic is mentioned in Chapter 6. The calculation will follow the following expression:

$$\textbf{TCO [On-Cloud]} = C[MediaCDN] + C[CloudStorage] + C[LiveStreamAPI] \quad (3.1)$$

Calculating Elisa Viihde's costs: The process include the following:

1. Get information on the type and quantity of hardware used by the company.
2. Get information regarding the amount of workers that maintain the hardware.
3. Get information on the electricity consumption of the hardware.
4. Calculate the possible real estate required to accommodate the hardware.
5. Aggregate the costs associated with each of these aspects.

As mentioned in 3.2.5, calculating TCO of on-premises can be very complex as it depends on many factors. An annual estimation of the costs associated with streaming on-premises will be based on educated approximation of costs relating to every single parameter.

The calculation will follow the following expression:

$$\begin{aligned} \textbf{TCO [On Premises]} = C&[\text{labor}] + C[\text{hardware}] \\ &+ C[\text{real estate}] + C[\text{electricity}] \end{aligned} \quad (3.2)$$

Where:

- **C [labor]**: This refers to the expenditure associated with the workforce or human resources needed to manage and uphold the streaming infrastructure on-site. It encompasses salaries, benefits, and any other costs linked to personnel involved in tasks such as installation, configuration, monitoring, troubleshooting, and ongoing management of the streaming system.
- **C [hardware]**: This represents the financial outlay for owning the necessary hardware components required for the on-site streaming infrastructure. It includes expenses related to servers, storage devices, networking equipment, switches, routers, and any other hardware essential to support the streaming workflow.
- **C [real estate]**: This represents the expense incurred for renting physical space or facilities required to accommodate the on-site streaming infrastructure. The real estate cost does not include depreciation expense of a property as we assume the property will

be rented and depreciation is not applicable to rented properties since the ownership and associated depreciation expenses lie with the property owner, not the renter.

- **C [electricity]**: This parameter pertains to the cost of electricity consumed by the on-site streaming infrastructure during the specified period (in this case, one year). It incorporates the expenses of powering servers, networking equipment, cooling systems, and other electrical components essential to keep the streaming system operational.

This step is presented in Chapter 6.

3.2.7 Tools for data extraction

This subsection defines the way the data (cost) associated with both streaming solutions is estimated.

Tool 1: MUX

MUX is a company specializing in video streaming infrastructure and analytics. Mux offers a range of tools and services that help streaming companies deliver high-quality video content to their audiences, monitor video performance, and gather analytics for better insights. Some of the key features and functionalities of Mux tools include: Video Encoding and Transcoding, Video Quality Monitoring, Real-Time Analytics, Error Monitoring and Alerts, and Live Streaming.

Mux provides Elisa Viihde with detailed analytics about their videos. This data can be used to track viewer engagement, identify technical problems, Identify usage patterns and optimize video delivery. The following are essential aspects of Mux's real-time analytics:

- Video Performance: Mux tools continuously monitor various video performance metrics, including playback failures, buffering events, rebuffering ratio, startup time, and other vital indicators.
- Audience Engagement: Includes information on audience engagement, such as concurrent viewership, playbacks, and viewer behavior patterns.
- Viewer Demographics and Devices: This includes details about the geographic distribution of the audience, viewer preferences, and the devices utilized for streaming.
- Analytics Dashboards and Reports: Streaming companies can monitor key metrics, track trends, and generate customized reports to gain a comprehensive understanding of their streaming performance.

Tool 2: Billing Account Management Console

The Billing Account Management Console offered by Google Cloud serves as a tool for handling billing-related aspects within a Google Cloud account. It presents a web-based interface that enables users to observe and manage billing information, establish budgets and alerts, scrutinize usage and costs, and undertake diverse billing tasks. Key features include:

- Billing Overview: The console furnishes a comprehensive summary of the billing account, showcasing the present balance, payment history, and invoices.
- Billing Reports: Users gain access to detailed billing reports that facilitate the analysis of resource usage, costs, and patterns over time. These reports prove invaluable in comprehending and optimizing usage and expenses on the Google Cloud platform.
- Cost Forecast: The Cost Forecast report will show a chart that plots forecasted costs for the next 12 months. The chart will also show actual costs for the past 12 months, so the user can see how their costs are trending.
- Cost breakdowns: The user can view cost breakdowns by project, service, and specific resources or instances within a service. This allows the user to analyze and understand the individual cost components associated with running your streaming solution or any other workload. Google Cloud costs are affected by various components, including compute, storage, network, SQL, and serverless pricing.

Tool 3: Google Cloud Pricing Calculator

The Google Cloud Pricing Calculator is an interactive online tool offered by Google Cloud Platform (GCP) . Its primary function is to enable users to approximate the expenses associated with utilizing various GCP services. By taking into account specific usage requirements, individuals, businesses, and organizations can gain insights into the potential costs incurred when deploying applications or workloads on the Google Cloud infrastructure.

The calculator considers multiple factors, including compute resources, storage usage, networking data transfer, and other relevant GCP services, to generate accurate cost estimates. It features a user-friendly interface that allows users to input their desired resource quantities, and based on this input, the tool calculates the projected costs accordingly. Additionally, users have the flexibility to select different GCP services, configurations, and settings, empowering them to obtain more precise estimates tailored to their particular use case.

4 Elisa Viihde: Case Study

This chapter includes necessary background information regarding Elisa Viihde as well as its architecture and an example of a user request into their system.

4.1 Background

In this section, an overview on the current broadcasting scenery in Finland is provided. Moreover, the company that is used as a case study is introduced.

4.1.1 The Finnish broadcasting scene

The Finnish broadcasting scene offers a diverse range of programming options, combining public service values, commercial competition, and the growing presence of digital streaming platforms. This allows viewers and listeners to access a wide array of content, both domestic and international, catering to various interests and preferences. Broadcast layout systems are software or hardware systems used by broadcasters to manage and control the presentation of content on television channels. These systems allow broadcasters to organize and schedule programming, display graphics, and create a visually appealing and cohesive on-air presentation.

Old channels in the Finnish broadcasting industry are usually available through layout systems. The broadcasters offer streams of these channels, and they might partner with a streaming company, like Elisa Viihde, to include those channels in their own streaming service. Broadcasters have traditionally held a dominant position in the broadcasting industry and this might be a way for them to maintain this status. By partnering with broadcasters and offering their streams, streaming companies can tap into the existing infrastructure and benefit from the well-established advertising practices. This allows different customers to receive tailored commercials, maximizing the commercial revenue potential for the broadcasters. This partnership would allow the streaming company to tap into the established commercial practices of broadcasters, providing a win-win situation where both parties benefit from increased viewership and revenue opportunities.

4.1.2 Elisa Oy

Elisa Oy [30], also known as Elisa, is a Finnish telecommunications and digital service provider headquartered in Finland. It stands as one of the major telecommunications operators in the country, delivering a diverse range of services such as mobile and fixed-line telecommunications, broadband internet, digital television, and corporate IT solutions.

Having been established in 1882, Elisa boasts a rich history in the telecommunications industry. Throughout the years, it has continuously adapted and expanded its offerings to keep pace with technological advancements and evolving customer demands. Elisa operates an extensive network infrastructure that spans both urban and rural areas of Finland, ensuring its customers have access to high-speed internet and mobile connectivity.

Beyond its core telecommunications services, Elisa has ventured into various digital solutions and services. These include cloud services, data center solutions, cybersecurity services, and Internet of Things (IoT) solutions for businesses. The company has gained recognition for its commitment to innovation and has actively participated in the development and deployment of 5G networks across Finland.

Elisa serves around 2.8 million customers in Finland, Estonia and internationally. Elisa employs over 5,600 professionals in over 20 countries, and it is listed on the Nasdaq Helsinki stock exchange. However, please note that the company's current figures and details may have changed since then, so it's advisable to verify the most recent information from reliable sources.

4.1.3 Elisa Viihde

Elisa Viihde [31], provided by Elisa Oy, is a digital television and entertainment service that offers subscribers a wide array of on-demand content, live TV channels, and diverse entertainment options.

Through Elisa Viihde, users can access an extensive library of movies, TV series, documentaries, and other video content, which can be streamed on various devices including smart TVs, computers, tablets, and smartphones. The service encompasses both local Finnish content and international programming, catering to a range of preferences and tastes.

In addition to its on-demand content selection, Elisa Viihde provides access to live TV channels, enabling users to enjoy their favorite shows and sports events in real-time. The service often incorporates features such as pausing, rewinding, and recording live TV, empowering users with enhanced flexibility and control over their viewing experience.

Elisa Viihde is well-regarded for its user-friendly interface, personalized recommendations, and the option to create user profiles for different family members. The service also offers supplementary features such as parental controls, subtitles, and audio descriptions to promote accessibility for all users.

4.2 Overview of Elisa Viihde's architecture

In this section, a high-level view of Elisa Viihde's streaming is described using Figure 2.

In a television system, a satellite dish is a parabolic-shaped antenna that receives signals from a satellite in orbit. The dish collects the signals and reflects them towards an Integrated Receiver/Decoder (IRD). The IRD, also referred to as a satellite receiver or satellite set-up box, is an electronic gadget that gathers a radio-frequency signal and decodes it to verify that it's in the appropriate format for an end-user device or display. Its primary function is to unscramble TV signals, allowing viewers to access various types of content.

