TestConX™

Archive

DoubleTree by Hilton
Mesa, Arizona
March 3-6, 2024

Test Time and Cost Reduction using Intelligent Prediction from ML Models

Lisa Taubensee, Yiwen Liao, Matthias Sauer, Sarah Rottacker
Advantest

TestConX 25

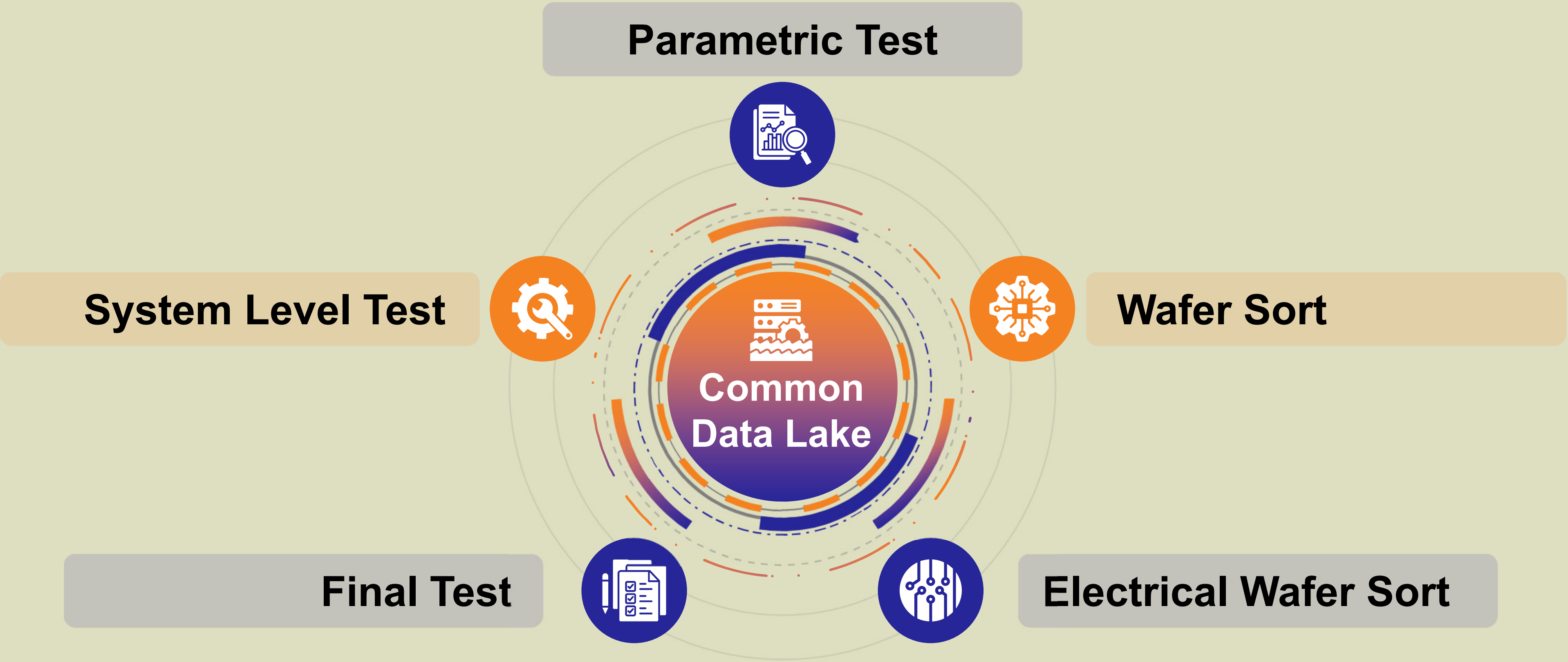
Mesa, Arizona ● March 3–6, 2024

ADVANTEST®

Agenda

- Introduction
- Background
- Method
- Demo
- Results
- Conclusion

Data Synergy: Integrating Insights for Competitive Edge



