MEDIA STREAMING IBM AND CLOUD VIDEO DEFINING MEDIA STREAMING

It is a method of delivering multimedia elements usually video or audio from a datastreaming service provider to the users. The protocols it uses includes: · HTTP (Hypertext Transfer Protocol) · TCP/IP (Transmission Control Protocol / Internet Protocol) · HTML Protocols Streaming media is either video or audio content sent in compressed form over internet, rather than saved to hard-disk. Because of media streaming, users does not have to download a file to play and view it. Users can fast-forward, rewind, pause, stop that streaming video what they can with a downloaded file. Cloud technology has a number of advantages that aid content providers to deliver systems. A large storage facility is available for cloud computing providers for maintaining libraries along with a high computation power for streaming servers. It also provides an engine for encoding, decoding and transcoding content. Audio streaming was the first to gain widespread fame as a media application on Internet. The streamed audio gets expanded its field using cloud computing to make radio. VoIP is a form of audio streaming based on specialized protocol. Some of the advantages of streaming media are: · It makes possible for cloud users to view interactive applications like video, searching them and personalized playlist. · It provides the content makers with more control over his intellectual property as the video file isn't stored on the viewer's computer. · It provides an efficient use of bandwidth. This is because the transferred file is the only part that is being watched. · It allows content delivers for monitoring what visitors are watching. Encoding.com offers a great example of how the cloud can be leveraged for providing service on demand. This site advertises itself as the world's most popular encoding/transcoding service. Encoding.com also provides static picture file conversion, audio, and video file conversion. Most of the cloud business focuses on streaming file formats used for audio and video work. There is the most popular conversion provided by that site. These are: · AVI to WMV · Any video to 3GP · MP4 to WMV · FLV to MPG · 3GP to WMV · WMV to MPEG · AVI to MPEG-4 · VP6 to MPEG · 264 to MPEG · VP6 to AVI · 264 to AVI · FLV to MOV · MP4 to 3GP etc...

AgilityandScaleForStreamingInCloudComputing

Streamingplatformsunderstandthat theywillnotalwaysneedthesame amountofbandwidthandspeedto keeptheirservicesrunningsmoothly and efficiently.Insteadofpayingfor morespacethanaplatform mayneed,itmakes sensetoutilizethecloudtoscaleupor downdependingontheneedsof the business.Havingbetter controlofscaleandbeingableto react swiftly toany changesensures that thestreamingprocessisbothcost-effectiveandperforms at itsbest. Thisensuresthat thestreamingexperienceisthebest that itcanbe, whichcanbeespecially important for subscription-basedplatforms. Evensmall businessescanstream videowithefficiency thanksto thescalability option. High-qualityservicesarenolonger reservedsolelyfor largerbusinesses that canaffordtheextraspace.Itmaymeanmorecompetitionfor bignameslike Netflix,but cloudcomputingbringsmorechoicesto consumers.

TheHighestPotentialForStorageandData

Alongsidethebenefitsofstreamingonlinecomesasetofuniquetechnical challenges.Streamingvideo means that largeamountsofdataarebeing transmitted.This couldresult inlatencyissues whichmeanonly onedreaded thing:buffering.Weallknowthatalongbufferingtimecanruineventhebest viewingexperiences.

Similarly,cloudcomputingallowsstreamingplatformsto leveragestorageand datato ensurethehighest viewingquality for consumers. Thisbecomes particularlyimportant incasesoflivestreaming. Nomatter ifit'saconference calloralivestream video,no viewer wantsto experiencealagintheir streaming.

