Intro (10 sec)

“Hi, this is Lambda Everywhere, a fully automated, serverless pipeline for scalable first-page website scraping and classification using AWS Lambda, ECS, Glue, SQS, and S3. In this walkthrough, I’ll show how we can scrape websites at scale and classify them under labels like News and Education, while Lambda and ECS containers work together to orchestrate, execute, and automate the entire workflow without manual intervention.

Slide 2: Solution Architecture (20 sec)

[Point to architecture diagram]

“Here’s the high-level architecture:

A .txt file of websites is uploaded to S3, which activates a Lambda function. This file could potentially include millions of website names, but in our demo, it contains just 100 due to sandbox environment limits and demo time constraints.

Lambda triggers Glue to chunk the list.

Glue uploads the chunks to S3 and sends messages to SQS. Each message contains the S3 path to a chunk file containing a list of websites to scrape.

These messages trigger a Lambda function that performs the actual scraping and stores the results in an S3 folder. S3 then sends messages to another SQS queue, with each message consumed by one ECS task.

Since Lambda scales horizontally, we could potentially invoke up to 10,000 Lambda functions in parallel to scrape different chunks efficiently.

The Glue job also triggers a Lambda to start ECS tasks, so ECS service creation and chunking happen in parallel, with the number of tasks matching the number of chunks.

ECS tasks classify the scraped data and upload labeled outputs to another S3 folder.

A monitoring Lambda automatically stops ECS when classification completes.”

Live Tabs Walkthrough (2 min)

Open these tabs in order, with each <10-15 sec explanation:

✅ Tab 1: S3 Bucket

“Here’s the S3 bucket where users upload the .txt file with URLs.”

✅ Tab 2: Glue Jobs

“Glue chunks the file, sends SQS messages, and triggers Lambda to create the ECS service.”

✅ Tab 3,4,5: SQS Queue

Here’s the SQS queue website-chunks-queue, which receives chunk messages from Glue and triggers the Lambda scraper function to start scraping; when the results are saved in the output\_sqs folder, S3 automatically sends a message to another queue called EcsTaskQueue.  
  
Now we can see there are 10 messages waiting for ECS tasks to start and perform the classification.

✅ Tab 4: Lambda Console

“These Lambda functions orchestrate scraping, manage ECS lifecycle, and monitor pipeline completion.”

✅ Tab 6: Scraper Lambda Logs

“Lambda functions scrape websites using requests and BeautifulSoup, respecting robots.txt. In this sandbox, we use lightweight scraping instead of Selenium for JS-heavy pages.”

✅ Tab 7: S3 Scraped Data

“Scraped data is stored in S3 for classification.”

✅ Tab 8: ECS Worker Logs

“ECS tasks continuously poll SQS, classify the scraped content using Hugging Face BART MNLI, and upload labeled results.”

✅ Tab 9: Final Classified Outputs in S3

“Once all chunks are classified, a Lambda function automatically deletes the ECS service, completing the pipeline.”

Conclusion (20 sec)

“Lambda Everywhere demonstrates how we can use AWS Lambda across the pipeline to orchestrate, scrape, and classify millions of websites efficiently without manual server management. This architecture is scalable, event-driven, and cost-effective, unlocking powerful web-scale data workflows using serverless tools.”

“Thank you!”