

AWS Architecture Monthly



**May 2020
Education**

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers, or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

© 2020 Amazon Web Services, Inc. or its affiliates. All rights reserved.

Editor's note

One of the missions of the education industry is to educate the next generation of the industry-ready workforce. Whether K-12, higher education, or continuing education, enabling teachers and professors to effectively deliver curriculum and improve student performance is a goal of Education Technology (EdTech) and learning companies. Two trends for AWS use cases in education are: 1) accessible remote learning; and 2) remote collaboration. For brevity, there are other innovation trend areas in education that Yuriko Horvath didn't focus on in our *Ask an Expert* interview despite their importance. Use cases around learning accessibility, student performance, and campus experience have taken advantage of Amazon Alexa, Amazon Lex, and a variety of AWS technology areas including artificial intelligence (AI) and machine learning, data lakes, analytics, and mobile development. To dive deep into a wider range of education use cases, I invite everyone to look at our AWS Education blog (<https://amzn.to/AWS-ed-ed-blog>).

I hope you'll find this edition of Architecture Monthly useful, and my team would like your feedback. Please give us a star rating and your comments on the Amazon Kindle page (<https://amzn.to/Kindle-magazine>). You can view past issues at <https://aws.amazon.com/whitepapers/kindle/>, and you can reach out to aws-architecture-monthly@amazon.com anytime with your questions and comments.

In May's Education issue:

- **Ask an Expert:** Yuriko Horvath, Manager, Education Solutions Architect
- **Blog:** How to Build a Chatbot for Your School in Less Than an Hour
- **Case Study:** Virginia Tech: Building Modern Analytics on Amazon Web Services
- **Solution:** Video on Demand on AWS
- **Whitepaper:** Teaching Big Data Skills with Amazon EMR

Annik Stahl, Managing Editor

Ask an Expert:

Yuriko Horvath
Manager, Education Solutions Architect

What are the general architecture pattern trends for educational resources in the cloud?

Access to content and learning resources anytime, anywhere, and in multiple formats is key to accessible remote learning. To convert video, audio, and image content into searchable resources, EdTechs and learning companies are using AWS media, machine learning, and AI services. These are hosted on the backbone of Amazon Simple Storage Service (S3)

(<https://aws.amazon.com/s3/>) and Amazon CloudFront

(<https://aws.amazon.com/cloudfront/>) as global content delivery networks.

To provide quality student experiences on mobile devices, these learning experts are making video recordings of lectures that are transcoded by AWS Elemental Media Convert

(<https://aws.amazon.com/mediaconvert/>). They are also using Amazon Transcribe

(<https://aws.amazon.com/transcribe/>) and Amazon Rekognition

(<https://aws.amazon.com/rekognition/>) to convert lectures and library photograph archives into searchable catalogs of content.

For STEM (science, technology, engineering, and math) subjects, reinforcing lectures through labs, homework, and projects is key to cementing lessons into practice. In the past, K-12 schools and universities concentrated on having on-campus computer labs with specific software installed, enabling students to work on projects and homework. With Amazon AppStream 2.0 (<https://aws.amazon.com/appstream2/>) and Amazon WorkSpaces (<https://aws.amazon.com/workspaces/>), they can stream the same lab-installed software to any student's browser in the convenience of his or her home or dorm. Schools and EdTechs have worked to integrate these end user compute technologies with their directory services, and they automate creation on environments for cohort classes.

When putting together an AWS architecture to solve business problems specifically for education customers, do you have to think it all differently?

Thinking big around automation and scale needs to be top of mind when building for education customers. Schools rarely differentiate themselves based off IT infrastructure, and within a university, there are multiple departments with similar needs. If a solution fits one application, chances are good that multiple opportunities exist to use the same solution within the same institution and across peer institutions. With the cyclical nature of the

school year, every deployed solution needs to be ready to scale out quickly in response to changing demands.

What are some questions education customers need to ask themselves before considering AWS?

When school systems or institutions begin to consider moving to the cloud, they need to go beyond modernizing one application, one lab, or one research project. Schools, EdTechs, and learning companies are busy focusing on their mission of bettering students, so they often don't take time to reflect and evaluate where they are spending unnecessary effort around IT infrastructure. Contrary to intent, this be slowing them down from focusing on their goal.