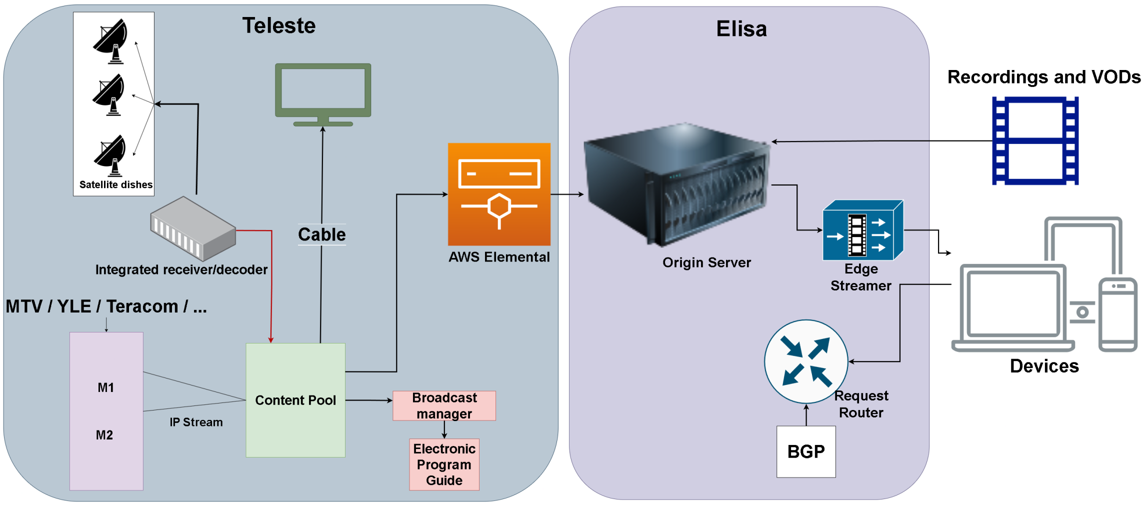

Figure 2: Elisa Viihde's streaming architecture

A content pool is a compilation of media resources, including video content, that is either owned or licensed by a company or organization. The connection between an IRD and a content pool is reliant on the particular context in which they are used. In some circumstances, an IRD might be utilized to receive and decode signals that contain content from a content pool. To illustrate, a broadcaster could use an IRD to acquire a signal from a satellite that contains a live sports event, which is part of their licensed content pool. On the other hand, a content pool might be employed to distribute media assets to multiple IRDs or other devices. For example, a television network might utilize a content pool to deliver on-demand video content to viewers who access the content via IRDs or other streaming devices.

Broadcast managers are responsible for managing the distribution and scheduling of content from the content pool to various channels and platforms. The Electronic Programme Guide (EPG) functions as a digital guide that offers comprehensive program information, including title, start time, and duration, enabling viewers to browse through available programming and choose the content they wish to watch. These elements work together to help broadcasters manage and distribute media assets effectively. The broadcast manager is responsible for handpicking the content from the content pool and scheduling it for broadcast across different channels. Consequently, the EPG serves as a tool to facilitate access to the content by providing viewers with necessary information.

YLE, MTV, and Teracom are examples of companies related to television broadcasting or distribution. Those companies are using multicast technology, M1 and M2 multicast addresses, to distribute content to a large number of end-users. M1 multicast addresses are frequently used for routing protocols, which enable devices on a network to exchange information. In media delivery scenarios, M1 multicast addresses may be employed by media companies to help direct the multicast stream between servers and other network devices. M2 is a common multicast address utilized to stream media, such as audio and video, over IP networks. By utilizing M2, media companies employ multicast technology to broadcast real-time, high-quality media content to a vast audience.

Cable television networks can use the content from the content pool. In our case, the content is used by Elisa Viihde through AWS transcoders. AWS Elemental MediaConvert is a video transcoding service that operates in the cloud and offers a scalable and economical approach to converting video content into several formats, including those necessary for OTT distribution, cable distribution, and broadcast. MediaConvert can handle various input and output formats, codecs, and frame rates, and offers capabilities like closed captioning, audio normalization, and frame rate conversion. With this, the parts of the infrastructure of Teleste that is utilized by Elisa ends.

The origin server is a component responsible for content storage, management, distribution, and security. It manages the content by organizing it into categories or playlists, updates metadata, and adds or removes content from storage as needed. The origin server distributes the requested content to the appropriate edge servers or streamers using various algorithms and technologies, and applies on-the-fly encryption to protect the content from unauthorized access or piracy. Additionally, it implements security measures such as access controls, firewalls, and intrusion detection systems to prevent unauthorized access to the server.

The Request Router is responsible for efficiently directing incoming user requests for content to the nearest or most suitable edge server or cache in the CDN network. It makes use of a variety of algorithms and metrics to determine the best possible routing path, considering factors like network proximity, server availability, and server load. The global routing system of the internet is based on Border Gateway Protocol (BGP), which manages the routing of packets from one network to another by exchanging routing and reachability information among edge routers. The primary role of BGP is to ensure network stability by quickly finding new paths when a route fails. It makes routing decisions based on paths defined by rules or network policies established by network administrators, which contributes to the efficient and reliable functioning of the internet.

4.3 Additional components

This section introduces software tools that are used to stream as well as protect the content by Elisa Viihde.

4.3.1 Digital Rights Managment (DRM)

Digital Rights Management (DRM) encompasses access control technologies that restrict the use of proprietary hardware and copyrighted works. These technologies aim to regulate the utilization, modification, and distribution of copyrighted works (e.g., software and multimedia content) and the enforcing systems within devices. In the digital era, DRM holds considerable significance for several reasons. It safeguards the intellectual property rights of content creators, preventing unlimited copying and distribution that could result in substantial financial losses. Additionally, DRM empowers content creators to establish various business models, such as subscriptions or pay-per-view, which rely on controlling the distribution and utilization of digital content.

Main Concepts of DRM:

- **Separation of content from rights:** This concept refers to the idea that the rights to use a piece of digital content are separate from the content itself. This allows for flexible distribution models where a user might buy a piece of content once and then pay separately for the rights to use it on different devices.
- **Granularity of content:** This refers to the ability of DRM systems to control access to specific parts of a digital content. For example, in a book, this could mean allowing access to certain chapters or pages.
- **Interoperability:** It refers to the ability of different DRM systems to work together. This is important because it allows consumers to use content across different devices and platforms.

Elements of a DRM System:

- **License server:** This is responsible for issuing licenses that define the consumer's rights to use the digital content.
- **License Database:** Stores licensing information like user rights, keys, access duration etc. It communicates with the license server.
- **Content Encryption:** The content is encrypted using cryptographic techniques to restrict access.
- **Content metadata:** This includes information about the digital content, such as its title, author, and the rights associated with it.

How a DRM System Works:

1. **Request for content:** The process begins when a consumer requests a piece of digital content from the content server.
2. **License generation:** The license server generates a license that defines the consumer's rights to use the content. The license is then sent to the consumer's device.
3. **Use of content:** The consumer's device uses the license to determine how the consumer can use the content. For example, the license might specify that the consumer can only view the content and not copy or print it.

DRM Techniques for protecting digital content

- **Encryption:** This involves encoding the digital content in such a way that only authorized users can decode it.
- **Watermarking:** This involves embedding a hidden mark in the digital content that can be used to track its distribution.
- **Access control:** This involves controlling who can access the digital content and what they can do with it.
- **Digital signatures:** This involves using a mathematical algorithm to prove the authenticity of a digital message or document.
- **Tamper resistance:** This involves designing the DRM system in such a way that any attempt to tamper with it is either impossible or will result in the digital content becoming unusable.

4.3.2 DRMToday

This is the DRM used by Elisa Viihde. DRMToday is DRM service provided by castLabs, a company specializing in cloud-based DRM solutions for the video streaming industry. DRMtoday provides licensing solutions tailored specifically for OTT businesses, ensuring streamlined DRM integration and enhancing content protection measures across multiple video streaming platforms.

4.3.3 Unified Origin

Unified Streaming Platform (USP) is a suite of software products that provide a complete end-to-end solution for streaming video and audio content. The platform includes Unified Origin, Unified Remix, Unified Packager, Unified Capture, and Unified Virtual Channel.

Unified Origin is a software plugin for industry-standard webservers that allows streaming any format to any device. It supports protocols such as HTTP Live Streaming (HLS), MPEG-DASH, as well as DRM, timed metadata, multi-channel audio, and accessibility features.

DRMToday complements Unified Origin by providing robust DRM capabilities for streaming companies like Elisa Viihde. When users access the streaming service, the client devices request the necessary licenses from DRMToday. DRMToday act as a license manager and verifies the user's authentication and entitlements, and upon successful validation, it issues the appropriate licenses to the client devices. These licenses enable decryption of the encrypted content delivered by Unified Origin, ensuring secure playback on authorized devices.

4.4 Example of a user scenario

Here is a user scenario of a video request into the CDN of a Elisa viihde which applies on-the-fly encryption on the content before delivering it to the user:

1. A user launches a video streaming application on their playback device, such as a smartphone, tablet, or smart TV, and chooses a video to watch.

2. The playback device initiates a content request to the CDN's Request Router, transmitting essential information like the video ID, device ID, and user login credentials.

3. Leveraging the Border Gateway Protocol, the Request Router determines the most optimal Edge Streamer location, considering the user's geographical proximity.

4. The request is then directed to the designated Edge Streamer server.

5. The Edge Streamer verifies with the Origin Server whether the requested content is available and accessible to the user.

6. The Origin Server component communicates with the DRMToday license server to procure the encryption keys required for encrypting the specific content intended for this user.

7. DRMToday supplies the necessary encryption keys to the Origin.

8. Unified Origin retrieves the raw video content, dynamically encrypts it using the provided keys, and returns the encrypted content to the Edge Streamer.

9. The Edge Streamer stores the encrypted content in its cache and promptly initiates streaming to the user's playback device.

10. The video player decrypts the encrypted video file using the decryption key from the license as it plays the content. It enforces any restrictions in the license like expiring after a certain time.

11. The license has to be periodically renewed by having the video player re-request it from the license server. This allows the service to control ongoing access rights.

Throughout the entire delivery process, the content remains encrypted, ensuring robust security, while on-the-fly encryption facilitates adaptability in licensing and access control measures.

Content delivery systems frequently store various copies of the same content in different cache servers spread across different locations. These copies have different qualities or resolutions, accommodating diverse user preferences and device capabilities. For instance, an image or video may exist in low and high resolutions. This approach enables the system to dynamically deliver the most suitable version to each user, enhancing the overall user experience.

