Diversion Diagnostics Report Automation

Law firm

Automated the process of generating a diversion diagnostic report that helped identify sellers who may be potentially diverting goods to unauthorized channels/sellers

Diversion diagnostics report automation for a law firm

DIVERSION DIAGNOSTICS REPORT AUTOMATION

Situation

- Attorneys at the client use Diversion Diagnostics reports to identify authorized sellers/distributors who may be diverting the goods to unauthorized channels. However, it took more than 2 days for the client's data analyst to create reports every time using R-based scripts, involving multiple manual steps and iterations.
- Partnered with the client to develop an end-to-end automated process to request, generate and deliver the reports without the need of manual intervention

Accordion Value Add

- Converted existing R code of the report into Python (Azure Databricks) to enhance and optimize the same for improved memory and reduced runtime using parallel processing with robust logic
- Automated the entire workflow including report request using PowerApps, Azure Data factory for orchestration, Databricks based code having SQL queries for fetching data, Pyspark/Python code for data transformation, Excel report generation and Logic Apps based emails to deliver the reports published on SharePoint
- Created a PowerApps based UI that provided real time information of available data that allowed flexibility to choose various control parameters to customize the report based on the context
- Standardized and templatized the final reports to a consistent and user-friendly format for easier consumption

Impact

- Code improvement reduced code runtime per report from over 8-12 hours to 1 hour (~90% reduction in runtime)
- The deployed solution reduced the TAT (request to delivery) for diversion diagnostics report to less than 2 hours depending on size of specific dataset from 2 days and eliminated need of a data analyst completely
- Upfront information on available data, with the ability to select custom parameters reduced number of iterations and back-forth communications for each report

Methodology/ approach

OPTIMIZE

- Translated the existing **R code to Python** and optimized the same for **faster run-time**
- **Improved logical inconsistencies** in the code which include error handling and corner case handling
- Modularization and parallel processing **improved performance and consistency**

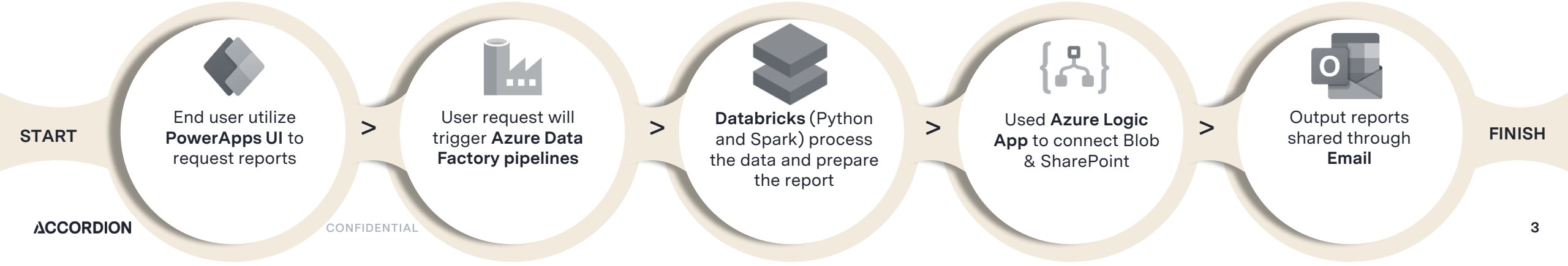
STANDARDIZE

- Understood the business use case and **created standard templates** for the reports
- Standardized the user parameters and choices to enable consistent calculation and analysis
- Improved the UI of the report for **increased usability and clarity**