When evaluating AWS, schools should have a sense of IT infrastructure priorities. They need to consider the value of maintaining infrastructure in an on-campus data center versus moving to the cloud. Schools should be finding project priorities that allow them to realize major savings in cost and management:

- Where do they have similar use cases across workloads, storage, and across departments?
- Would there be workforce advantages to managing these use cases together in automation and maintenance?
- Would there be cost savings with tiered pricing and discounts?
- Are there major events coming up about which they need to make decisions, such as renewing the lease of a data center, renewing software licenses, or refreshing hardware?

Do you see different trends within the industry in cloud versus on-premises?

Data centers are a reality with many school districts and universities. While some institutions may be scaling back the number of data centers they support, there will still be a need for on-premises connectivity. It's slower to scale within data centers, and our Solutions Architects often work to prioritize what use cases are ideal to keep on-premises versus move to AWS:

- Research workloads that require massive short-term scale like high performance computing (HPC), parallel clusters, and MapReduce
- Workloads that take advantage of AWS's pace of innovation around analytics, AI, and machine learning
- IT Infrastructure that needs to scale to meet the school year demands like virtual desktops
- Disaster recovery and archiving of enterprise applications and storage

About our expert



Yuriko Horvath leads a team of Amazon Web Services Solutions Architects focused on universities and research. Her team has helped large universities migrate research applications and data centers to the cloud. Yuriko has worked as a web application developer for over 20 years and started her cloud journey back in 2009 helping innovate and modernize the news industry through AWS adoption.

Real-world example

Fife School Districts Use Amazon AppStream 2.0 to Provide Equitable Access

By using Amazon AppStream 2.0 (<https://aws.amazon.com/appstream2/>), Fife School District's students can now access applications from anywhere, anytime, on any device. Fife Public Schools in Washington aspires to be a top-tier learning organization that prepares students for college, career, and life. Fife School District provides a device for every kid in the district. While the kids automatically got access to some applications, there were certain programs they couldn't access especially those used in CTE (Careers in Technical Education) Classes. In addition to providing equitable access with Amazon AppStream 2.0, the district turned labs back into learning spaces, which saved the district money they would have otherwise spent on building costs.

<https://amzn.to/AWS-ed-fife>

Blog:

How to Build a Chatbot for Your School in Less Than an Hour

By Brooke Gleason

Available online at: <https://amzn.to/AWS-ed-build-chatbot>

Your students have questions and you want to get them answers quickly and easily. With the rise in remote learning and education, virtual solutions such as chatbots are one way to make information available on-demand—anywhere and anytime.

Did you know that you can create a Q&A chatbot for your educational institution in less than an hour? To get started, check out the AWS Q&A Chatbot Self-Paced Guide (<https://amzn.to/AWS-ed-chatbot-guide>) that walks you through all the needed steps to set one up.

You'll learn how to:

- Deploy the chatbot in your AWS account
- Set up the chatbot environment with a sample question bank
- Enhance the chat solution with voice by adding an Amazon Alexa experience
- View usage analytics
- Enable the solution to support multi-language capability

Many educational institutions have already built their own chatbots. At Oklahoma State University–Oklahoma City (OSU-OKC), staff members were handling questions from both phone calls and walk-in students, which have coinciding peak periods. Up to 50% of calls were going unanswered, and many of the questions were simple to answer. To free up their staff to focus on more personalized and in-person interactions, OSU-OKC used QnA Bot and Amazon Connect to build a solution populated with frequently asked questions. They created a multi-channel question and answer bot that's used via web chat, mobile chat, and voice, for commonly asked questions with intelligent multi-part interaction. The voice Lexbot interaction seamlessly escalates to live staff member for unanswered questions or more complicated issues.

"Integrating the technology into our hybrid system and processes is helping us automate routine interactions enabling human time to be reinvested into the student experience. This technology is proven in other industries so we were eager to discover its impact on higher education," said Brad Williams, President OSU-OKC.

You can also check out these short how-to videos below to get started and follow along with the self-paced guide (<https://amzn.to/AWS-ed-chatbot-guide>).

How-to videos

Deploy a Q&A Chatbot using AWS CloudFormation

<https://amzn.to/AWS-ed-chatbot-1>

Using the Q&A Chatbot content designer, question bank, and trying the chatbot experience

<https://amzn.to/AWS-ed-chatbot-2>

How to enable multi-language support in Q&A Chatbot

<https://amzn.to/AWS-ed-chatbot-3>

Creating an Alexa experience from Q&A Chatbot

<https://amzn.to/AWS-ed-chatbot-4>

The header graphic features a dark blue background with several overlapping, lighter blue geometric shapes, primarily triangles and polygons, creating a modern, abstract design. The text is centered and white.