When the user loses content rights (e.g., canceled subscription) or the license expires, the player cannot decrypt and play the video until obtaining a new valid license. Throughout this process, the encrypted video file remains safeguarded, with decryption keys exclusively provided to authorized user devices via the license. This ensures DRM's secure access control.

5 Simulation setup

In this chapter, the tools and technologies used in the simulation are introduced. Additionally, their interactions are shown and, finally, details regarding its deployment is showcased.

5.1 Overview of the tools and technologies

The section briefly introduces the various tools and technologies used in the simulation setup, including Google Cloud services like Cloud Storage, Live Streaming API, and Media CDN, as well as third party tools like FFmpeg, Unified Origin, DRMToday, and VLC Player.

5.1.1 Google cloud

Google Cloud is a cloud computing platform provided by Google. It provides users with a suite of management tools and it offers a range of modular cloud services that encompass computing, data storage, data analytics, and machine learning capabilities. Google Cloud provides Infrastructure as a Service (IaaS) [1], Platform as a Service (PaaS) [2] , and SaaS solutions.

5.1.2 FFmpeg

FFmpeg is an open-source software suite that provides a collection of libraries and programs for handling multimedia data. It is widely used for tasks such as video and audio encoding, decoding, transcoding, filtering, and streaming. FFmpeg supports a wide range of multimedia formats, codecs, and protocols, making it a versatile tool for working with audio and video content.

In our case, it will be used to send the live signal. It supports various streaming protocols, including Real-Time Messaging Protocol (RTMP), Real-Time Transport Protocol (RTP), and HLS.

[1] IaaS is a cloud computing paradigm that grants users instant access to essential computing resources like servers, storage, networking, and virtualization. This model offers a compelling alternative to the conventional method of procuring computing resources, which typically involves significant time and financial investments.

[2] PaaS is a comprehensive cloud ecosystem that encompasses all the essential elements developers require to create, execute, and oversee applications. This encompasses not only servers and operating systems but also networking, storage, middleware, tools, and a variety of other components.

5.1.3 Live Streaming API

A Live Streaming API [3] is a software interface that allows developers to create, manage, and deliver live streaming content. The Live Stream API transcodes live signals into direct-to-consumer streaming formats, including Dynamic Adaptive Streaming over HTTP (DASH/MPEG-DASH), and HLS, for multiple device platforms. It supports only one type of video codecs, H.264 (AVC), as well as one type of Audio codecs, AAC.

This tool supports integration with Cloud Storage, Cloud Audit Logs, and Google Cloud infrastructure. It also supports Content encryption and automatic infrastructure provisioning.

5.1.4 Cloud Storage

Google Cloud is a RESTful[4] online scalable and durable object storage service provided by Google as part of its GCP suite of cloud computing services.

It can be used to store a wide variety of data, including images, videos, logs, backups, and more. It is designed to handle large amounts of unstructured data such as files, backups, multimedia content, and other types of data.

Google Cloud is a highly customizable service, as an instance, storage Classes Includes different tiers with varying performance and pricing to meet for the type of data stored. For example, standard class can be chosen for frequently accessed data and an archival class for infrequently accessed data. Additionally, it has customization regarding Data Lifecycle Management, Access Controls, Encryption, Integration and API s , and Interoperability (allowing data transfer between different cloud providers or on-premises storage systems).

5.1.5 Media CDN

Media CDN is Google Cloud's media delivery solution. It is a global edge network for streaming media, backed by Google's global network of edge caches in thousands of locations. It uses the edge-caching infrastructure to serve content as close to users as possible. It leverages YouTube's infrastructure to deliver video streams, VoD, and large file downloads efficiently. It is designed specifically for media and entertainment companies and provides

[3] APIs, short for Application Programming Interfaces, act as intermediaries between software applications, allowing them to communicate and interact with each other. They define the rules for interaction and provide a standardized way for applications to access specific features or data from external services, libraries, or systems, abstracting the complexity of underlying processes. This allows applications to request and exchange information or perform actions seamlessly without needing to understand the internal workings of the systems they interact with.

[4] A RESTful service, also known as a Representational State Transfer service, is an architectural approach used in web development to design networked applications. Its primary purpose is to enable the creation of Application Programming Interface (API)s that facilitate communication between various systems via the internet.

features such as universal support for any origin and backend, fine-grain cache controls, real-time logging and metrics, and modern protocols for better user experience. The service also supports advertising insertion and Artificial Intelligence (AI) / Machine Learning (ML) analytics.

The Media CDN service from Google Cloud is based on three main components:

- EdgeCacheService segments streaming media as metadata and video segments.
- The Cloud Storage origin processes segmented video from the segment storage service.
- The Load balancer origin handles media metadata for lookup and delivery.

Media CDN allows customers to define cache configurations for different routes, enabling them to customize the caching behavior based on content type and client request attributes.

It supports Compute Engine, Cloud Storage, and Google Kubernetes Engine [5] backends and it can retrieve content from any HTTP-capable origin, including buckets outside of Google Cloud.

In table 4, some of differences between this CDN and Elisa's is displayed:

Table 4: Differences between Elisa Viihde's CDN and Media CDN

Feature	Elisa Viihde's CDN	Media CDN
Geographic reach	Primarily Finland	Global
Content types	Streaming video	Streaming video, static web assets, large file downloads
Optimized for	Low latency	High throughput
Pricing	Per-stream	Per-GiB

5.1.6 Unified Origin and DRMToday

For this streaming flow, a cloud version of Unified Origin and DRMToday will be used. Furthermore, DRMToday will be used with Unified Origin to create On-the-fly encryption.

On-the-fly encryption, also known as real-time encryption, is a technique where data is encrypted and decrypted in real-time as it is being accessed or transmitted. This means that the encryption and decryption processes happen on-the-fly, without requiring the data to be fully encrypted or decrypted beforehand. On-the-fly encryption is often employed as a part of DRM systems to protect the digital content. When a user tries to access the protected content, the DRM system decrypts the content on-the-fly using the appropriate decryption keys,

[5] Google Kubernetes Engine is a fully managed platform designed for deploying containerized applications. Containerized applications are software applications that are bundled and executed using container technologies such as Docker. Google Kubernetes Engine (GKE) offers a reliable and production-ready environment for running these containerized applications efficiently.

allowing the user to view or use the content. Similarly, when the user saves or transmits the content, the DRM system encrypts it on-the-fly to ensure its security during transmission or storage.

5.1.7 VLC Player

A video player is a type of software or app that enables users to watch video content on their devices. It comes with a user interface that allows control over playback, adjustment of settings, and interaction with the video content. The process of a video player requesting content involves the following steps:

- **Initialization:** This is when the video player is set up and loaded onto a webpage or app.
- **Source Selection:** The video player identifies the source of the video content, which could be a direct Uniform Resource Locator (URL) to a video file or a streaming service.
- **Content Request:** The video player communicates with the server that hosts the video content, requesting the specific video file or stream.
- **Server Response:** The server responds to this request by supplying the requested video content, either as a video file or a stream.
- **Content Loading:** The video player begins to load the video content it received from the server.
- **Playback:** Once the video content is fully loaded, the video player initiates the playback, showing the video on the screen.

5.1.8 Cloud Shell

Google Cloud Shell is a Command-Line Interface (CLI) tool offered by GCP, enabling users to directly access and control their cloud resources through a web browser. It presents a shell environment within the browser, equipped with a pre-configured collection of command-line tools and utilities. This eliminates the requirement to locally install and configure these tools on the user's machine.

It can be accessed from any web browser and it comes with a pre-installed set of common development tools and utilities, such as the Google Cloud SDK, Docker, Git, and many others, eliminating the need for many manual installations. Additionally, Cloud Shell provides a seamless interface to interact with various Google Cloud services and resources, allowing you to create, manage, and monitor your infrastructure and applications.

5.2 Overview of architecture

In figure 3, the interaction between the components on the cloud is shown.

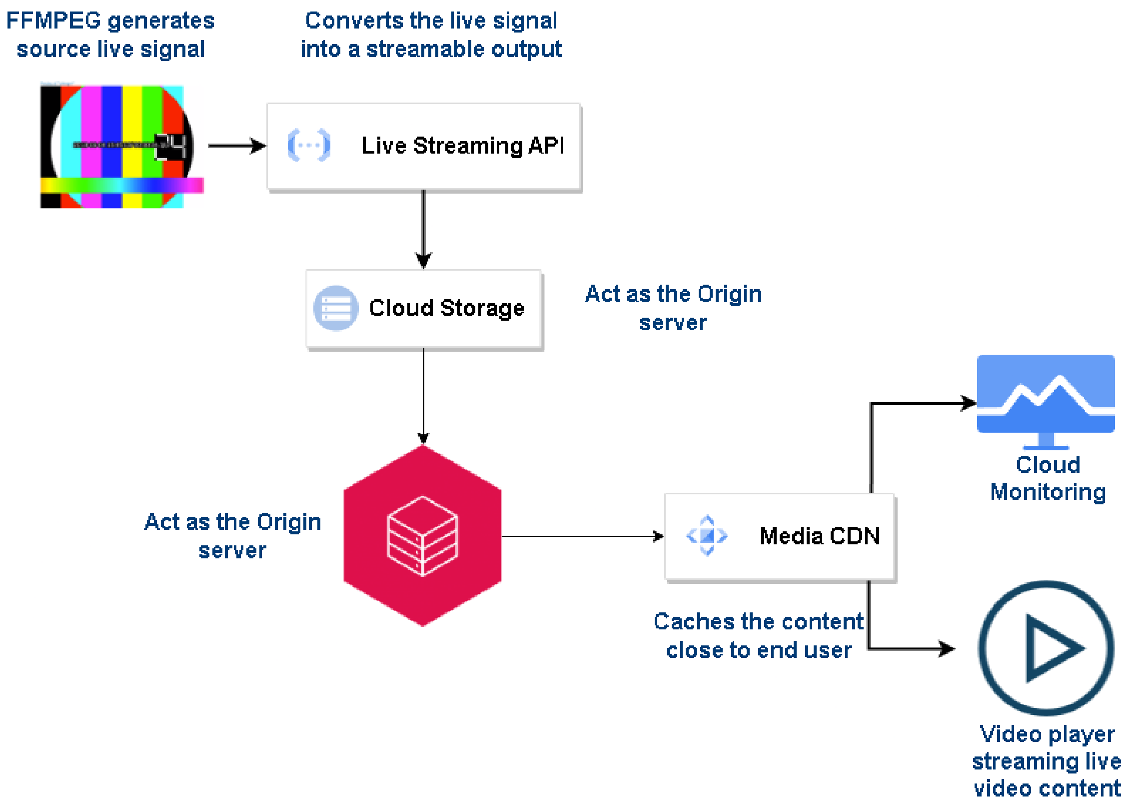

Figure 3: On the cloud streaming flow

We will begin by configuring the Live Streaming API components - namely, the Input and Channel. Next, we'll initiate a live feed using FFmpeg, as it can generate a live signal. The Live Streaming API will then handle the transcoding of the live feed. The resulting transcoded video manifest and segments will be stored in a designated Cloud Storage bucket.