Parametric Test

System Level Test

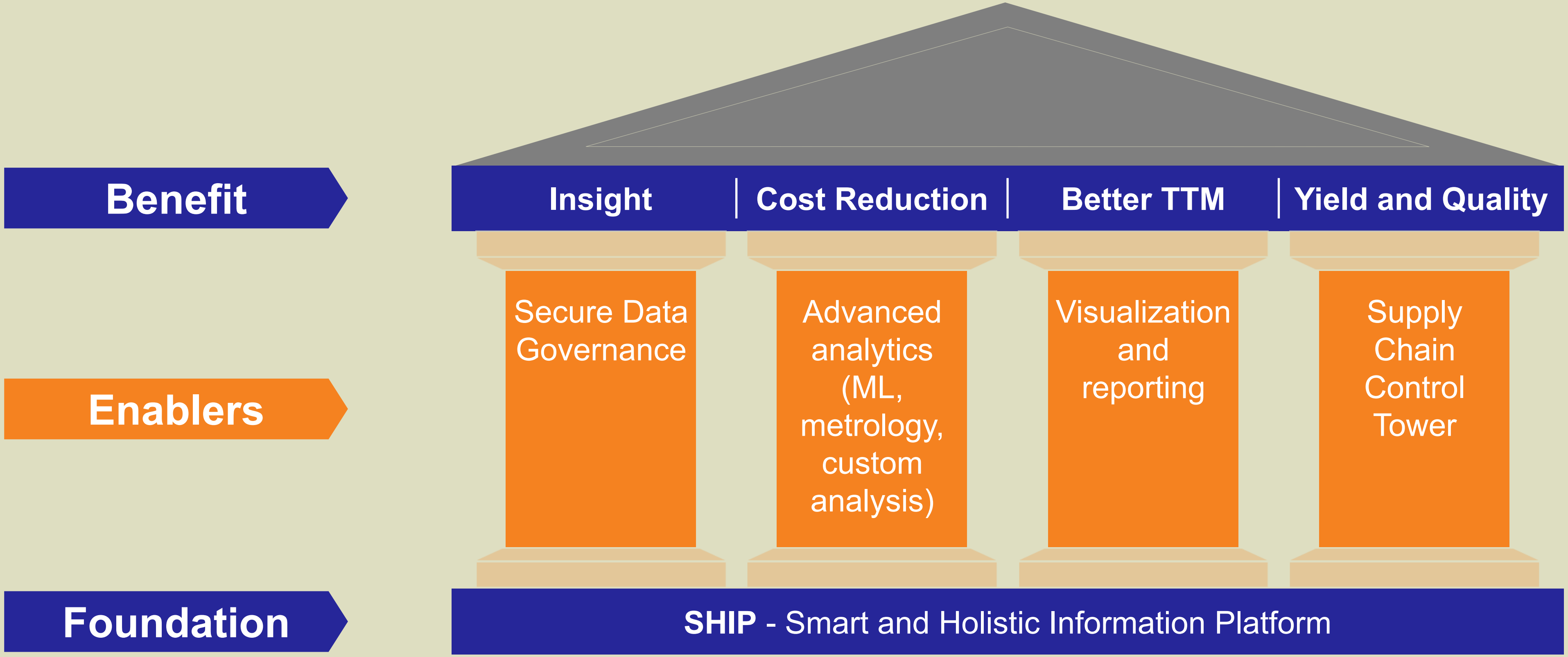
Wafer Sort

Common Data Lake

Final Test

Electrical Wafer Sort

Transformative Workflow for Enhanced Performance



Benefit

Insight | **Cost Reduction** | **Better TTM** | **Yield and Quality**

Enablers

Secure Data Governance

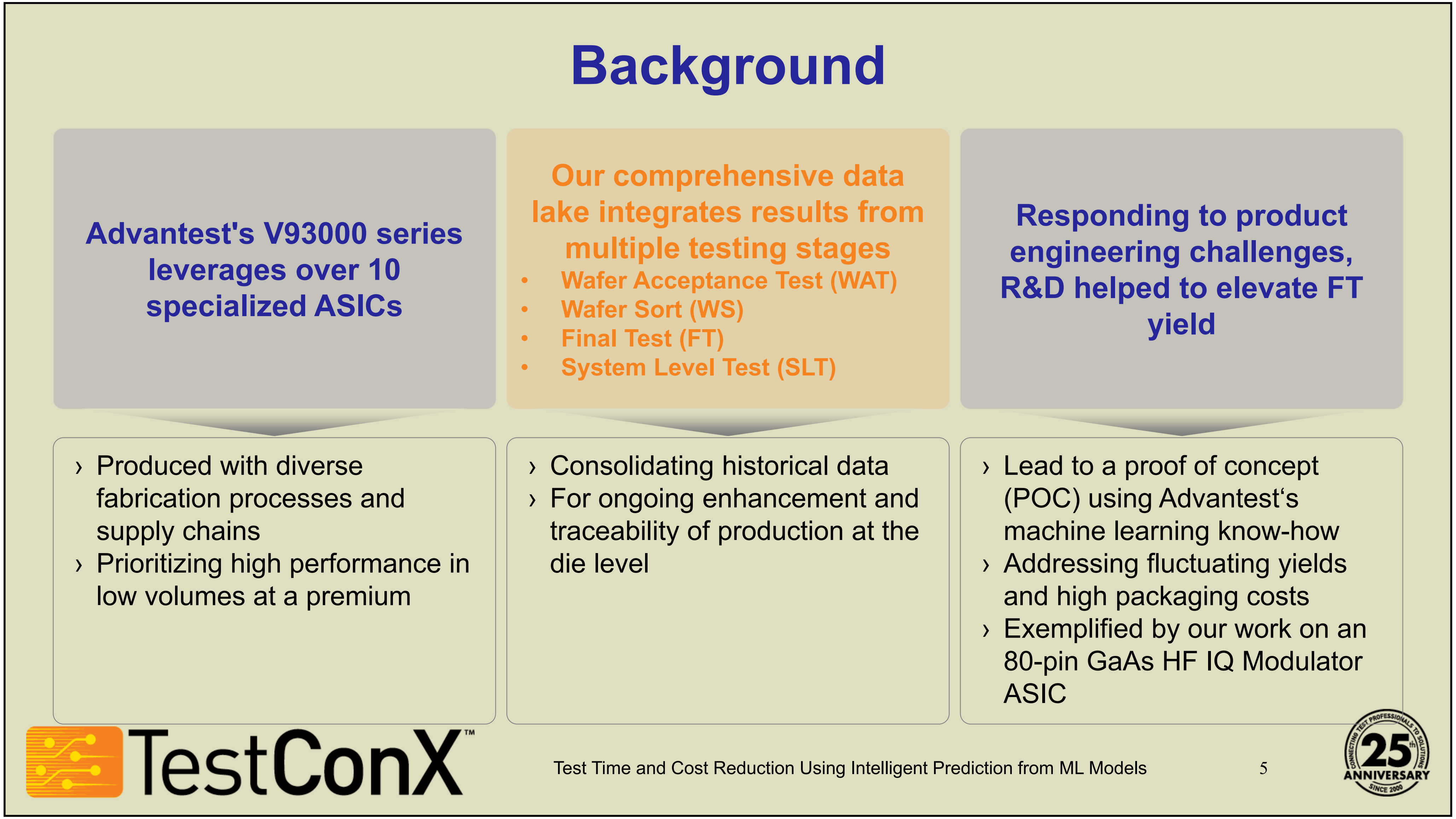
Advanced analytics (ML, metrology, custom analysis)

Visualization and reporting

Supply Chain Control Tower

Foundation

SHIP - Smart and Holistic Information Platform



Background

Advantest's V93000 series leverages over 10 specialized ASICs

- Produced with diverse fabrication processes and supply chains
- Prioritizing high performance in low volumes at a premium

Our comprehensive data lake integrates results from multiple testing stages

- Wafer Acceptance Test (WAT)
- Wafer Sort (WS)
- Final Test (FT)
- System Level Test (SLT)

