Improving Reclaim Wafer Traceability for Vendor Performance Optimization

Department of Industrial Systems Engineering and Management (ISEM)
IE3100R Systems Design Project AY 2025/26

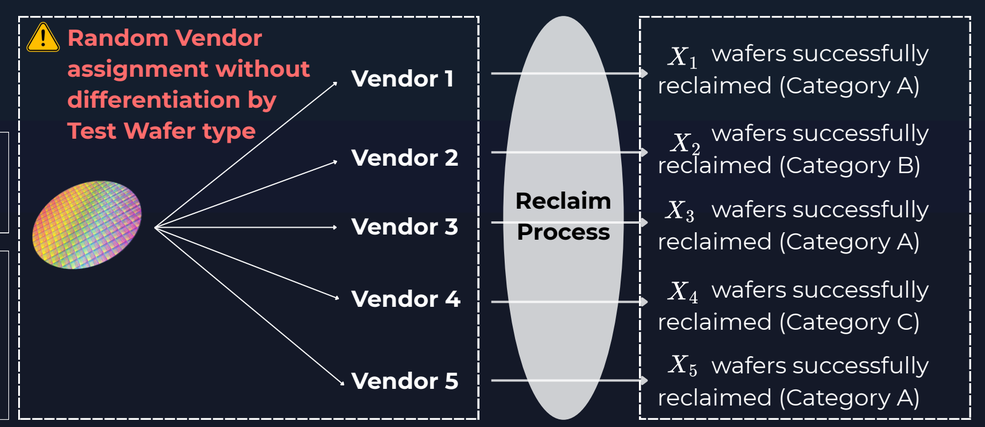
Team Members (Group 2): Ratik Bhatnagar, Kaushik Raman, Wang Xudong, Jerrick Goh Shang Yun, Mukund Kaverirangan, Wu Yixuan
Micron Supervisor: Ms. Farah Ramjoo Baug
NUS Supervisor: Prof. Cheung Wang Chi

Project Background

Test Wafer(TW)/Reclaim Wafer: Wafer used to test semiconductor manufacturing processes.
Reclaim Process: Restoring used Test Wafers by removing films and polishing the surface so wafers can be reused for further testing.

Reclaim Test Wafers are currently classified into three main categories within Micron:

Category A Non-Copper Non-Pattern	Category B Non-Copper Pattern	Category C Copper
$TestWafer_A$ $TestWafer_B$ $TestWafer_C$ . . .	$TestWafer_D$ $TestWafer_E$ $TestWafer_F$ . . .	$TestWafer_G$ $TestWafer_H$ $TestWafer_I$ . . .



Project Objectives

Problem Description: The lack of an automated Test Wafer tracking system prevents analysis of reclaim yield by Vendor or Test Wafer type, resulting in random wafer allocation and limiting reclaim yield optimization. This also prevents wafer tracking from Micron to Vendors.

Objective I (Data Analytics): To implement automated data driven decisions to optimize Vendor allocation, thereby increasing Test Wafer reclaim yield.

Objective II (Operational Procedures): To roll out a comprehensive Test Wafer handling SOP, thereby enhancing wafer traceability from Micron to Vendors.

Methodology

01	02	03	04	05
Problem Identification	**System Design**	**Solution Implementation**	**Solution Improvement**	**Monitoring & Control**
• Process Observation • Stakeholder Consultation • Root Cause Analysis	• System Architecture • Operational Process Design • Planning	• Data Integration • Dashboard Development • Operational Process Testing	• Edge Case Analysis • Stakeholder Feedback Integration • Iterations	• System Performance Evaluation • Dashboard Debugging • Handover

Key Formulas

1. $$\text{Reclaim Yield[TW A]} = \frac{\text{Pass Count[TW A]}}{\text{Total Count[TW A]}}$$

2. $$\text{Total Count [TW A]} = \text{Pass Count[TW A]} + \text{Fail Count[TW A]} + \text{Hidden Fail[TW A]}$$

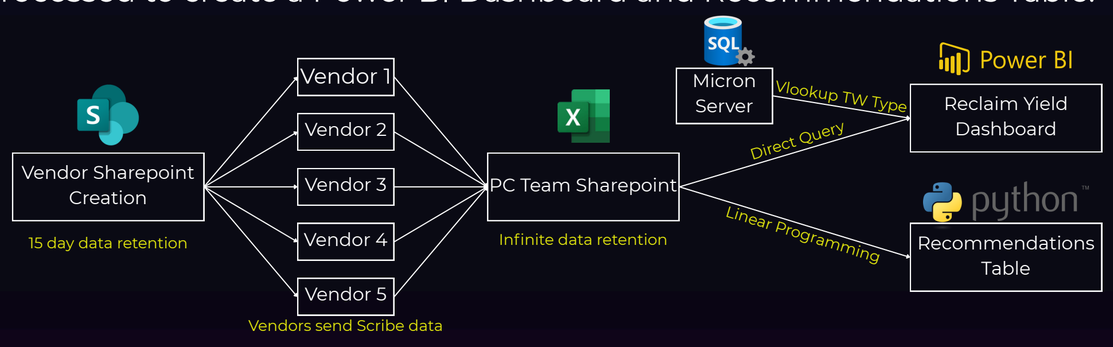
3. $$\text{Hidden Fail[TW A]} = \text{Weightage[TW A]} \times (\text{Total Count[All TWs]} - (\text{Total Pass[All TWs]} + \text{Total Fail[All TWs]}))$$