Afterward, a connection between Unified Origin and the Storage bucket will be established, allowing us to fetch the data. To ensure security, on-the-fly encryption will be applied to the content using DRMToday as the license manager, providing encryption and decryption keys while also performing validation.

To optimize content delivery, Media CDN will act as an edge cache for the Origin. By doing this, VLC Player can efficiently play the live content that is cached through the Media CDN, ensuring a smooth and seamless viewing experience.

Finally, a Cloud Monitoring dashboard to visualize and track the activities of the Media CDN will be implemented, providing insights into its performance and usage.

Table 5 is showing how these components resemble that of Elisa Viihde.

Table 5: The equivalent of Elisa Viihde components on the cloud

Elisa Viihde flow	On-cloud flow
Satellite dish	FFmpeg
AWS Transcoders	Live streaming API
Origin server	Cloud Storage together with Media CDN (EdgeCacheOrigin)
Edge caches	Media CDN (EdgeCacheService)
Unified Origin	Unified Origin
DRMToday	DRMToday
Any playback devices	VLC Player

5.3 Simulation settings

- Type and source of content: A signal from FFmpeg of dimensions 1280x720 and H.264 as the video codecs and (AVC) as the Audio codecs.
- Protocol: The content will be transmitted over HLS.
- Location: The location of the VMs will be Northern Europe.
- Metric: Through Cloud Monitoring, record data Transferred between the CDN and the user in Gigabyte (GB) and Terabyte (TB).

More details on the deployment are in Appendix B.

6 Calculating the TCO of streaming solutions

In this chapter, the cost associated with both streaming flows is calculated using information we have from Google Cloud and Elisa Experts.

To make this comparison, we have to rely on a normalized unit. In our case, it will be Data Transferred, especially, Outbound Data Transfer. Outbound Data Transfer refers to the movement of data between the streaming platform (Elisa Viihde or Google's Media CDN) and the end-users or clients who are accessing the streaming service. We will normalize the data based on the amount of data transferred between the streaming platform and the users. This will be measured in terms of TB of data transferred.

Data transfer volume in streaming services is influenced by:

- **Streaming Quality:** Higher quality streams (e.g., HD, 4K) require more data to be transferred compared to lower-quality streams.
- **Number of Users:** Data transfer needs increase with more simultaneous content access.
- **Streaming Traffic:** Events like new content launches or live broadcasts can lead to spikes in data transfer.
- **Content Library Size:** Larger content libraries may require more data transfer to cater to diverse user preferences.
- **Geographical Distribution:** Different CDNs may have varying data transfer costs.

There are two types of daily traffic that Elisa Viihde system experiences : **Daily CDN traffic:** during the summer weeks of 2023, the traffic ranged between 1700-2000 Terabytes per day. **Spring time:** the traffic ranged between 2000-2700 Terabytes per day. The costs will be calculated in the currency of Euro.

For the sake of this calculation, **a traffic of 1800 Terabytes/day will be used.**

6.1 Calculating the cost on the cloud

To use Google Cloud services, the pricing generally follows a tiered model, where per-unit pricing decreases at higher usage volumes. The following are examples of this model:

- Compute resources such as Compute Engine and Cloud Run follow a sustained use pricing model, where the per-hour rate decreases as resource usage increases.
- Storage services like Cloud Storage and Cloud SQL utilize a scalable pricing structure based on tiers, resulting in lower per-GB rates as more data is stored.
- Network egress pricing is tiered, meaning the per-GiB bandwidth rate decreases with higher data volumes.
- Google provides committed use discounts for 1-3 year commitments on resource usage, effectively reducing the per-unit price.
- Some services include volume discounts for higher usage. For instance, Cloud Tasks offers lower per-task pricing at larger scales.
- Google Cloud operates on a ”pay-as-you-go” model, with no fixed overhead costs or upfront fees.

Since we can not simulate the whole data transfer on the cloud. We will assume that the cost will be of a linear nature, so the different discounts on the total price that we can get will be applied.

The calculation will be based on a daily CDN traffic of 1800 Terabytes. This accounts for 54,000 Terabytes each month. Which is 49,112 TiB.

As mentioned in section 3.2.7, the calculation will follow the following expression:

$$\textbf{TCO [On Cloud]} = C[MediaCDN] + C[CloudStorage] + C[LiveStreamAPI] \quad (6.1)$$

Table 6: Cache egress Price (per GiB) usage tiers

Amount of Usage	Price (per GiB)
1 - 10 TiB/month	€0.06/GiB
10 - 100 TiB/month	€0.035/GiB
100 - 1000 TiB/month	€0.02/GiB
1000 - 10000 TiB/month	€0.012/GiB
10000 - 50000 TiB/month	€0.008/GiB

Table 6 shows the pricing tiers of Cache egress per GiB usage on a monthly bases. While table 7 shows the total monthly price of using **Media CDN**.

Table 7: Price calculation of cache egress

Amount of Usage	**Price (per GiB)**	**Total Price**
1 - 10 TiB	10 TiB * 1024 GiB/TiB * €0.06/GiB	€614.4
10 - 100 TiB	90 TiB * 1024 GiB/TiB * €0.035/GiB	€3,145.6
100 - 1000 TiB	900 TiB * 1024 GiB/TiB * €0.02/GiB	€18,432
1000 - 10000 TiB	9000 TiB * 1024 GiB/TiB * €0.012/GiB	€110,592
10000 - 50000 TiB	30112 TiB * 1024 GiB/TiB * €0.008/GiB	€245,299.68

Total price: €614.4 + €3,145.6 + €18,432 + €110,592 + €245,299.68 = €378,083 / Month. Which is 4,536,996 EUR for 1 year.

As for **Cloud Storage**, a storage that has a total amount of storage of 1 PiB that is placed in Finland (europe-north1), would cost 19,000 EUR monthly. Which is 228000 EUR for 1 year.

As for **Live Stream API**, the stream that constists of 4 channels, each channel has 2 SD and 2 HD inputs. Each channel has 2 SD and 2 HD outputs. The stream is working 24 hours a day, 7 days a week. The cost is 4,700 EUR per 1 month. Which is 56400 EUR for 1 year.

6.2 Calculating the cost of using Elisa Viihde

As mentioned in section 3.2.7, the calculation will follow the following expression:

$$\textbf{TCO [On Premises]} = C[\text{labor}] + C[\text{hardware}] + C[\text{real estate}] + C[\text{electricity}] \tag{6.2}$$

6.2.1 C[labor]

According to [32], from 2020-2021, the average Network engineer salary in Finland is: 3,860 EUR per month. According to the information from Elisa Viihde, there are 2 full-time employees that are working to maintain the hardware and. So, the annual cost is:

C [labor] = 2 X 3,860 X 12 = 92640 EUR

6.2.2 C[hardware]

According to Elisa Viihde experts, the hardware is owned by the company and is not rented. Table 8 shows the current used hardware and its specifications.

Table 8: Server specifications

Server model	CPU	Data disk	Power consumption	Dual power supply	RAM	Number of units
Dell R730	2x Xeon E5-2660 v4 14c	11x 800GB SATA	196W	yes	268	19
Dell R740	1x Xeon Gold 5120 14c	8x 1.92TB SATA	190W	yes	268	21
Dell R7515	1x AMD EPYC 7402P 24c	13x 1.92TB SATA	200W	yes	400	9
Dell R7515	1x AMD EPYC 7543P 32c	9x 3.84TB SAS	340W	yes	268	24

Estimating the cost of hardware is difficult, as it depends on many factors such as Location, timing as hardware prices tend to decline over time. It also depends on vendor and support level as costs would be higher if the purchase includes a multi-year warranty/support package from the vendor. Moreover, Volume discounts are applied when large enterprises or resellers buy in bulk will get better pricing than one-off purchases. Finally, taxes - Finland has a 24% VAT that would need to be factored into the final pricing.

In table 9, the prices (in Euro) of the components of each server were estimated based on multiple resources [33], [34], [35], [36], [37], and [38].