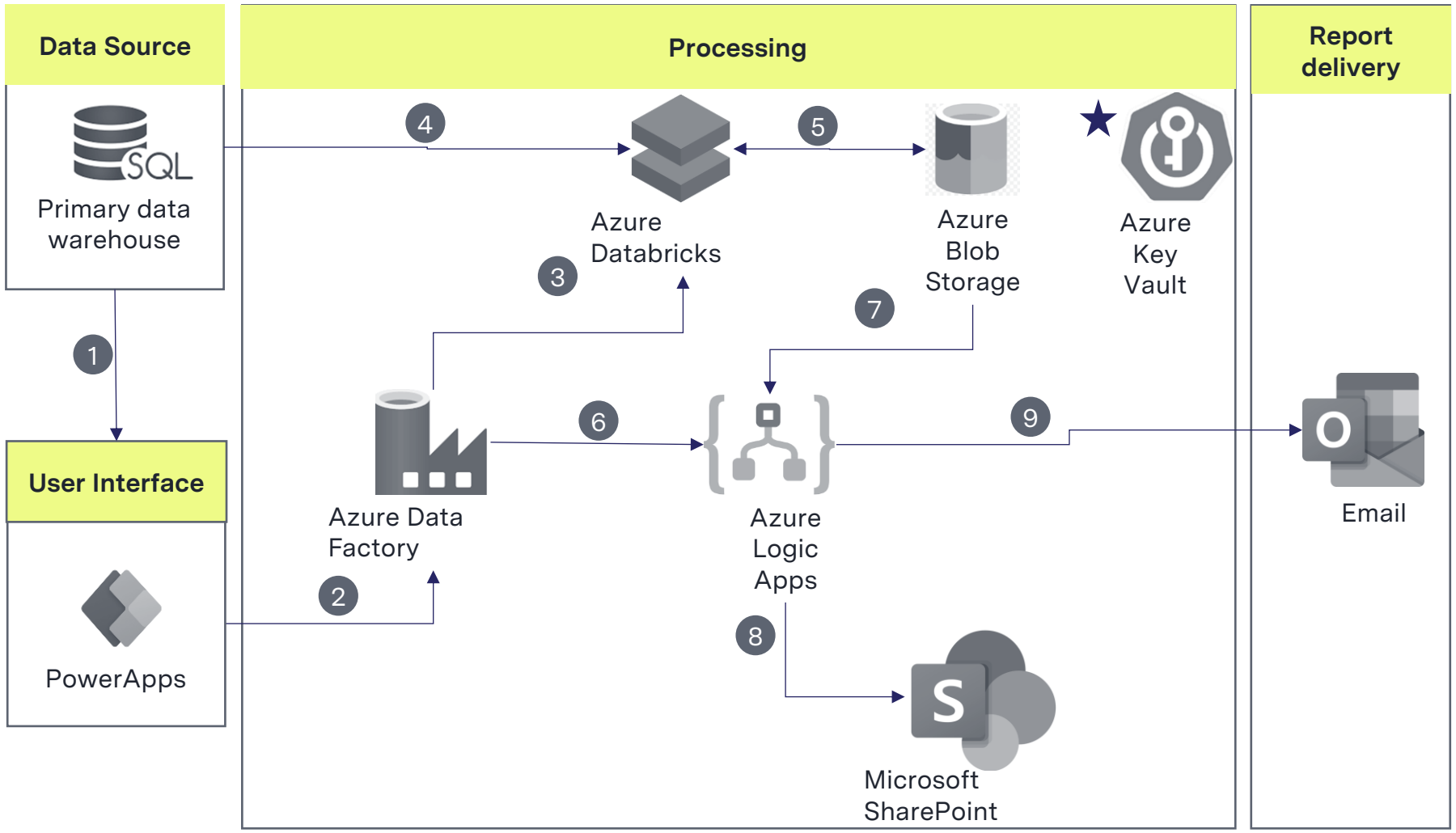
AUTOMATE

- **Created** a complete system including a **user interface for taking report request**, a pipeline to fetch data from database and then run the analysis
- Automatically **pulled user email from PowerApps login**, passed it along with other parameters to data pipeline
- Deployed Python and SQL queries in Azure Databricks taking advantage of auto-scalability and parallel processing optimizing performance and cost
- Used **Logic Apps to share final reports** with user over SharePoint link to avoid file size limitations on direct mails
- **Implemented security measures** like email domain check to ensure reports are not sent to unauthorized email domains