Inconclusion, video streamingincloudcomputingis somethingthatallowsus to experiencehigh-qualitystreamingatamorecompetitiveprice. Thequality of thestreamingisvastly improved, whichonlyenriches theexperienceof theuser. Cloudtechnology willno doubtonlycontinuetoimprovethestreaming experienceandnewinnovationswillimproveuser experiencetomatch. VEXXHOSTCloudSolutions AsareputedIaaSprovider, weensurethatourclientsget thebest typeofcloud services for their data.AtVEXXHOST, weprovidecloudsolutionsfor amultitude ofclients worldwide.WeprovideOpenStack-basedclouds,including public clouds anddedicatedandhighly secure privatecloud environments,ensuring utmost securityandagility. Takeadvantageofour limited-timedeal just to setupaone-time, OpenStack-basedprivateclouddeployment -at50%off!Thecloud willbe runningonthelatestOpenStack release,Wallaby, whichallowsyoutorun KubernetesandVMsinthesameenvironment,andcanbedeployedinyour own datacenters withyour hardware. Furthermore,all these will bedeployedand testedinunder amonth! 5 TOP CLOUD COMPUTATING Out of 347, the Global Startup Heat Map highlights 5 Top Cloud Computing Startups impacting the Media Industry Startups such as the examples highlighted in this report focus on video inventory management, brand engagement, and content rendering. While all of these technologies play a significant role in advancing media, they only represent the tip of the iceberg. This time, you get to discover five hand-picked cloud computing startups impacting the media industry. The Global Startup Heat Map below reveals the geographical distribution of 347 exemplary startups & scaleups we analyzed for this research. Further, it highlights five media startups that we hand-picked based on scouting criteria such as founding year, location, funding raised, and more. You get to explore the solutions of these five startups & scaleups in this report. For insights on the other 342 cloud computing solutions impacting media, get in touch with us. ClouPlay improves Media Experience Management Founding Year: 2019 Location: Istanbul, Turkey Partner with ClouPlay for Media

Content Organization Turkish startup ClouPlay provides media experience management. The startup's cloud-based platform, , manages, distributes, and broadcasts live media on the channels. It increases the speed and efficiency of media encoding servers. The startup's other product, , collects real-time data from different channels which enables broadcasters to make better decisions. It also assists media agencies and content creators in organizing media content, encoding live videos at higher speeds, and facilitating post-production cStreamingisamethodof viewingvideoor listeningtoaudio content withoutactuallydownloadingthemediafiles. Streamingperformancecanbeimproved,andbuffering timereduced,if theowner of thefilesusesaCDN. Overview of Stream Data Processing Today's data is generated by an infinite amount of sources - IoT sensors, servers, security logs, applications, or internal/external systems. It's almost impossible to regulate structure, data integrity, or control the volume or velocity of the data generated. While traditional solutions are built to ingest, process, and structure data before it can be acted upon, streaming data architecture adds the ability to consume, persist to storage, enrich, and analyze data in motion. As such, applications working with data streams will always require two main functions: storage and processing. Storage must be able to record large streams of data in a way that is sequential and consistent. Processing must be able to interact with storage, consume, analyze and run computation on the data. This also brings up additional challenges and considerations when working with legacy databases or systems. Many platforms and tools are now available to help companies build streaming data applications. Examples Some real-life

```
"); } //getvideostats(about 61MB) const videoPath= "bigbuck.mp4"; const videoSize= fs.statSync("bigbuck.mp4").size; //ParseRange //Example: "bytes=32324-" const CHUNK_SIZE =10 **6;//1MB const start =Number(range.replace(/\D/g, "")); constend=Math.min(start + CHUNK_SIZE, videoSize- 1); //Createheaders const contentLength=
```

end- examples of streaming data include use cases in every industry, including real-time stock trades, up-to-the-minute retail inventory management, social media feeds, multiplayer games, and ride-sharing apps. For example, when a passenger calls Lyft, real-time streams of data join together to create a seamless user experience. Through this data, the application pieces together real-time location tracking, traffic stats, pricing, and real-time traffic data to simultaneously match the rider with the best possible driver, calculate pricing, and estimate time to destination based on both real-time and historical data. In this sense, streaming data is the first step for any datadriven organization, fueling big data ingestion, integration, and real-time analytics. Batch Processing vs Real-Time Streams Batch processing requires data to be downloaded as batches before it can be actionable, whereas streaming data allows for simultaneous, real-time processing, storage, and analytics. With the complexity of today's modern requirements, legacy batch data processing has become insufficient for most use cases, as it can only process data as groups of transactions collected over time. Modern organizations need to act on up-to-the-millisecond data, before the data becomes stale. Being able to access data in real-time comes with numerous advantages and use case . BASIC EXAMPLE: const videoEl = document.querySelector('video'); const mediaSource = new MediaSource(); video.src = URL.createObjectURL(mediaSource); mediaSource.addEventListener('sourceopen' , () => { const mimeString = 'video/mp4; codecs="avc1.42E01E, mp4a.40.2"'; const buffer = mediaSource.addSourceBuffer(mimeString); buffer.appendBuffer(/* Video data as `ArrayBuffer` object. */) }); Part 1:Setupproject You'llneedtoinstallNodeJSandrun: mkdir http-video-stream cdhttp-video-stream npm init npm install--saveexpressnodemon Part 2:index.html Weneedtocreatea HTML5Videoelement,andset the source as "/video" , whichis where server'sendpoint is.