Table 9: Estimated cost of servers

Server	Base Model	CPU	Data Disk	RAM	Dual power supply	Total Cost
Dell R730	900 - 1,300	1,100 - 1,400	2,000 - 3,000	1,600	150	5,400 to 6,300
Dell R740	1,800 - 2,700	1,300	2,200 - 3,300	2,500	150	7,500 - 9,700
Dell R7515 (1x AMD EPYC 7402P 24c)	1,500 - 2,000	800 - 1,000	3,900 - 5,200	1,600	150	7,800 - 9,800
Dell R7515 (1x AMD EPYC 7543P 32c)	1,500 - 2,000	2,000 - 2,500	4,500 - 6,750	1,600	150	9,600 - 12,850

Table 10 shows the final prices that will be used in the calculation:

Table 10: Final cost of servers in Finland

Server Name	Average Total Cost	Total Cost (with 24% VAT)
Dell R730	€5,850	€ 7,254
Dell R740	€8,600	€10,664
Dell R7515 (24c)	€8,800	€10,912
Dell R7515 (32c)	€11,225	€13,919

Table 11 shows the total cost of buying the quantity mentioned by Elisa Viihde experts of each type of server.
Note: There were 2 servers of type R7515 (64c) but they were added in the calculation to the R7515 (32c) as information regarding its energy consumption was unattainable and to keep the calculation consistent.

Table 11: Total servers costs

Server name	Price per server (€)	Number of servers	Total price (€)
Dell R730	7,254	19	137,826
Dell R740	10,664	21	223,944
Dell R7515 (24c)	10,912	9	98,208
Dell R7515 (32c)	13,919	24	334,056

So, the cost of owning the necessary hardware would be:

C[hardware] = 137,826 + 223,944 + 98,208 + 334,056 = 794,034 EUR

Considering that we have 3 types of servers, according to [39], [40], and [41], information about the sizes of the servers that are used is displayed in table 12.

Table 12: Server sizes

Server Model	Rack Units (RU)
Dell PowerEdge R730	2RU
Dell PowerEdge R740	2RU
Dell PowerEdge R7515	2RU

Moreover, based on the information in Table 12 and Table 13, we need to add the cost of the server cabinets to the total cost. The Total number of cabinets required is 9 cabinets, as 42U server cabinets can accommodate up to 22 of the mentioned servers. According to [42] and [43], the cost for 42U server cabinets in Finland including VAT is 3400 - 3600 Euros. The average is 3500 Euros.

Total cost of server cabinets is 9 X 3500 = 31,500 Euros. So, the new cost of owning the necessary hardware would be:

C[hardware] = 794,034 + 31,500 = 825,534 EUR

6.2.3 C[real estate]

To calculate this cost, the next steps will be followed:

- Get the number of servers
- Calculate how many server cabinets are required.
- Determine the size of space needed per cabinet.
- Calculate total space needed for each cabinet.
- Research average rental rates per sq meter.
- Calculate total monthly or annually rent.
- Multiply by 12 months to get the estimated annual real estate cost. This step depends on which cost was found – monthly or yearly.

Data centers and network operations centers can be considered as industrial real estate due to their unique demands for specific environmental conditions, notably substantial power and cooling requirements, which set them apart from other conventional commercial real estate properties.

Table 13: Servers location and real estate costs

Location	Number of Servers	Server Cabinet Required
Järvenpää	3	1
Joensuu	3	1
Pasila	18	1
Raision luola	7	1
Säteri	12	1
Tampere Nalkala	6	1
Tapiola	10	1
TNL 2210	11	1
Ylivieska	3	1

According to the **annual prime rents per square meter** for industrial properties in Finland [44], and due to the inability to attain precise information regarding rents for each location in Table 13, we will use the average between the four cities mentioned in the source which are: Helsinki (111 Euros), Turku (99 Euros), Tampere (99 Euros), and Oulu (87 Euros). **The Average is 99 Euros annual prime rents per square meter**.

According to [45], [46], [47], and [48], 1 server cabinet requires space of approximately 1 square meter.

Considering that we have 9 servers, this means 9 square meters.

C[real estate] = 9 X 99 = 900~Euros

6.2.4 C[electricity]

Power consumption varies wildly depending on current usage of the server and can not be compared without taking into account what the systems are doing at the given time. This serve as an estimate of cost based on information from Elisa Viihde experts.

To calculate this cost, the next steps will be followed:

- Get the number of servers per each type.
- Get the power consumption of each type.
- Calculate the total power consumption of the devices for a year.
- Get the business electricity prices in Finland.
- Use the price of electricity and total power consumption to calculate the total cost.

In Table 14 we can see the number of units of each server, power consumption of each server, and the total energy consumption in Wattage [1]

Table 14: Total energy consumption of all the units of each type

Server Model	Power Consumption (W)	Number of units	Total energy consumed (W)
Dell R730	196	19	3724
Dell R740	190	21	3990
Dell R7515	200	9	1800
Dell R7515	340	24	8160

That means the total hourly energy consumption is:
3724 + 3990 + 1800 + 8160 = 17674 W per hour

[1] The watt (symbol: W) is the unit of power or radiant flux in the International System of Units (SI), equal to 1 joule per second

To get the annual consumption, we multiply by the number of hours in a year:

17.674 KWh X 24 X 365 = 154824.24 KWh

Which is approximately 155 MWh. According to Statistics Finland [2] [49], the cost (Including VAT) of electricity for enterprise and corporate clients that consumes between 20-499 MWh annually is 12.32 c/kWh, which is 0.1232 Eur/kwh.

With this information, we can conclude:

C[electricity] = Consumption (kWh) x Price (Eur/kWh)
= 154,824.24 kWh x 0.1232 Eur/kWh
= **19,080.13 Eur**

[2]Statistics Finland is the national statistical institution in Finland. It was established in 1865 to serve as an information service and to provide statistics and expertise in the statistical sciences. The institution employs more than 800 experts from varying fields.

7 Results, discussion, and limitations

In this chapter shows the results, an interpretation of the results we got out of this study, and several limitation of the study.

7.1 Results

From Chapter 6 we obtain the cost of streaming over Media CDN and Elisa Viihde CDN.

7.1.1 On-cloud:

Table 15 represents the cost of all the parameters in equation 7.1 with the TCO of 1 and 3 years.

Table 15: On-Cloud TCO

Cost item	1-year cost (EUR)	3-year cost (EUR)
C[Media CDN]	4,536,996	13,610,988
C[Cloud Storage]	228,000	684,000
C[Live Stream API]	56,400	169,200
TCO [On-Cloud]	4,821,396	14,464,188

7.1.2 On-premises:

Table 16 represents the cost of all the parameters in equation 7.2 with the TCO of 1 and 3 years.

Table 16: On-premises TCO

Cost item	1-year cost (EUR)	3-year cost (EUR)
C[Labor]	92,640	277,920
C[Hardware]	275,178 [a]	825,534
C[Real estate]	900	2,700
C[Electricity]	19,080	57,240
TCO [On Premises]	387,798	1,163,394

[a] Considering that the hardware can be replaced in 3-5 years, the price we obtained is the cost for 3 years. This price represents the 3-year price divided by 3.

The results reveal significant variations in operating costs when comparing Elisa Viihde's on-premises CDN infrastructure usage with migrating to Google's cloud-based Media CDN over a 3-year period. The total cost of ownership (TCO) for the on-premises solution was estimated at €1,163,394 for 3 years, while the projected TCO for the cloud-based Media CDN reached €14,464,188 during the same period.

The major contributing factor to this stark contrast lies in the high costs associated with Google's Media CDN, shown in Figure 4, primarily due to the substantial expenses for bandwidth and content delivery, which were forecasted at €4,536,996 annually. Cloud storage costs were relatively minor at €228,000 per year, and Live Streaming API expenses added another €56,400 annually.

On the other hand, for the on-premises infrastructure, shown in Figure 5, the largest cost component was hardware acquisition and maintenance, projected at €825,534 over 3 years. Other significant factors were labor costs at €92,640 annually and electricity costs at €19,080 per year. Real estate costs were negligible at €900 per year.

