

Company Type Summary Dashboard (SQL-Based Analysis)

This dashboard summarizes user activation data across real campaigns. The data was extracted and aggregated using advanced SQL queries and visualized in Power BI. The main goal was to analyze user activation levels by company size and automation status.

Key Focus Areas:

- SQL joins, filters, and custom business logic-
- Company type classification based on total user population
- User activation rate and automation segmentation

Summary by Population

1.9M

Sum of ees

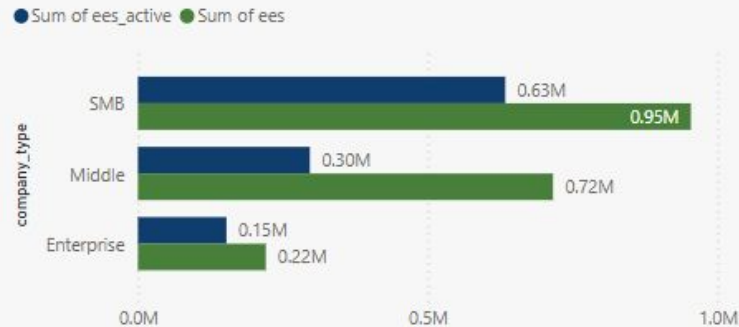
1.1M

Sum of ees_active

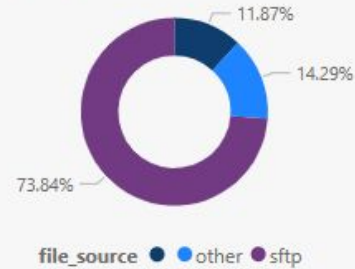
59 %

Average Activation Rate

Population divided by types

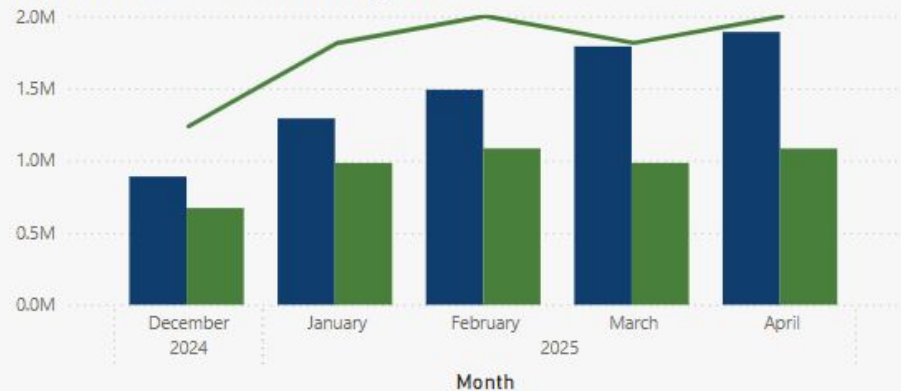


Amount of import source



automation_status	Count of automation_status
	3
SFTP Fully Automated	31
Pending	53
Non-SFTP Fully Automated	79
Non-SFTP Standard	208
Fully Automated	665
Semi-Automated	815
Total	1854

Population and Activation level by Months



SQL Query (Code Section)

SELECT

```
campaign.alias AS alias, campaign.name AS title, campaign.data_source AS import_file_source,
SUM(CASE WHEN participants.organization_id IS NOT NULL THEN 1 ELSE 0 END) AS ees,
SUM(CASE
  WHEN users.employer_id IS NOT NULL AND users.active = 'true' THEN 1 ELSE 0 END) AS ees_active,
campaign.automation_state AS automation_status,
CASE
  WHEN SUM(CASE WHEN participants.organization_id IS NOT NULL THEN 1 ELSE 0 END) < 500 THEN 'SMB'
  WHEN SUM(CASE WHEN participants.organization_id IS NOT NULL THEN 1 ELSE 0 END) BETWEEN 500 AND
2999 THEN 'Middle'
  ELSE 'Enterprise'
END AS company_type
FROM participants
JOIN participant_profiles ON participant_profiles.participant_id = participants.id
JOIN profiles ON participant_profiles.profile_id = profiles.id
JOIN campaign ON participants.campaign_id = campaign.id
WHERE participant_profiles.relation_type = 'primary'
  AND participants.parent_id IS NULL
  AND participants.is_blocked = FALSE
  AND campaign.is_production = TRUE
  AND campaign.is_blocked = FALSE
  AND campaign.is_live = TRUE AND ( campaign.alias NOT LIKE ALL (...) AND
(campaign.name NOT LIKE ALL (...) OR campaign.name = 'Generic Campaign A') )
  OR campaign.alias IN ('campaign_special_1', 'campaign_special_2', 'campaign_special_3')
GROUP BY campaign.alias, campaign.name, campaign.data_source, campaign.automation_state
ORDER BY enrolled_participants;
```

SQL Logic Summary

Overview of Logic

- **Join Operations:** Data from participants, their profiles, and associated campaigns are joined using relational keys.
- **Filtering:**
 - Only participants marked as active (not blocked or archived) are included.
 - Filters out campaigns with names/aliases commonly used for testing, demos, or placeholder data.
- **Aggregation:**
 - The number of participants linked to an organization is counted using a **SUM(CASE WHEN ...)** pattern.
- **Grouping:**
 - Results are grouped by campaign metadata: alias, name, data source, and automation status.
- **Ordering:**
 - Campaigns are sorted by the number of organizational participants to highlight the largest or most engaged ones.