4. $$\text{Weightage[TW A]} = \frac{\text{Pass Count[TW A]} + \text{Fail Count[TW A]}}{\text{Pass Count[All TWs]} + \text{Fail Count[All TWs]}}$$

Objective I (Data Analytics)

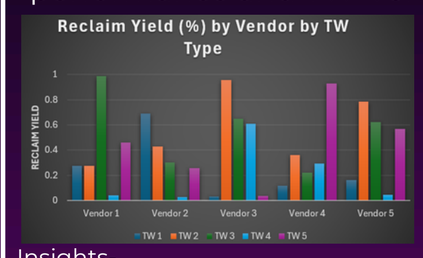
Data Collection Pipeline

Stakeholder collaboration with 5 Vendors enabled a data collection cadence. Data integrity was maintained through data validation techniques and was processed to create a Power BI Dashboard and Recommendations Table.

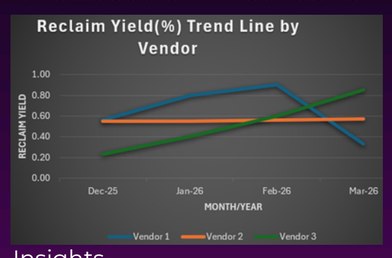


Power BI Dashboard

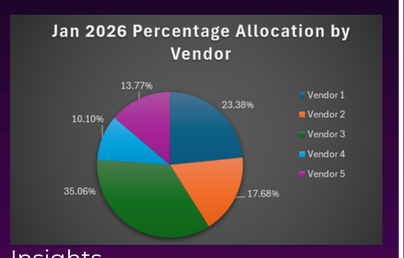
Monthly auto refresh to improve visibility of real time Reclaim Yield, Vendor performance and TW-Vendor allocation metrics for the Micron PC Team.



Insights-
- Vendor 1 best reclaims TW3
- TW2's ideal Vendor is Vendor 3



Insights-
- Vendor 1 saw a drop in Yield
- Vendor 3 has rising Yield



Insights-
- Vendor 3 recieved the largest proportion of TWs

Vendor Recommendations Table

Assigns TW Types in a 1:1 mapping to the optimal Vendor with the highest Reclaim Yield using Linear Programming.

$$\text{Max } Z = \sum_t \sum_v \left(\text{ReclaimYield}_{t,v} \times \text{MonthlyQuantity}_t \times \text{Assign}_{t,v}\right)$$

Variables:
$\text{ReclaimYield}_{t,v}$ = Vendor v′s Reclaim Yield for TW Type t
MonthlyQuantity_t = Number of TW t wafers sent per month
$\text{Assign}_{t,v}$ is the decision variable where 1-assign, 0-otherwise

Constraints:
1. 1 TW Type can be assigned to 1 Vendor
2. Vendor capacity should not exceed
3. Generic Vendor ≥ 50% × all TWs

Objective II (Operational Procedures)

End-to-End Test Wafer Dispatch Process

- Test Wafer identified for external reclaim
- Auto-Tagged in internal system with "***Wafer Category_ Assigned Vendor***"
- Containers pre-assigned in system by Wafer Category_Vendor
- Wafers with matching attributes grouped together
- Containers labeled with color-coded stickers and tags before warehouse dispatch
- Containers loaded into dedicated pre-labeled sharkcages
- Wafers retrieved from shark cages for designated Vendor pickup

Key Operational Improvements & Benefits

1. New System Attribute for Two-Level Wafer Segmentation

Each Test Wafer type is assigned a Vendor based on the Vendor Recommendations table. The addition of a Vendor attribute alongside Wafer Category enables finer wafer segmentation, improving traceability and enabling performance-based Vendor allocation instead of random assignment.

2. Enhanced Wafer Container Labeling (System & Physical)

Dedicated containers are assigned by Wafer Category and Vendor, replacing random allocation. This improves wafer-to-Vendor traceability and enables accurate Vendor dispatch. Automated system pre-tagging reduces manual work, while pre-pasted labels allow easy identification of wafer contents during inter-facility movement.

3. Dedicated Shark Cage Allocation

Shark cages are assigned by Wafer Category_Vendor, with no mixing permitted. This aligns operational handling with system-level sorting, reducing confusion and errors. Clear labeling enables easier identification and movement.

Key Outcomes

- Live data improves wafer tracking and decision-making
- Data-driven recommendations enabled more optimal Vendor allocation
- Optimized wafer–Vendor allocation boosts reclaim yield across TW types
- Reduced manual workload and improved process consistency through refined SOP

Future Direction

- Expand project scope to include Copper wafers *
- Scale the model to onboard new Vendors
- Negotiate with Vendors to minimize the number of Hidden Fail Wafers to further improve Reclaim Yield accuracy
- Enhance data collection pipeline by pivoting from Excel SharePoint to a cloud based storage (Azure Databricks)

**** Copper wafers are currently excluded from this project due to resource constraints.***