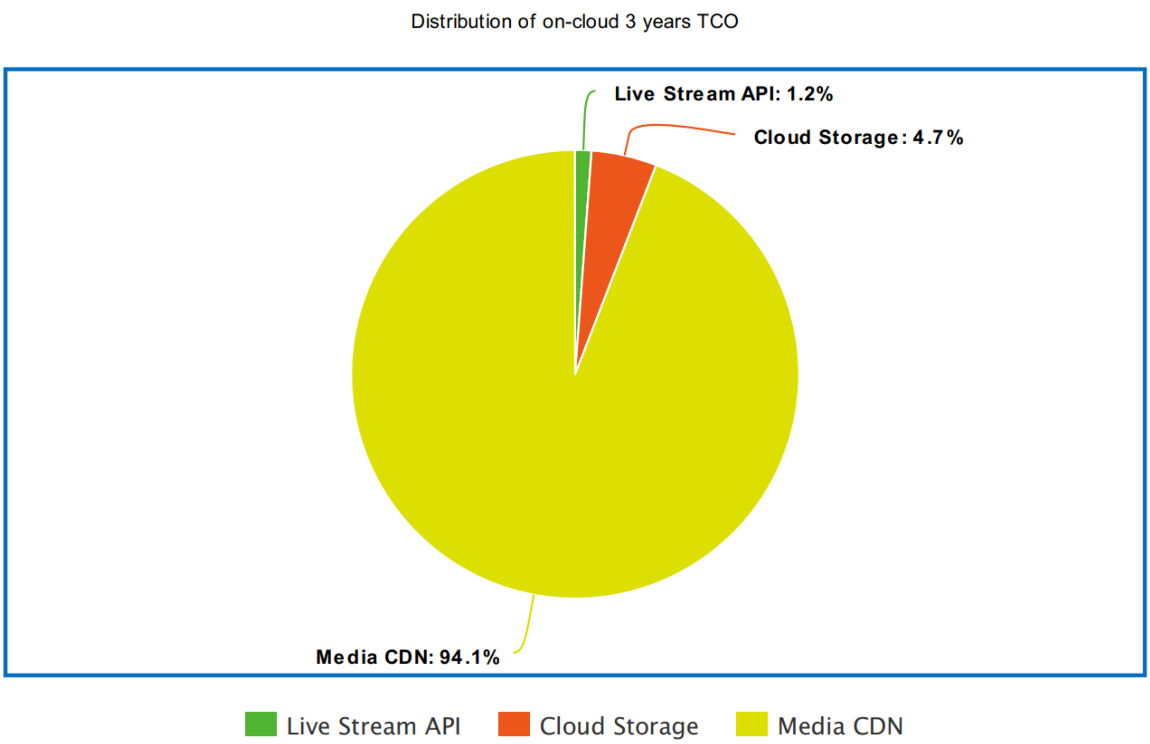

Figure 4: Distribution of on-cloud 3 years TCO

Distribution of on-premise 3 years TCO

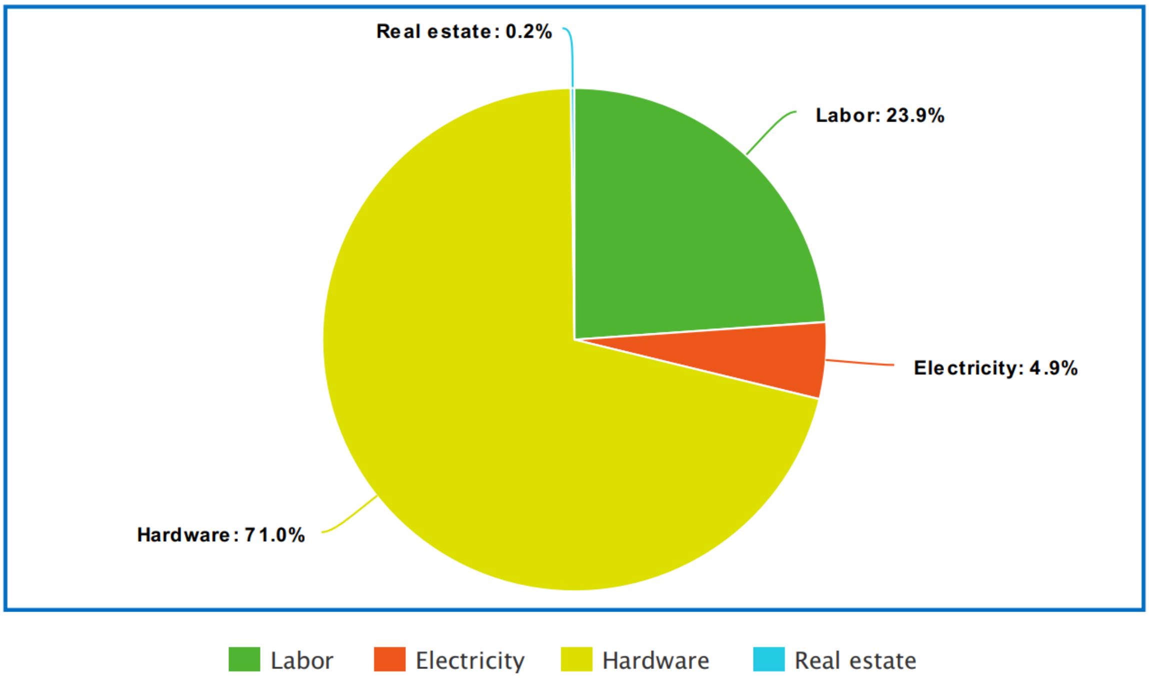

Figure 5: Distribution of on-premise 3 years TCO

7.2 Discussion

To interpret the results, it is important to note that the on-cloud costs were calculated based on averages and did not account for fluctuations in usage and demand throughout different times of day, days of the week, and yearly seasons in Finland. Cloud pricing is dynamic and consumption-based, so actual costs might deviate from the estimates provided. Factoring in expected demand cycles and modeling costs accordingly would strengthen the analysis. For Example, the cost of Live Streaming API was calculated on the bases of being used 24 hours a day, 7 days a week with the same intensity. However, the amount of channels can change after midnight due to less usage.

The on-cloud estimate also did not include potential savings from Sustained use discount (SUD) [1] or Committed Use Discount (CUD) [2], which could significantly lower costs. However, using these discounts requires committing to certain service levels, impacting flexibility, and necessitating a thorough evaluation of the trade-offs.

[1] Sustained use discounts (SUDs) are discounts are available for customers who use the CDN for a significant portion (25%) of the billing month.

[2] Committed Use Discounts (CUDs) are a type of pricing model offered to customers who commit to using a certain amount of cloud resources over a fixed period, 1 - 3 years.

Furthermore, the on-premises TCO might be underestimated due to the challenges in accurately capturing all hidden costs, such as spare parts and upgrade expenses over time. In contrast, cloud costs are more transparent and predictable, while on-premises costs can fluctuate unexpectedly, leading to increased uncertainty.

On-premises solution offers complete control over the infrastructure, allowing for potential optimizations in energy efficiency and labor costs over time. In contrast, cloud customers are limited by the provider's configurations. With active resource management, the on-premises TCO could potentially decrease further.

Moreover, cloud providers are increasingly investing in renewable energy sources. Google cloud optimization practices [50], such as leveraging energy-efficient hardware or adopting green data center practices, can also explain the difference in cost. On another note, cloud-based solutions can potentially shift the demand for human resources as maintenance tasks will be handled by the cloud provider. This transition may contribute to a slight job displacement as the maintenance responsibility might be carried out by offshore workers.

The analysis didn't consider the qualitative differences between cloud and on-premises solutions, such as flexibility and scalability. Additionally, the ease of switching between cloud providers compared to replacing on-premises infrastructure should be taken into account. For example, Google's Media CDN offers high scalability potential, allowing streaming content providers to meet fluctuating demands effectively. On the other hand, on-premises infrastructure may have limitations in terms of scalability, especially during peak usage periods. Adopting cloud-based solutions could improve scalability, ensuring uninterrupted service delivery to a growing user base.

Strategic factors should be considered alongside the quantitative cost data. Although the TCO estimate favors on-premises, the actual difference is likely smaller due to the challenges in forecasting cloud usage, discounts, and on-premises expenses accurately over time. To make the optimal choice, a broader perspective that considers strategic value in addition to costs is necessary. The findings provide a baseline comparison for costs, but further analysis, such as sensitivity analysis of variables, is needed to reinforce the results.

7.3 Limitations

Limitations refer to the deficiencies or drawbacks of a study that could impact the understanding of its outcomes. These are aspects acknowledged to present a well-rounded perspective on the study's findings. Limitations may arise from multiple factors, including resource constraints, methodological decisions, sample size, data collection procedures, or inherent challenges in investigating a specific phenomenon.

- Simulating Elisa Viihde's system on the cloud may not accurately represent the actual system and processes. The simulation may miss key components or simplify complex processes, threatening the accuracy of cost comparisons.
- The adequacy of the chosen tools and technologies to simulate the streaming process accurately.
- Assumptions made about hardware, electricity, real estate, and labor costs may not reflect true costs. Lack of precise data on these costs reduces accuracy of on-premise cost estimates.
- Forecasting costs using billing consoles has uncertainties. Actual costs may differ from forecasts.
- The study focuses solely on Elisa Viihde's specific system and processes. Results may not generalize to other streaming providers with different architectures.
- The study only compares two options - Elisa Viihde's CDN and Google's Media CDN. Other CDN options are excluded, reducing the generalizability of the study.
- The cost comparison is time-bound, representing a snapshot. Cost dynamics may change over time, affecting longevity of findings.

8 Conclusion

This study aimed to compare the operating costs between Elisa Viihde's on-premises CDN infrastructure and Google's Media CDN cloud solution. The study followed a methodology of simulating Elisa Viihde's system on Google Cloud and forecasting cloud costs associated with using the different services to get the cost of the cloud solution. While collecting data on the hardware equipment used, in addition to Labor, electricity and real estate was done to calculate on-premises expenses.

The study provides a quantitative cost comparison between the two options, demonstrating a large price differential currently favoring the on-premise solution. The Total Cost of Ownership estimates over 3 years showed streaming on-cloud to be significantly more expensive at 14.5 million EUR in comparison to 1.2 million EUR for the existing on-premises system.

The major contributing factor to this stark contrast lies in the high costs associated with Google's Media CDN. It accounted for 94.1% of the total costs. While owning the hardware to stream on-premise accounted for 71.0%.

In these calculations, the cloud pricing did not account for fluctuations in demand or utilize committed use discounts, likely overestimating the actual costs. Meanwhile, the on-premises costs may be underestimated due to challenges in capturing all expenses over time accurately. Although the comparison favours the cloud solution, strategic factors such as scalability, flexibility, ease of switching providers, and qualitative differences should also be evaluated when choosing the appropriate streaming option.

While the research achieved its goal of comparing estimated costs, the uncertainties limit definitive conclusions. The findings suggest on-premises is currently more economical. This baseline analysis can inform Elisa Viihde's decision-making, but further research is needed to account for dynamic cost factors and strategic considerations beyond purely quantitative TCO. Extending the comparison to other CDNs, improving simulations with real usage data, and weighing in qualitative differences would provide a more holistic perspective. As streaming demand and technologies evolve rapidly, continuous re-evaluation of cost-benefit trade-offs will be crucial.

REFERENCES

[1] A. Vakali and G. Pallis, "Content delivery networks: Status and trends," *IEEE Internet Computing*, vol. 7, no. 6, pp. 68–74, 2003. DOI: `10.1109/MIC.2003.1250586`.

[2] A. Vakali and G. Pallis, "Content delivery networks: Status and trends," *IEEE Internet Computing*, vol. 7, no. 6, pp. 68–74, 2003. DOI: `10.1109/MIC.2003.1250586`.

[3] J. Dilley, B. Maggs, J. Parikh, H. Prokop, R. Sitaraman, and B. Weihl, "Globally distributed content delivery," *IEEE Internet Computing*, vol. 6, no. 5, pp. 50–58, Sep. 2002, ISSN: 1089-7801. DOI: `10.1109/MIC.2002.1036038`. [Online]. Available: `https://doi.org/10.1109/MIC.2002.1036038`.

[4] N. E. S. G. H. T. W. B. M. D., "Unix and linux system administration handbook," *IEEE Internet Computing*, vol. 5, 2018. [Online]. Available: `https://www.worldcat.org/title/1005898086`.