Process Flow



Application architecture



1. When user access the PowerApps, it pulls information about available data from database
2. When user sends request, Azure data factory is triggered using user inputs and parameters
3. Data factory runs notebooks in data bricks for analysis
4. Databricks notebooks fetch data from SQL database and runs analysis
5. Databricks fetches report templates from blob storage and inserts analysis outputs and saves to Blob storage
6. ADF triggers Logic app and passes report meta data
7. Logic app fetches the created reports from blob storage and pushes it to a
8. SharePoint folder.
9. Logic app creates a shareable link to the report and sends a mail to user with the link to the SharePoint folder.

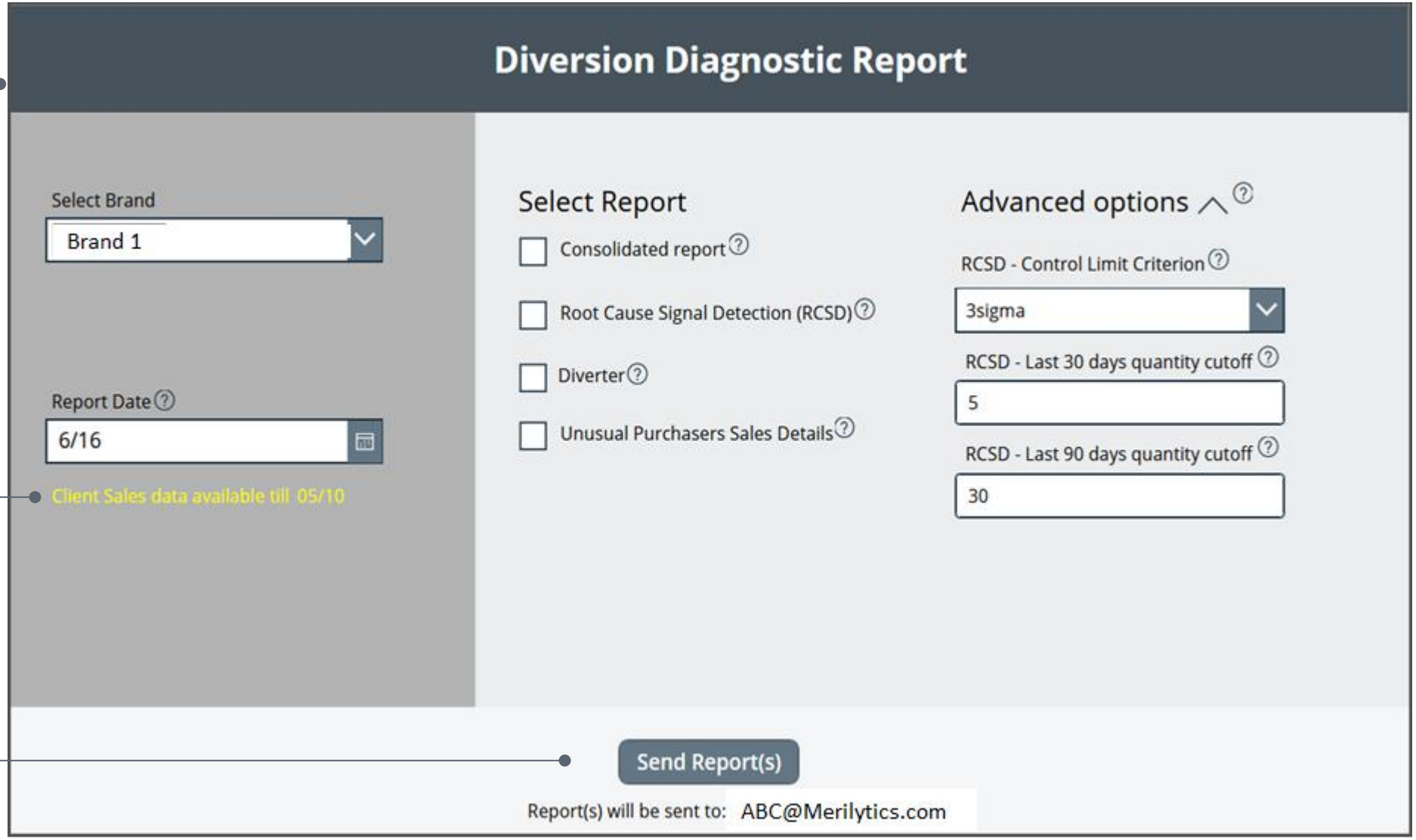
★ Azure Key vault used for storing all access tokens and keys for security