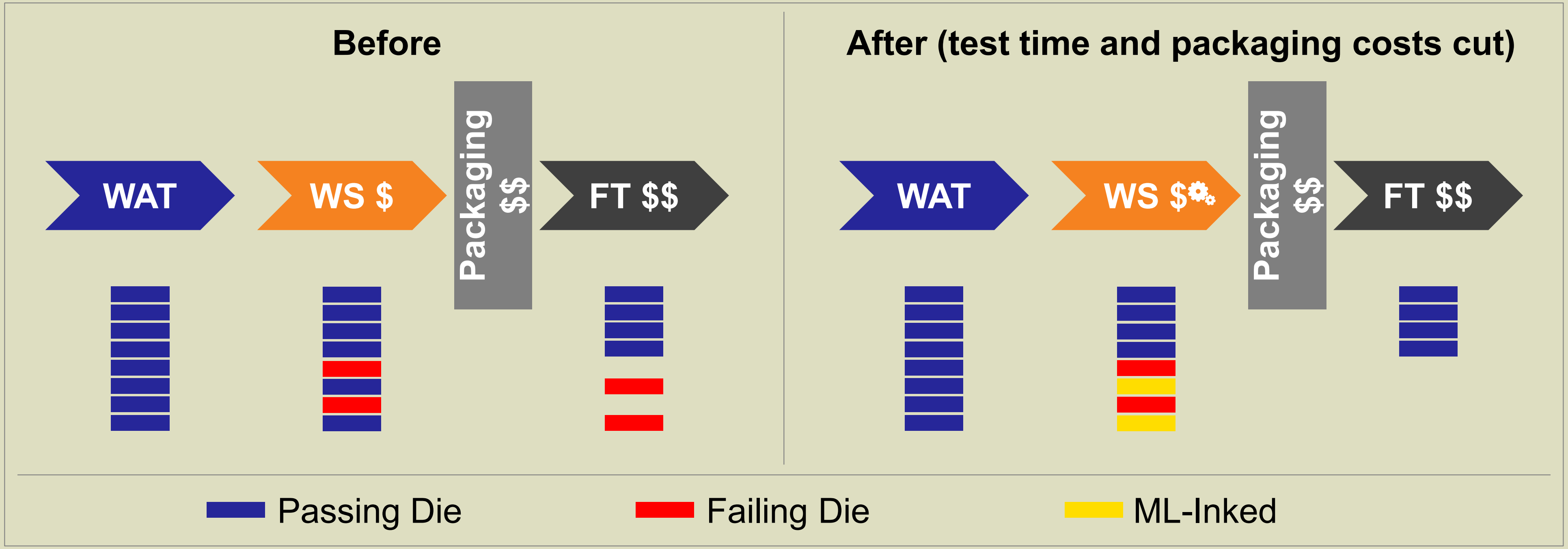
- Consolidating historical data
- For ongoing enhancement and traceability of production at the die level

Responding to product engineering challenges, R&D helped to elevate FT yield

- Lead to a proof of concept (POC) using Advantest‘s machine learning know-how
- Addressing fluctuating yields and high packaging costs
- Exemplified by our work on an 80-pin GaAs HF IQ Modulator ASIC

One Challenge – Early bad die detection ('Shift failures left')

Maximizing ROI with **ML: Predicting** FT outcomes from WS data **cuts** potential **test escapes** by 36% and **minimizes overkill**, **saving** on **costly steps** like packaging.



Before

WAT → WS $ → Packaging $$ → FT $$

After (test time and packaging costs cut)