[5] *Data volume of global content delivery network internet traffic from 2017 to 2022*, 2022. [Online]. Available: `https://www.statista.com/statistics/267184/content-delivery-network-internet-traffic-worldwide/`.

[6] L. M. Hilty and B. Aebischer, "Ict for sustainability: An emerging research field," in *ICT Innovations for Sustainability*, ser. Advances in Intelligent Systems and Computing, L. Hilty and B. Aebischer, Eds., vol. 310, Cham: Springer, 2015. DOI: `10.1007/978-3-319-09228-7_1`. [Online]. Available: `https://doi.org/10.1007/978-3-319-09228-7_1`.

[7] C. Freitag, M. Berners-Lee, K. Widdicks, B. Knowles, G. Blair, and A. Friday, "The real climate and transformative impact of ict: A critique of estimates, trends, and regulations," *Patterns*, vol. 2, p. 100 340, 2021. DOI: `10.1016/j.patter.2021.100340`.

[8] ISO/IEC, *ISO/IEC (2010) ISO 26000 guidance on social responsibility*, 2010.

[9] M. C. Branco and L. L. Rodrigues, "Corporate social responsibility and resource-based perspectives," *Journal of Business Ethics*, vol. 69, no. 2, pp. 111–132, 2006. DOI: `10.1007/s10551-006-9071-z`.

[10] J. Szymankiewicz, "Going green: The logistics dilemma," *Logistics Information Management*, vol. 6, pp. 36–43, 1993.

[11] B. Li, Z. Wang, J. Liu, and W. Zhu, "Two decades of internet video streaming: A retrospective view," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 9, no. 1s, Oct. 2013, ISSN: 1551-6857. DOI: `10.1145/2505805`. [Online]. Available: `https://doi.org/10.1145/2505805`.

[12] D. Wu, Y. Hou, W. Zhu, Y.-Q. Zhang, and J. Peha, "Streaming video over the internet: Approaches and directions," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 282–300, 2001. DOI: 10.1109/76.911156.

[13] H. Sun, A. Vetro, and J. Xin, "An overview of scalable video streaming," *Wireless Communications and Mobile Computing*, vol. 7, pp. 159–172, Feb. 2007. DOI: 10.1002/wcm.471.

[14] L. De Cicco and S. Mascolo, "An adaptive video streaming control system: Modeling, validation, and performance evaluation," *IEEE/ACM Transactions on Networking*, vol. 22, no. 2, pp. 526–539, 2014. DOI: 10.1109/TNET.2013.2253797.

[15] A. O. Alabbasi, "A quantitative framework for cdn-based over-the-top video streaming systems," Ph.D. dissertation, Purdue University Graduate School, 2020. [Online]. Available: https://doi.org/10.25394/PGS.11455869.v1.

[16] *Cloud Based Video Streaming Market Size, Share, Growth, and Industry Analysis, By Type(Public Cloud, Private Cloud, Hybrid Cloud and Others), By Application(Broadcasters Operators and Media, Education, Healthcare, Government and Others), Regional Insights, and Forecast From 2023 To 2030*, 2022. [Online]. Available: https://www.businessresearchinsights.com/market-reports/cloud-based-video-streaming-market-103175.

[17] X. Li, M. A. Salehi, and M. Bayoumi, "Vlsc: Video live streaming using cloud services," in *2016 IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom) (BDCloud-SocialCom-SustainCom)*, 2016, pp. 595–600. DOI: 10.1109/BDCloud-SocialCom-SustainCom.2016.93.

[18] X. Li, M. Darwich, M. A. Salehi, and M. Bayoumi, "Chapter four - a survey on cloud-based video streaming services," in ser. Advances in Computers, A. R. Hurson, Ed., vol. 123, Elsevier, 2021, pp. 193–244. DOI: https://doi.org/10.1016/bs.adcom.2021.01.003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0065245821000280.

[19] K. Stamos, G. Pallis, A. Vakali, D. Katsaros, A. Sidiropoulos, and Y. Manolopoulos, "CDNsim: A simulation tool for content distribution networks," *ACM Transactions on Modeling and Computer Simulation*, vol. 20, no. 4, 2010. DOI: 10.1145/1837277.1837282.

[20] T. Bektas, J.-F. Cordeau, E. Erkut, and G. Laporte, "Exact algorithms for the joint object placement and request routing problem in content distribution networks," *Computers & Operations Research*, vol. 35, no. 12, pp. 3860–3884, 2008. DOI: 10.1016/j.cor.2007.02.007.

[21] G. Pierre and M. Steen, "Globule: A collaborative content delivery network," *IEEE Communications Magazine*, vol. 44, no. 8, pp. 127–133, 2006. DOI: 10.1109/MCOM.2006.1678171.

[22] A. Tchernykh, J. M. Cortés-Mendoza, J. E. Pecero, P. Bouvry, and D. Kliazovich, "Adaptive energy efficient distributed voip load balancing in federated cloud infrastructure," in *2014 IEEE 3rd International Conference on Cloud Networking (CloudNet)*, 2014, pp. 27–32. DOI: 10.1109/CloudNet.2014.6968964.

[23] V. Mathew, R. K. Sitaraman, and P. Shenoy, "Energy-aware load balancing in content delivery networks," in *2012 Proceedings IEEE INFOCOM*, 2012, pp. 954–962. DOI: 10.1109/INFCOM.2012.6195846.

[24] M. Lin, A. Wierman, L. L. H. Andrew, and E. Thereska, "Dynamic right-sizing for power-proportional data centers," in *2011 Proceedings IEEE INFOCOM*, 2011, pp. 1098–1106. DOI: 10.1109/INFCOM.2011.5934885.

[25] J. He, Y. Wen, J. Huang, and D. Wu, "On the cost–qoe tradeoff for cloud-based video streaming under amazon ec2's pricing models," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 4, pp. 669–680, 2014. DOI: 10.1109/TCSVT.2013.2283430.

[26] P. Goudarzi, "Stochastic total cost of ownership optimization for video streaming services," *Telematics and Informatics*, vol. 31, no. 1, pp. 79–90, 2014, ISSN: 0736-5853. DOI: https://doi.org/10.1016/j.tele.2013.02.001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S073658531300004X.

[27] G. Pejman, "Dynamic total cost of ownership optimization for iptv service deployment," *Journal of Applied Sciences*, vol. 9, Apr. 2009. DOI: 10.3923/jas.2009.707.715.

[28] G. Ciccarella, R. Giuliano, F. Mazzenga, F. Vatalaro, and A. Vizzarri, "Edge cloud computing in telecommunications: Case studies on performance improvement and tco saving," in *2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC)*, 2019, pp. 113–120. DOI: 10.1109/FMEC.2019.8795351.

[29] A. Mazrekaj, I. Shabani, and B. Sejdiu, "Pricing schemes in cloud computing: An overview," *International Journal of Advanced Computer Science and Applications*, vol. 7, Feb. 2016. DOI: 10.14569/IJACSA.2016.070211.

[30] Elisa, *Elisa*, 2023. [Online]. Available: https://elisa.fi/.

[31] Elisa, *Elisa viihde*, 2023. [Online]. Available: https://elisaviihde.fi/.

[32] I. Shifting, *Salaries in finland*, 2023. [Online]. Available: https://itshifting.com/finland/salaries.

[33] Exascend, *Exascend | industrial ssd, enterprise storage, pro cinematography cards*, 2023. [Online]. Available: https://exascend.com/.

[34] eBay, *Ebay uk*, 2023. [Online]. Available: https://www.ebay.co.uk/.

[35] D. S. Oy, *Data systems*, 2023. [Online]. Available: https://www.data-systems.fi/.

[36] Dell, *Dell usa*, 2023. [Online]. Available: https://www.dell.com/en-us.

[37] Dustin, *Dustin | it-tarvikkeet ja elektroniikka netistä*, 2023. [Online]. Available: https://www.dustin.fi/.

[38] S. Electronics, *Smicro electronics*, 2023. [Online]. Available: https://smicro.eu/.

[39] Dell, *Dell poweredge r730 spec sheet*, 2023. [Online]. Available: https://i.dell.com/sites/doccontent/shared-content/data-sheets/en/Documents/Dell-PowerEdge-R730-Spec-Sheet.pdf.

[40] Dell, *Dell poweredge r740 spec sheet*, 2023. [Online]. Available: https://i.dell.com/sites/csdocuments/Shared-Content_data-Sheets_Documents/en/poweredge-r740-spec-sheet.pdf.

[41] Dell, *Dell poweredge r7515 spec sheet*, 2023. [Online]. Available: https://i.dell.com/sites/csdocuments/Product_Docs/en/poweredge-r7515-spec-sheet.pdf.

[42] APC by Schneider Electric, *Apc netshelter sx server rack enclosure 42u black 1991h x 750w x 1200d mm*, https://www.apc.com/shop/fi/en/products/APC-NetShelter-SX-Server-Rack-Enclosure-42U-Black-1991H-x-750W-x-1200D-mm/P-AR3350, Accessed: 2023-07-20.

[43] R. Components, *Apc 42u server rack*, https://fi.rsdelivers.com/product/apc/er8220/apc-42u-server-rack/2387226, Accessed: 2023-07-20.

[44] Statista, *Annual prime rents for industrial properties in finland by city*, 2019. [Online]. Available: https://www.statista.com/statistics/1154962/annual-prime-rents-for-industrial-properties-in-finland-by-city/.

[45] Sysracks, *How to choose the appropriate height of a server rack*, https://sysracks.com/blog/how-to-choose-the-appropriate-height-of-a-server-rack, Accessed: 2023-07-20.

[46] 42U, *42u cabinets*, https://www.42u.com/42U-cabinets.htm, Accessed: 2023-07-20.

