

## Case Study: **MediaVision**

### **AWS AI Enhances product uploads**

This case study details the development of an automated content creation solution for MediaVision, a marketing technology company focused on enhancing e-commerce experiences. Leveraging the AWS cloud, MediaVision is building an AI-powered pipeline to automatically ingest and process images of clothing items, generating high-quality, SEO-optimised titles and descriptions for their customers websites.

By utilising AWS's scalable compute and storage solutions, MediaVision has significantly accelerated the content creation process while maintaining flexibility and cost efficiency. The automation enabled by AWS not only reduces the manual effort and time associated with content generation but also allows MediaVision to consistently produce accurate, compelling product descriptions at scale.

#### Partnership

This was the customer's first engagement with transACT and their first experience working with any cloud provider. Although the customer had a team of highly skilled developers and data scientists, they lacked AWS expertise. transACT ensured that this lack of cloud experience did not impede their business objectives. We provided full support throughout the project lifecycle, assisting with design, architecture, deployment, and ongoing support, as well as addressing any ad hoc requests that arose during the project.



## Business Challenges

Before implementing their AWS solution, MediaVision faced significant challenges in generating high-quality product titles and descriptions for their website. Initially relying on OpenAI's models, they found that the responses to their prompts were inconsistent in quality, often requiring extensive manual editing to meet their standards. This inefficiency not only slowed down the content creation process but also impacted the uniformity and accuracy of descriptions across their catalogue. Scalability was another concern, as producing descriptions at scale with these limitations proved time-consuming and resource-intensive.

By transitioning to AWS, MediaVision aimed to create a more reliable and scalable solution, ensuring consistently high-quality outputs that would reduce manual intervention and streamline their e-commerce operations.

The solution involved the following AWS services:

### 1. Amazon S3

Store raw images.

### 2. Amazon Bedrock

Run the Claude Sonnet 3.5 Foundation model to generate product titles and descriptions.

### 3. Amazon API Gateway

Act as ingestion point from AWS, integrating with customer internal sites and workflows.

### 4. Amazon Rekognition

Tag & label images to be sent to Bedrock to enhance output accuracy

### 5. Amazon DynamoDB

Store outputs from Amazon Bedrock on a temporary basis

### 6. AWS Lambda

Orchestration of the system

### 7. Amazon CloudWatch / SNS / IAM

CloudWatch for monitoring, SNS for notifications, and IAM for secure authentication and access control

## Implementation:

### Data Collection and Storage

#### Data Sources:

- Product images uploaded from clothing retailers are uploaded to their own internal company portals where a customised integration adds the images to Amazon S3 and sends a request to Amazon API Gateway to kick off the processing architecture for the request.

#### Event Automation:

- Amazon Lambda is used to perform data validation before kicking off the processing architecture, mainly ensuring that all images uploaded are the correct size & quality that Amazon Bedrock can accept and process. If images are not appropriately sized, a response is sent back, alerting users that they must resize images before upload.

### Data Processing:

- Amazon Rekognition is triggered (DetectObjects Job) with the uploaded images to provide a list of tags with items associated within the image in order to assist the Bedrock Foundation Model (Amazon Titan) with the description generation.
- Once Rekognition has provided an output, the images, tags & a predefined prompt are sent to Amazon Bedrock in order to generate both a title and description for the items.

### Outputs:

- All generated data is stored in a DynamoDB with the product code as they key & the Bedrock outputs as the value in order to keep a safe record of previous generations, with a TTL setting configured in order to keep only recent description generations.

### Choosing a Foundation Model:

- MediaVision were keen to reduce their operating and cloud costs as a part of this project and were open to exploring various Foundation Models.
- This was done by selecting a plethora of FMs and beginning a rigorous testing period judging models using Amazon Bedrock's Model Evaluation feature on 4 key metrics: Coherence, Consistency, Completeness & Correctness with a thumbs up rating method using a customer managed work team.

- After comparing three core models from three different providers (Anthropic, Meta & Amazon) & adding In additional comparison features such as cost, the Amazon Titan Text G1 – Premier model was selected for this workload.

### Monitoring and Security

#### Monitoring:

- Amazon CloudWatch: Monitored AWS resource performance, set up alarms, and automated responses to specific conditions.
- AWS X-Ray: Traced and analysed requests across services to identify bottlenecks and performance issues.

#### Security:

- **AWS IAM:** Managed secure access to AWS resources by setting up roles and policies to ensure that only authorised personnel and services could access sensitive data and critical systems.
- Access to S3 was restricted to authorised users only.

#### Compliance and Data Governance:

- AWS Config: Used to monitor and audit AWS resource configurations, ensuring compliance with industry standards.

## Products and Services

Bedrock, Rekognition, S3, DynamoDB, Lambda, API Gateway, IAM.

## Outcomes

- ▶ **15% Reduction in Operating costs:**  
Achieved a significant decrease in the cost associated with the Generative AI Architecture vs 3rd party alternatives.
- ▶ **Increased SEO results:**  
Enhanced the SEO optimisation of product details when generated through AI.
- ▶ **Increased throughput:**  
Increased throughput of images uploaded due to the streamlined and scalable architecture build in the AWS Cloud.