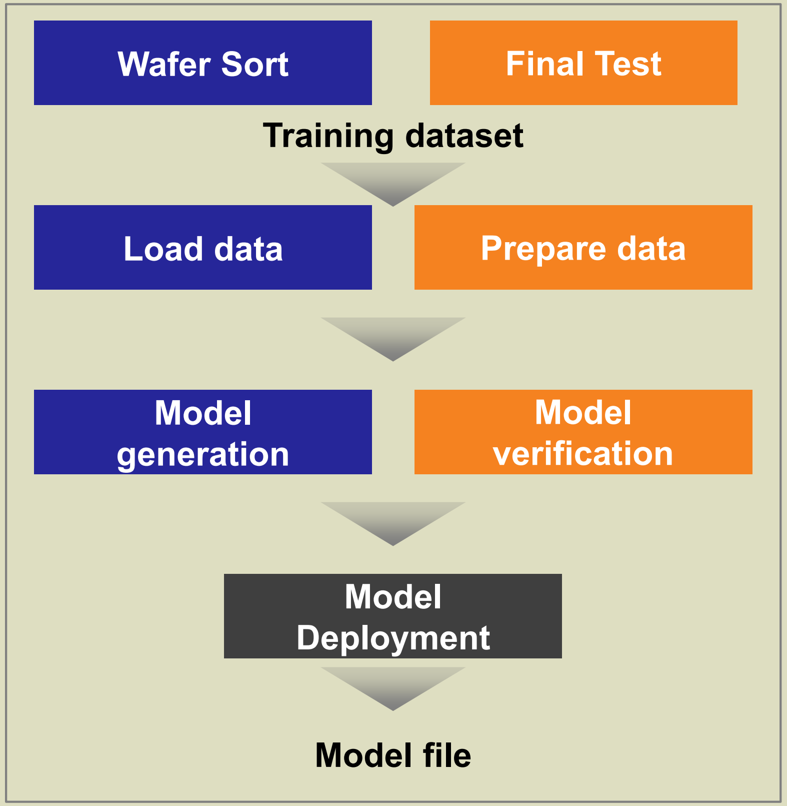
WAT → WS $ → Packaging $$ → FT $$

🟦 Passing Die
🟥 Failing Die
🟨 ML-Inked

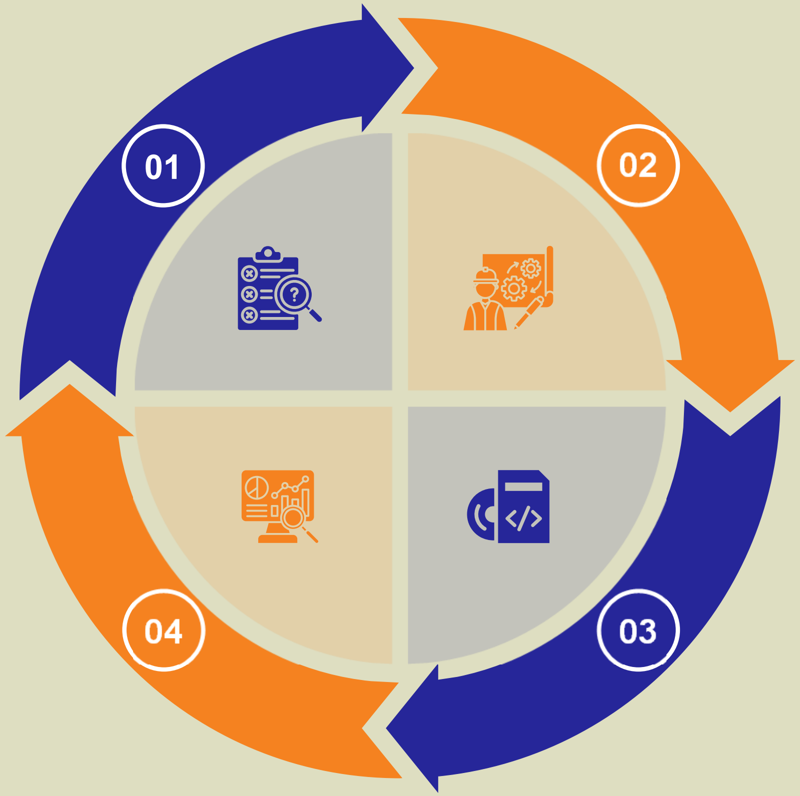
The Method

Detailed Steps

Load Data	Via API pull data from the database, which is part of the integrated workflow
Prepare Data	Clean the data (invalids, NAN, duplicates). Create a training and verification data set.
Create Model	Create a model to predict FT from WS data.
Verify Model	Apply to ‘fresh’ data in the verification set to analyze model quality. Pull and use new data on demand.
Deploy Model	Save model and predictions.
Monitor Model	Ensure performance level.



Deployment – Machine-Learning Lifecycle



Problem Exploration & Understanding

- Activities:
  - Visual data exploration
  - Clean data availability
  - Assess potentials
- Outcome: Find opportunities

Model Engineering

- Data science part
  - Training of models
- Use-case specific implementations
  - ML models
  - Customer applications