Case Study:

Virginia Tech's Experience Building Modern Analytics on AWS

By Brent Kramer

Available online at: <https://amzn.to/AWS-ed-virginia-tech>

Virginia Polytechnic Institute and State University (Virginia Tech) wanted to build a modern data warehouse to complete new requests and quickly answer difficult questions in order to make more informed decisions. To do this, we turned to Amazon Web Services (AWS).

We were looking for a way to build forecasting models faster so we could quickly react to changing conditions. Most tables in our Online Transactional Processing (OLTP) databases do not track the history of data, which meant we had no way of understanding what the data looked like over time. When we wanted to see if a forecasting model had value, we had to start capturing the history in a data mart, which could take three months to gather requirements and develop. Once developed, we would have to wait another six to nine months to capture enough data to derive any value.

With AWS Database Migration Service (AWS DMS), we can capture every table's history, which allows us to forecast. Instead of the model taking 12 months to mature before I could derive any value from it, I can now see immediate value by ruling out models that do not provide the necessary information.

To improve the learning experience, Virginia Tech has been purchasing industry-leading products, many of which are Software-as-a-service (SaaS) products without access to the underlying database. The backbone of the applications that run Virginia Tech is an Oracle appliance. For most of the student and administrative data, this is our system of record. For reporting, we built a custom data warehouse in Oracle using Talend as our extract, transform, load (ETL) tool.

Getting started

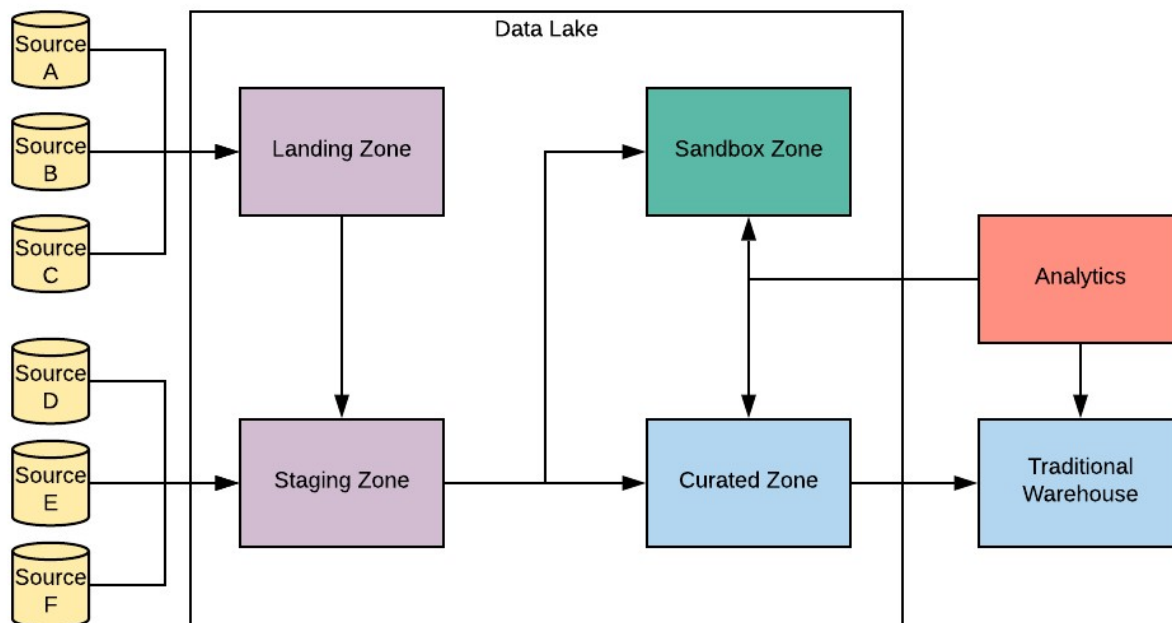
Before we started the data lake project, we did a six-month proof of concept with senior technology team members across multiple departments and skillsets. The data scientist team at Virginia Tech was the primary end user. They were building a new budgeting model that required data from many sources.

Once the proof of concept was completed and we had approval and funding for a production data lake, we assembled a team that included a cloud architect and senior data engineer. To get started with different AWS tools, we attended an AWS Data Lab. This AWS Data Lab helped us to accomplish more in a week than we probably could have over several months.