Part 3:index.js Nowlets setupour node serversothaton "/" endpoint itserves our index.htmlpage. constexpress = require("express");

constapp= express(); app.get("/" ,function(req, res){ res.sendFile(__dirname+ "/index.html"); }); app.listen(8000,function(){ console.log("Listeningonport8000!"); }); Part 4:package.json—Runour server Add a start script to your package.json sothatwe canrunourserver using npmstart command. There'smore inyour package.json filebut Ijustwantyoutocopy this startscript.Ituses nodemon torun index.js andrestartsthe server every time you savethe index.js filesoyou don'tneed torestart theserver yourself! { "scripts":{ "start": "nodemonindex.js" } } Nowyoushouldbeable torun npm start andseeour apprunningonport8000. Openyourbrowser andgo to http://localhost:8000 tosee ifitworked. Part 5:index.js(Again) We're almost there! Here's the "/video" endpoint for ourserver. //intheimportsabove const fs =require("fs"); app.get("/video" ,function(req, res){ //Ensurethereisarangegivenfor thevideo const range= req.headers.range; collaboration

Weneedtoadaptourmethodsfor thesedifferent,often fluctuatingconditions toprovide ahigh-qualityexperience for existingmembersaswellastoexpandinnewmarkets. AtNetflix, weobservenetworkanddeviceconditionsaswellasaspectsof theuser experience(e.g., videoquality) we wereabletodeliver forevery session,allowingus toleveragestatisticalmodeling andmachine learninginthisspace.A previouspost described howdatascience is leveragedfor distributingcontentonour servers worldwide.Inthispostwedescribesome technical challengeswefaceonthedeviceside.
Networkqualitycharacterizationandprediction
Networkqualityisdifficult tocharacterize andpredict. Whilethe averagebandwidthandroundtriptime supportedbyanetwork arewell-knownindicatorsofnetworkquality,other characteristics suchasstabilityandpredictabilitymake abig differencewhenitcomes tovideostreaming.Aricher characterizationofnetworkqualitywouldproveusefulfor analyzingnetworks (for targeting/analyzingproduct improvements),determininginitialvideoqualityand/or

adaptingvideoquality throughoutplayback(moreonthatbelow).
Beloware afewexamplesofnetworkthroughputmeasured duringreal viewingsessions.Youcanseetheyarequitenoisy andfluctuatewithina widerange. Can wepredictwhat throughputwilllooklikeinthenext15minutesgiventhelast15 minutesofdata? Howcan we incorporatelonger-termhistorical informationabout thenetworkanddevice?Whatkindofdata canweprovide from the server thatwouldallowthedeviceto adaptoptimally? Evenif wecannotpredictexactlywhena networkdropwillhappen(thiscouldbeduetoallkindsof things, e.g.amicrowaveturningonorgoingthroughatunnel while streamingfromavehicle),can we at leastcharacterize the of throughput thatwe expect toseegiven historicaldata?
Sinceweareobservingthese tracesatscale, thereis opportunity tobringtobearmore complexmodels thatcombine temporalpatternrecognition withvariouscontextualindicators tomakemoreaccuratepredictionsofnetworkquality.
Oneusefulapplicationofnetworkpredictionistoadaptvideo qualityduringplayback,which wedescribeinthe following section.
Videoqualityadaptationduringplayback
Moviesandshowsareoftenencodedatdifferentvideoqualities tosupportdifferentnetworkanddevicecapabilities.Adaptive streamingalgorithmsareresponsibleforadaptingwhichvideo qualityis streamedthroughoutplaybackbasedonthecurrent networkanddeviceconditions (see here foranexampleofour colleagues' researchinthisarea).Thefigurebelowillustrates thesetupfor videoqualityadaptation.Canweleveragedatato determine the videoquality thatwilloptimize thequalityof experience.Thequalityof experiencecanbemeasuredin severalways,includingtheinitialamountof time spentwaiting for videotoplay, theoverall videoqualityexperiencedby theuser, thenumberof timesplaybackpausedtoloadmorevideointothe buffer (" rebuffer"),andthe amountofperceptible fluctuation inqualityduringplayback. Thesemetricscantradeoffwithoneanother:

wecanchooseto beaggressiveandstreamveryhigh-qualityvideobut increase theriskofarebuffer.Or wecanchoosetodownloadmorevideo upfrontandreducetherebuffer riskat thecostofincreasedwait time.The feedbacksignalofagivendecisionisdelayedand sparse.Forexample,anaggressiveswitchtohigher qualitymay nothaveimmediaterepercussions,butcouldgraduallydeplete thebufferandeventuallyleadtoarebuffereventonsome occasions.This “ creditassignment” problem isawell-known challengewhenlearningoptimalcontrolalgorithms,and machine learningtechniques(e.g., recentadvances in reinforcement learning)havegreatpotential totackle these issues. Predictivecaching Anotherareainwhichstatisticalmodelscanimprovethe streamingexperienceisbypredictingwhatauser willplay in order tocache(partof)itonthedevicebefore theuserhitsplay, enablingthe videotostart fasterand/oratahigherquality.For example,wecanexploit the fact thatauser whohasbeen watchingaparticularseriesis very likelytoplay thenext unwatchedepisode. Bycombiningvariousaspectsof their viewinghistory together withrecentuser interactionsandother contextual variables, onecanformulate thisasasupervised learningproblemwherewe want tomaximizethemodel’ s likelihoodofcachingwhat theuseractuallyendedupplaying, whilerespectingconstraintsaroundresourceusagecoming fromthecache sizeandavailablebandwidth.Wehaveseen substantial reductions inthetimespentwaitingfor videoto startwhenemployingpredictivecachingmodels. Deviceanomalydetection Netflix operatesonover athousanddifferent typesof devices, rangingfromlaptops totabletstoSmartTVstomobile phones tostreamingsticks. Newdevicesareconstantlyentering intothisecosystem,andexistingdevicesoftenundergoupdates totheir firmwareor interactwithchangesonourNetflix application.Theseoftengo withoutahitchbutat this scaleit is notuncommontocause aproblemfor theuserexperience—e.g., theappwillnotstartupproperly,orplayback willbeinhibitedor degradedinsomeway. Inaddition, therearegradual trendsin devicequality thatcanaccumulateover time.Forexample,a chainofsuccessiveUI changesmay slowlydegradeperformance onaparticulardevicesuchthat itwasnot immediately noticeable afteranyindividualchange. Detectingthese changesisachallengingandmanually intensive process.Alertingframeworksareauseful toolfor surfacing potentialissuesbutoftentimes it istricky todetermine the right criteriafor labelingsomethingasanactualproblem.A“ liberal” trigger willendupwithtoomanyfalsepositives, resultingina large amountofunnecessarymanualinvestigationbyourdevice reliabilityteam, whereasavery strict triggermaymissoutonthe realproblems. Fortunately,wehavehistoryonalerts thatwere triggeredas wellas theultimatedetermination(madebya human)ofwhetherornot theissuewasinfact realand actionable.We canthenusethis totrainamodel thatcanpredict thelikelihoodthatagivensetofmeasuredconditions constitutesarealproblem. Even when we’ reconfidentwe’ reobservingaproblematic issue,it isoftenchallengingtodeterminethe rootcause.Wasit duetoafluctuationinnetworkqualityonaparticular ISP or ina particular region?AninternalA/Bexperimentorchange that was rolledout.Afirmwareupdateissuedby thedevice manufacturer.Is the changelocalizedtoaparticulardevice grouporspecificmodelswithinagroup?Statisticalmodeling canalsohelpusdeterminerootcauseby controllingfor various covariates.

Byemployingpredictivemodelingtoprioritizedevicereliability issues,we’ vealreadyseenlargereductionsinoverallalert volumewhilemaintaininganacceptablylowfalsenegative rate, which we expect todrivesubstantialefficiencygains for Netflix’ sdevicereliabilityteam.