Processing Bulk Sales Orders with Amazon Web Services

Gilmore Building Supplies, a building supplier in the United Kingdom, was confronted with a challenge: their clients would purchase thousands of items in advance yet needed to stagger deliveries across several months. These deliveries required careful tracking and coordination of items to ensure proper delivery without redundant or lost inventory. Although they were able to successfully handle the task, it required a meticulous, time-consuming manual approach.

Gilmore sought a new approach that would save them time while also alerting them of potential discrepancies. They had already identified AWS Textract as a valuable service role in that prospective process. Yet AWS Textract alone would be insufficient for their needs: they needed to further enhance and transform the results to help them track what was ordered versus what was delivered.

James Gilruth, Operations Manager at Gilmore Building Supplies, recalls the early discussions with MTech, "I wanted to automate my sales order processing, and after a couple of hours talking to them, they fully understood the process, from why it's done to how it's done, which other vendors weren't interested in.”

After a review of the initial specifications, MTech prepared a project plan involving multiple milestones that would incrementally deliver ever-increasing functionality. The incremental approach would permit James Gilruth and his team to evaluate the progress of the new system, adjust specifications as necessary, and evaluate whether to proceed to the next milestone without risking payment for the full system up-front.

The New Approach

User Experience

The new approach would aim to provide a much easier and faster user experience:

1. the staff would upload the scheduled order and actual order into a new web portal that was securely accessible by the Gilmore’s staff
2. the web portal would provide the staff with a consolidated spreadsheet containing:
    1. a detailed comparison of the scheduled versus actual order, and
    2. cautions and advisories based on various business rules.

That simple user experience would mask an intricate process taking place within Gilmore's AWS infrastructure.

Integration of AWS Services

The web portal would be delivered through AWS CloudFront to provide a low-latency response for the staff via its rapid content delivery network. The first line of defense made use of CloudFront's geolocation filtering to limit access from only selected location where the staff were permitted to connect from. The next line of defense utilized user authentication.

The web portal would allow staff to upload the scheduled and actual orders up to a secure and ephemeral AWS S3 bucket coordinated with a serverless AWS Lambda function.

The uploading of the documents would trigger an AWS CloudWatch event which would automatically trigger the main automation process handled by an AWS Step Functions state machine as illustrated below.

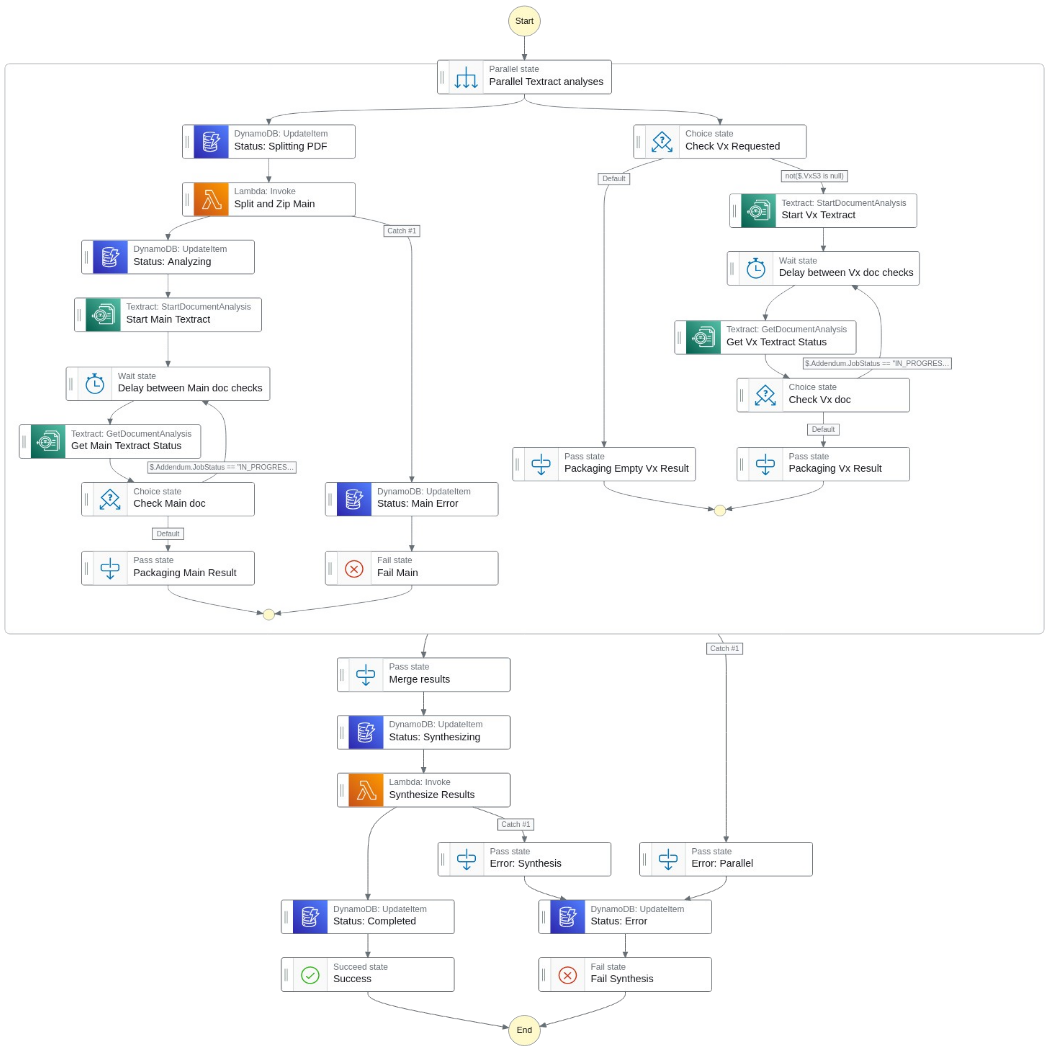

Figure 1: Automation diagram of the main order processing

Get it done right
Get MTech

That state machine would begin an entire process for the coordination of:

- posting status updates of the state machine’s process to AWS DynamoDB
    - those status updates would be relayed via a separate AWS Lambda function to provide a responsive user experience in the web portal
- triggering and monitoring the extraction of textual content from the scheduled and actual order documents with AWS Textract
- triggering the synthesis of the results subject to:
    - extensive enhancements of the text constrained by business-specific interpretations,
    - transformation of the data subject to various business rules,
    - packaging of the output as a spreadsheet to integrate with Gilmore’s subsequent business processes.

The entire process would occur securely within Gilmore’s own AWS infrastructure. As the various components were configured to be serverless and consumed minimal storage, they would incur practically zero cost while not in use.

Deployment of the AWS Services

Testing the various versions of the the system without affecting the current version in production required a method of provisioning new and updated infrastructure reliably and quickly. Towards that end, the entire system was defined in AWS CDK which is a framework for defining cloud infrastructure in code and provisioning it through AWS CloudFormation. It would provide the ability to audit the evolution of the infrastructure over time.

Outcome

After review and approval, the new automation was developed and deployed into test and production. The benefits were immediate and notable:

- easy usage,
- helpful advisories,
- fast results.

Across the thousands of items orders since deployment, the automation system has proven to be reliable and cost-effective with near zero-costs while idle. As James Gilruth notes, "All together, it's saved us hundreds of hours a year."

If you’d like help with your next project, reach out to discuss.

Get it done right
Get MTech