The framework

The AWS tools are easy to use. For example, after building a framework and figuring out how to ingest data from Oracle, the new sources and tables can be set up as inserting records in a JavaScript Object Notation (JSON) configuration file or Amazon DynamoDB. With AWS, if you build your framework correctly, the need for coding can be eliminated.

The initial source systems included were an Oracle database and two sets of application program interfaces (APIs). The below diagram is how we architected the different zones in our data lake in order to move different types of data. This framework helps us with data governance and security.



WS DMS: To get data out of our Oracle database and into our Staging Zone, we use AWS DMS. Once we had AWS DMS set up, adding new tables to the sync became simple. This is a near real-time replication service. Once running, we rarely see a record take more than 10 seconds to make its way to AWS.

In the most recent release of AWS DMS (3.1.4), there is an option to automatically add the action taken (insert, update, delete) and the transaction log timestamp for when it happened. Many of our tables in our databases are either undated or have unreliable dates recorded. These two features help make historical reporting possible and help us to create the curated zone.

AWS Glue: Although we use AWS Glue to run our jobs, the code consists of a simple, 20-line Python script. This is how we kept the process dynamic. When moving data from the staging zone to the curated zone, we take the most recent record and transform the data into Parquet files. When we need to add a new table, we insert a new entry in the JSON, and the Python code knows to begin processing the file. The dates and action indicators that AWS DMS creates make this all possible. We process 100 tables using the same 20 lines of Python.

Amazon RedShift: This is our final structured reporting layer. AWS Glue solves our need to get data from the Curated Zone into Amazon Redshift. We use the AWS Glue crawler to create Amazon Athena (Athena) tables, which allows us to do basic joins and SQL transforms. This is where the standard ETL comes in and where custom code for each dimension table is needed.



Next steps

We run reports with Amazon Redshift. The initial value we see is combining demographic data in Amazon Redshift with web log data by using Amazon Redshift Spectrum. The ability to understand how students use learning tools is valuable.

Currently, we are building out our Amazon Redshift data marts. The more dimensions we add to the database, the more value we can derive from the web logs.

We are also reviewing AWS Step Functions to streamline our AWS Glue jobs. Now that our data in AWS is structured in an optimal way for machine learning (ML) tools, we are looking to build prescriptive models to help us make better decisions that we otherwise aren't able to see immediately.

Having a strong understanding of AWS DMS, AWS Glue, Athena, and Amazon Redshift helped us get our project moving, and this will benefit us when we start using AWS Lake Formation.

Building our data pipeline in AWS made us rethink many best practices that typical data warehouse developers have relied on for many years. We look forward to expanding our data lake with more sources and features over the next several years.

Real-world example

JD Power: Building a Rankings & Insights Engine with Data Lakes & Machine Learning on AWS

Sandeep from JD Power tells us how the company built a data analytics solution on AWS called AI Cloud. JD Power processes terabytes of data performing customer satisfaction research on thousands of product categories producing product insights and rankings in industries such as automotive, healthcare, and electronics. Learn how to build a data lake on Amazon S3 pulling in data from multiple inputs, such as FTP, Amazon S3, and Amazon RDS. Once the data is centrally located, Sandeep shows us how to build catalogs with AWS Glue and custom crawlers running on AWS Fargate. You will also learn how to use Amazon Athena and Amazon Redshift for analytics and machine learning to draw insights from these datasets.

<https://amzn.to/AWS-ED-JD-Power>



Solution:

Video On Demand on AWS

Available online at: <https://amzn.to/AWS-ed-video-on-demand>

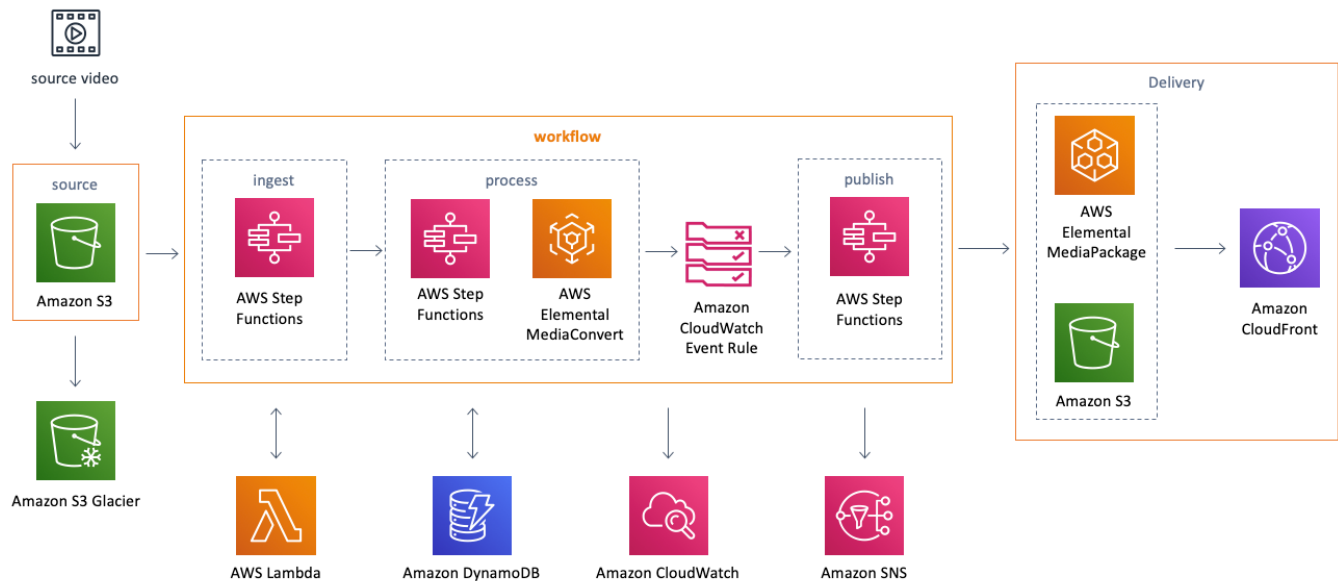
What does this AWS Solutions Implementation do?

Video on Demand on AWS automatically provisions the AWS services necessary to build a scalable, distributed video-on-demand workflow. The video-on-demand solution ingests metadata files and source videos, processes the videos for playback on a wide range of devices, stores the transcoded media files, and delivers the videos to end users through Amazon CloudFront.

Version 5.0 of the solution adds AWS Elemental MediaPackage functionality, and uses the most up-to-date Node.js runtime. Version 4.2 uses the Node.js 8.10 runtime, which reaches end-of-life on December 31, 2019. To upgrade to version 5.0, you must deploy the solution as a new stack. For customers who do not want to use the new functionality, you can update your existing stack to version 4.3. Version 4.3 keeps the same functionality as version 4.2 but uses the most up-to-date runtimes. For more information, see the deployment guide (<https://docs.aws.amazon.com/solutions/latest/video-on-demand/welcome.html>).

AWS Solutions Implementation overview

AWS offers a solution that ingests source videos, processes the videos for playback on a wide range of devices, and stores the transcoded media files for on-demand delivery to end users through [Amazon CloudFront](#). The diagram below presents the video-on-demand architecture you can deploy in minutes using the solution's implementation guide and accompanying AWS CloudFormation template.



Video on Demand on AWS solution architecture

This solution uses AWS Lambda to trigger AWS Step Functions for ingest, processing, and publishing workflows.

A Step Functions workflow ingests a source video, or a source video and metadata file, validates the source files, and generates metadata on the source video. A second Step Functions workflow generates an encoding profile based on the metadata and submits encoding jobs to AWS Elemental MediaConvert. After the video is encoded, a third Step Functions workflow validates the output.

AWS Elemental MediaConvert uses two-pass encoding to generate multiple high-quality versions of the original file. Source and destination media files are stored in Amazon Simple Storage Service (Amazon S3) and file metadata is stored in Amazon DynamoDB. If enabled, source files are tagged to allow the files to be moved to Amazon Glacier using an Amazon S3 lifecycle policy.

The solution also includes the option to use AWS Elemental MediaPackage as part of the workflow. When enabled, the solution creates a separate set of MediaConvert custom templates, and a packaging group in MediaPackage that is configured to ingest the MediaConvert HLS output stored in Amazon S3. MediaPackage packages the content, formatting it in response to playback requests from downstream devices. By default, this solution creates packaging configurations for HLS, DASH, MSS, and CMAF.