Application user interface

Application user interface leveraged by end user to request a report

Information displayed on UI are extracted from Azure SQL database

A request from end users will trigger the process set up in Azure resources



Output reports and emails

Output email received by the user

Action Items

CAUTION: External Email

Hi,
User directions to download report from SharePoint to local:
1. Copy and paste the URL of the report in a new tab on browser
2. Click on 'File' on top-right corner of the screen
3. Click on 'Save as'
4. Click on 'Download a copy'
5. Download the report to the intended folder.
6. Post downloading the report user must click on 'Enable Editing' button to allow the excel formulae to run

Please find the links to the requested report(s) for Brand-1 below.

Consolidated Report
https://merilytics.sharepoint.com/Brand-1/consolidated_report.xlsx

Root Cause Signal Detection (RCSD)
https://merilytics.sharepoint.com/Brand-1/root_cause_signal_detection.xlsx

Amazon Diverter
https://merilytics.sharepoint.com/Brand-1/amazon_diverter.xlsx

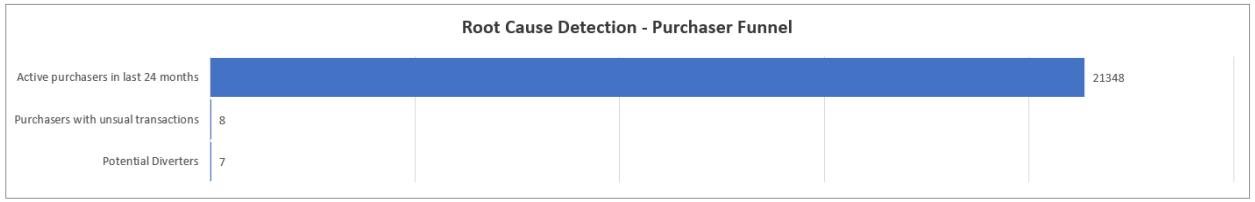
Unusual Purchasers Sales Details
https://merilytics.sharepoint.com/Brand-1/unusual_purchasers_sales_details.xlsx

File Home Insert Page Layout Formulas Data Review View Help

E37

User Inputs from Request Form:

Control limit	3σ	L1M Quantity Cutoff	5	L4M Quantity Cutoff	30



Purchasers	Count (#)
Active purchasers in last 24 months	21348
Purchasers with unusual transactions	8
Potential Diverters	7

Note >> *The below table captures all potential diverters grouped on the basis of Risk category and sorted on the basis of Quantity purchased in L4M*

Potential Diverters

Sl No	Purchaser	Risk Category	Sum of Quantity in L4M	Risk Basis	Signalled before	In the matching report	Diverter (User findings)
1	Purchaser 1	Very High Risk	20256	Count over 3-sigma	Yes	N/A	
2	Purchaser 2	High Risk	4598	Count over 3-sigma	Yes	N/A	
3	Purchaser 3	Moderate Risk	1764	Count over 3-sigma	No	N/A	
4	Purchaser 4	Moderate Risk	741	Both	No	N/A	
5	Purchaser 5	Moderate Risk	252	Both	Yes	N/A	
6	Purchaser 6	Low Risk	134	Count over 3-sigma	Yes	N/A	
7	Purchaser 7	Low Risk	102	Both	No	N/A	

Glossary | Outputs>> | Risk Profile Summary | Purchaser Data | Raw Data>> | Amazon Diverter Report Data | Count vs ...

Output report generated through automated process