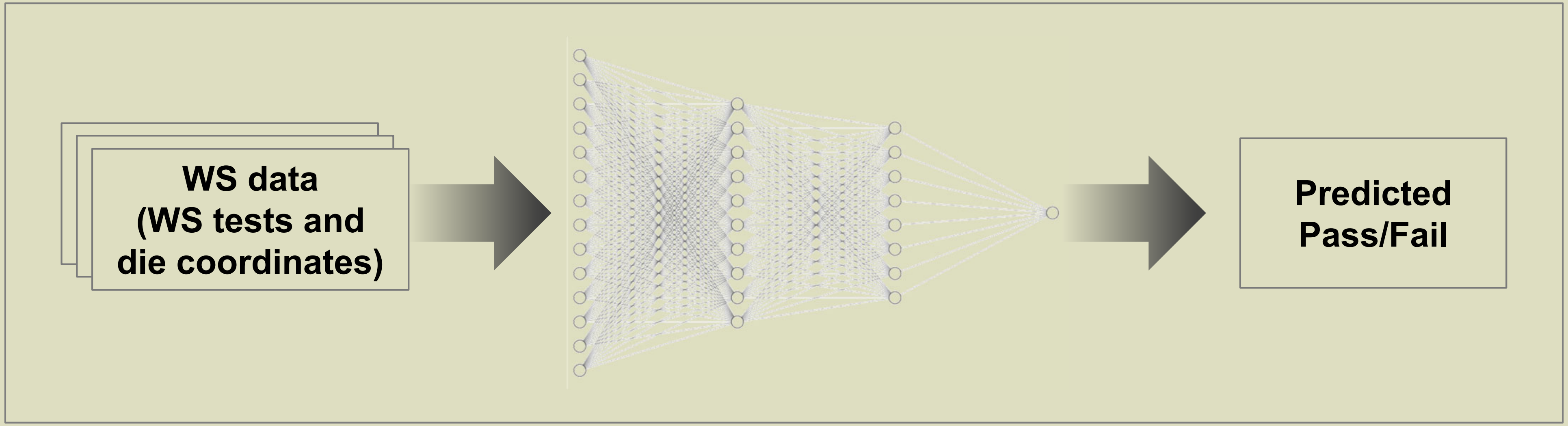
Software/ Firmware Implementation Deployment & Execution

- Secure test floor integration
  - Traceable deployments
- High-performance execution
- Ease of use

Monitoring & Validation

- Constant monitoring of effectiveness
- Detect environmental changes
  - Process variations
  - Test setup
  - Device changes

The Model



A Neural Network was used for forward prediction

700+ WS measurements with die coordinates were used as inputs for the neural network

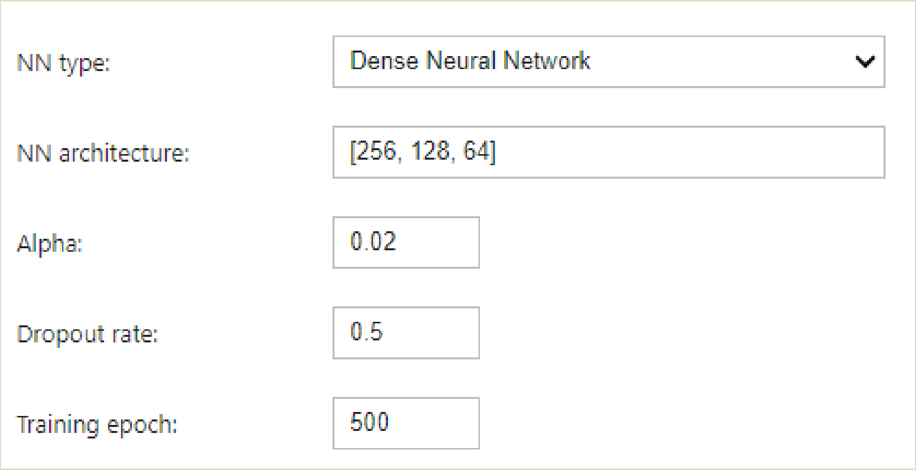
The output (the prediction) was a value between 0-1 with a tunable threshold (fail vs pass)