Full solution and deployment guide online: <https://amzn.to/AWS-ed-video-on-demand>

Real-world example

AWS Solutions: Video on Demand (VOD)

Tom from our very own AWS Solutions Builder team walks us through an end-to-end solution that he built for video on demand (VOD) on AWS. Customers are already using this solution to run over 60,000 encoding jobs every month. You'll learn how Tom used Step Functions for the orchestration layer, Lambda for Node.js microservices, Elemental Media Convert to generate videos in a variety of file formats, and many more services to complete the solution, including S3, CloudFront, CloudWatch, DynamoDB, and CloudFormation.

<https://amzn.to/AWS-ED-VOD>



Whitepaper:

Teaching Big Data Skills with Amazon EMR

The full whitepaper is available online at: <https://amzn.to/AWS-ed-big-data-emr>

About this guide

In today's competitive data analysis space, Apache Hadoop workloads have become more relevant to organizations of all sizes and purposes. As such, the skillsets associated with managing these workloads are in high demand. To keep up with this demand, universities have begun to offer classes that teach these concepts to the next generation of business and IT professionals. The IT staff supporting these organizations is often tasked with enabling students with an advanced, modern infrastructure to support a demanding curriculum. Due to short-term semesters, it is cost prohibitive for universities to invest in dedicated infrastructure for these classes, which can become quickly outdated.

This document discusses multiple deployment options available to higher-education organizations, enabling them to provide a modern AWS infrastructure to support a big data class offering using Amazon EMR. Although not limited in scope to higher education, this content was designed with the higher-education instructional use-case in mind. If you are part of an academic institution that is a member of AWS Educate, program benefits may be available. Contact your AWS Account Manager for more information on AWS Educate

Overview

Amazon EMR provides a managed Apache Hadoop service that makes it easy to deploy Hadoop open source applications quickly, such as Apache Spark and Apache Hive, enabling the processing of large amounts of data in a cost-effective way. The EMR service extracts the complexities associated with managing and scaling a Hadoop infrastructure by providing all infrastructure, configuration, and workload automation tasks for the customer. Amazon EMR helps simplify the setup of the infrastructure components such as cluster setup, auto-scaling data nodes, and permissions, making it easier to focus on teaching rather than infrastructure support.

Common EMR Applications

Amazon EMR makes it simple to provision Hadoop infrastructure, but also simplifies the deployment of popular distributed applications such as Apache Spark, Apache Pig, and

Apache Zeppelin. This document details three deployment strategies to provision EMR clusters that support these applications. For a full list of supported applications, see Amazon EMR 5.x Release Versions.

This document focuses on a few key applications that are relevant to teaching an introduction to big data with EMR.

Apache Spark

Apache Spark is a unified analytics engine used in large-scale data processing. In simple terms, Spark allows users to run SQL queries and create data frames for analysis using various common languages, mainly Java, Python, and Scala. Spark is a native component of EMR that is available to be automatically provisioned when deploying an Amazon EMR cluster.

Apache Pig

Apache Pig is an open-source Apache library that runs on top of Hadoop. It provides a scripting language used to modify large data sets with a high-level scripting language. Apache Pig enables the user to take commands that are similar to SQL (written in Pig Latin), and convert them to Tez jobs for execution in the Hadoop environment. Apache Pig works with structured and unstructured data in a variety of formats.

Apache Zeppelin with Shiro

Apache Zeppelin is an open-source, multi-language, web-based notebook that allows users to use various data processing back-ends provided by Amazon EMR. Zeppelin is flexible enough to provide functionality for data ingestion, discovery, analytics, and visualization. Zeppelin is included in Amazon EMR 5.0 and later, and provides built-in integration for Apache Spark. Configuration for how to setup Zeppelin is provided in the **Setting Up Access to Zeppelin Using Linux Credentials** section.

For multi-tenant deployments of Zeppelin, Apache Shiro is recommended as the authentication method. Shiro is a Java security framework that performs authentication, authorization, cryptography, and session management for Zeppelin notebooks.

Read the full whitepaper online at: <https://amzn.to/AWS-ed-big-data-emr>

Real-world example

Zaloni: Simplifying your Big Data Solution on AWS

Scott from Zaloni will show you how Zaloni adapted its ZDP enterprise governance and data management solution to work with AWS big data services and hybrid environments. You'll see how the company allows customers to leverage multiple services such as ephemeral EMR clusters, Athena, and Redshift, and how it integrated its product with AWS services such as S3, RDS, and Elasticsearch.

<https://amzn.to/AWS-ed-zaloni>