[47] T. Lite, *42u smartrack wide standard depth rack enclosure cabinet*, https://tripplite.eaton.com/42u-smartrack-wide-standard-depth-rack-enclosure-cabinet-doors-side-panels-shock-pallet-packaging~SR42UBWDSP1, Accessed: 2023-07-20.

[48] Kstar, *How to choose a server cabinet*, https://www.kstar.com/indexproblem/17427.jhtml, Accessed: 2023-07-20.

[49] S. Finland, *Price of electricity by type of consumer*, https://pxdata.stat.fi/PxWeb/pxweb/en/StatFin/StatFin__ehi/statfin_ehi_pxt_13rb.px/table/tableViewLayout1/, Accessed: 2023-07-20.

[50] Google, *Google sustainability*, 2023. [Online]. Available: https://sustainability.google/progress/energy/.

[51] G. Developers, *Codelab*, 2023. [Online]. Available: https://codelabs.developers.google.com/mediacdn-ls-codelab.

[52] U. Streaming, *Unified streaming documentation*, 2023. [Online]. Available: https://docs.unified-streaming.com/installation/license.html.

[53] CastLabs, *Drmtoday*, 2023. [Online]. Available: https://castlabs.com/free-trials/drmtoday/.

A APPENDIX: CDN caching

In this this appendix, some scenarios of user requests of CDN is shown.

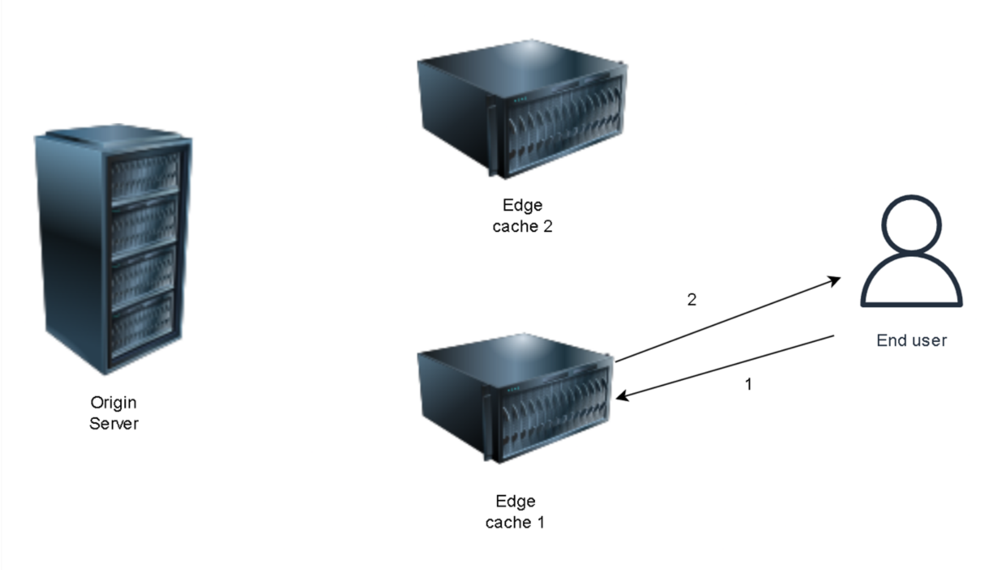

Figure 6: Case 1: The content requested by the user is available on the closest Edge (1) to the user.

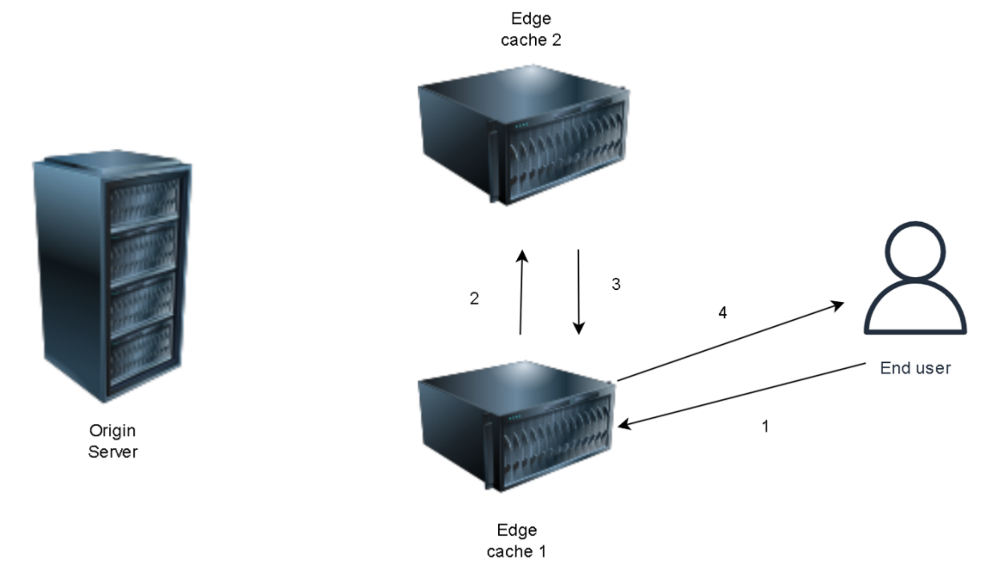

Figure 7: Case 2: The content is not available on the closest Edge (1) to the user. In this case, Edge 1 requests the content from Edge 2.

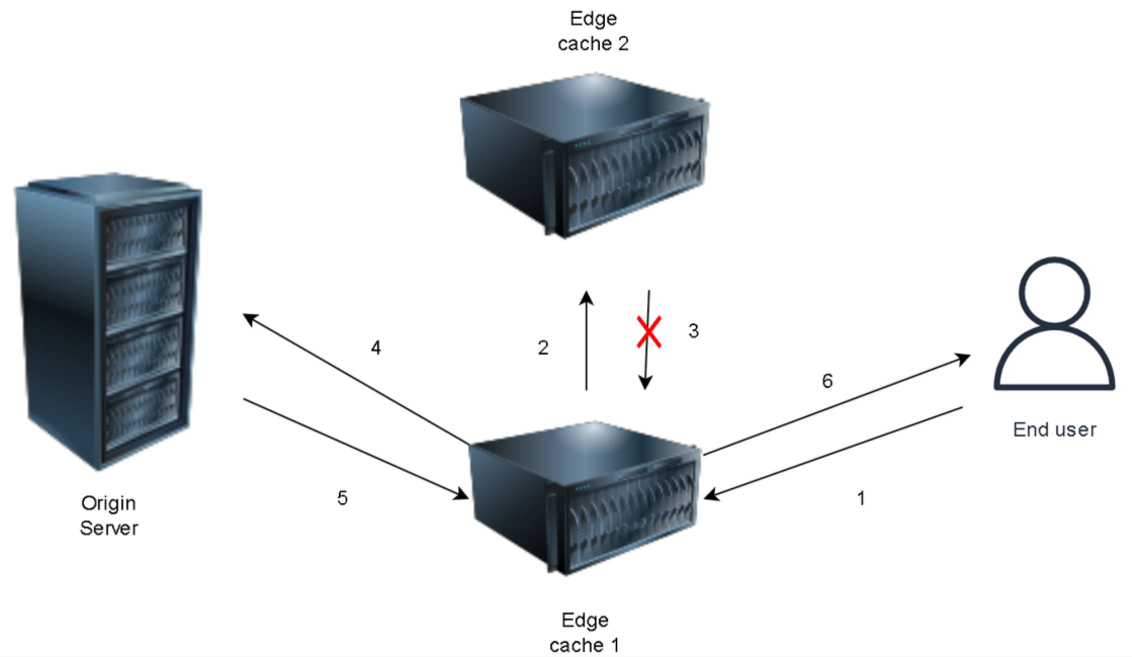

Figure 8: Case 3: The content is not available on the closest Edge (1) to the user. When, requested from Edge 2, Edge 1 did not find the requested content. In this case, Edge 1 fetches the content from the Origin Server.

B APPENDIX: Simulation Information

To create the simulation, we will execute the following tasks

- Create a project for Google Cloud.
- Make the project allow-listed to Media CDN as Media CDN access is restricted.
- Start Cloud Shell which will be used to run the configuration commands.
- Use the latest version of Google Cloud SDK version
- Setup environment variables that will be used in the configuration: Project ID, Project Number, User Name, Region, Input ID , Channel ID.
- Enable the required Service APIs, which are: Network Services API, Certificate Manager API, Livestream API, Media CDN Edge Cache API, Compute Engine API.
- Create a Cloud Storage Bucket with a globally unique name and make it publicly accessible to other resources. This will be used to store the transcoded video files.
- Setting up the Live Streaming API: two components are required for this. **The Live Streaming Input** serves as an endpoint to receive the live signal from the encoder (FFmpeg). It is utilized to define various stream configurations, including input resolution, input type, and video cropping. **Live Streaming Channel** is a resource responsible for receiving the input stream via an input endpoint, performing transcoding of the input stream into multiple renditions, and subsequently publishing the resulting live streams in specific formats at the designated location. The channel allows for the inclusion of both primary and backup input streams within its operation.
- Configure Media CDN: this includes creating resources, namely, **Edge Cache Origin** and **Edge Cache Service.**
- Set up Unified Origin using the free trial on a VM instance. Connect it to the Storage bucked to fetch the transcoded content and apply on-the-fly encryption on it.
- Get the DRM API using DRMToday free trial.
- Install FFmpeg and send the live signal to the Live Streaming API input endpoint.
- Verify the live signal is getting transcoded by the Streaming API Channel.

- Verify that the Edge Cache Service instance works by monitoring the cache miss/ cache hit status.
- Download VLC on a local device and use the and use it to stream transcoded content.

To execute the steps in the simulation, this CodeLab [51] was used as a reference. The used gcloud CLI version is 438.0. To use Unified Origin, the license key should be obtained from [52]. Once we obtain the UspLicenseKey, we can use it to download the trial version. To use DRMToday, we need to get the merchant UUID. For this we need to get a free trial access from [53]