Easy to use – Neural Network Training in one Click

Python package easy to integrate

Neural networks trained at the click of a button

Highly tunable with different neural networks supported



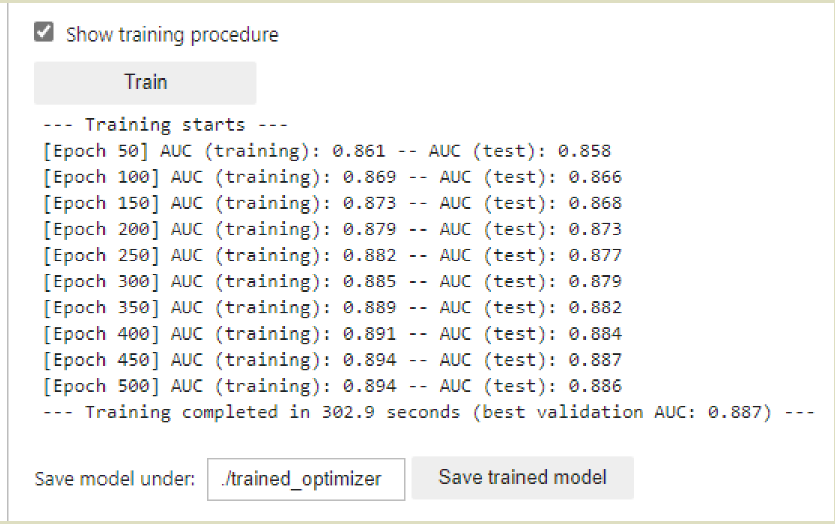
NN type: Dense Neural Network

NN architecture: [256, 128, 64]

Alpha: 0.02

Dropout rate: 0.5

Training epoch: 500



☑ Show training procedure

```
--- Training starts ---
[Epoch 50] AUC (training): 0.861 -- AUC (test): 0.858
[Epoch 100] AUC (training): 0.869 -- AUC (test): 0.866
[Epoch 150] AUC (training): 0.873 -- AUC (test): 0.868
[Epoch 200] AUC (training): 0.879 -- AUC (test): 0.873
[Epoch 250] AUC (training): 0.882 -- AUC (test): 0.877
[Epoch 300] AUC (training): 0.885 -- AUC (test): 0.879
[Epoch 350] AUC (training): 0.889 -- AUC (test): 0.882
[Epoch 400] AUC (training): 0.891 -- AUC (test): 0.884
[Epoch 450] AUC (training): 0.894 -- AUC (test): 0.887
[Epoch 500] AUC (training): 0.894 -- AUC (test): 0.886
--- Training completed in 302.9 seconds (best validation AUC: 0.887) ---
```

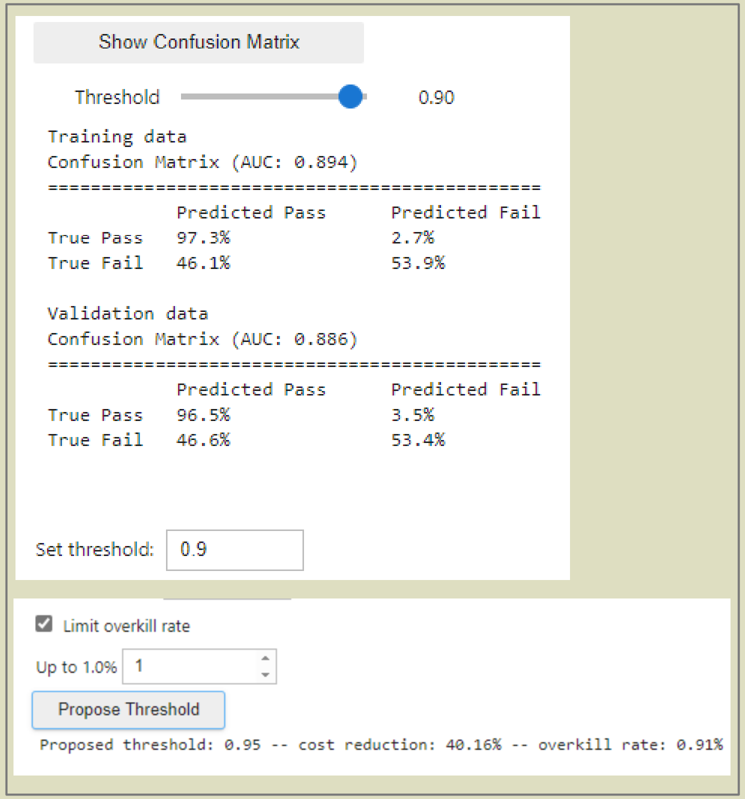
Save model under: ./trained_optimizer

Real-time Results

Real time threshold tuning & performance statistics

Threshold suggestion and capping of overkill

Cost reduction estimate



Threshold 0.90

```
Training data
Confusion Matrix (AUC: 0.894)
==============================================
            Predicted Pass       Predicted Fail
True Pass   97.3%                2.7%
True Fail   46.1%                53.9%

Validation data
Confusion Matrix (AUC: 0.886)
==============================================
            Predicted Pass       Predicted Fail
True Pass   96.5%                3.5%
True Fail   46.6%                53.4%
```

Set threshold: 0.9

☑ Limit overkill rate

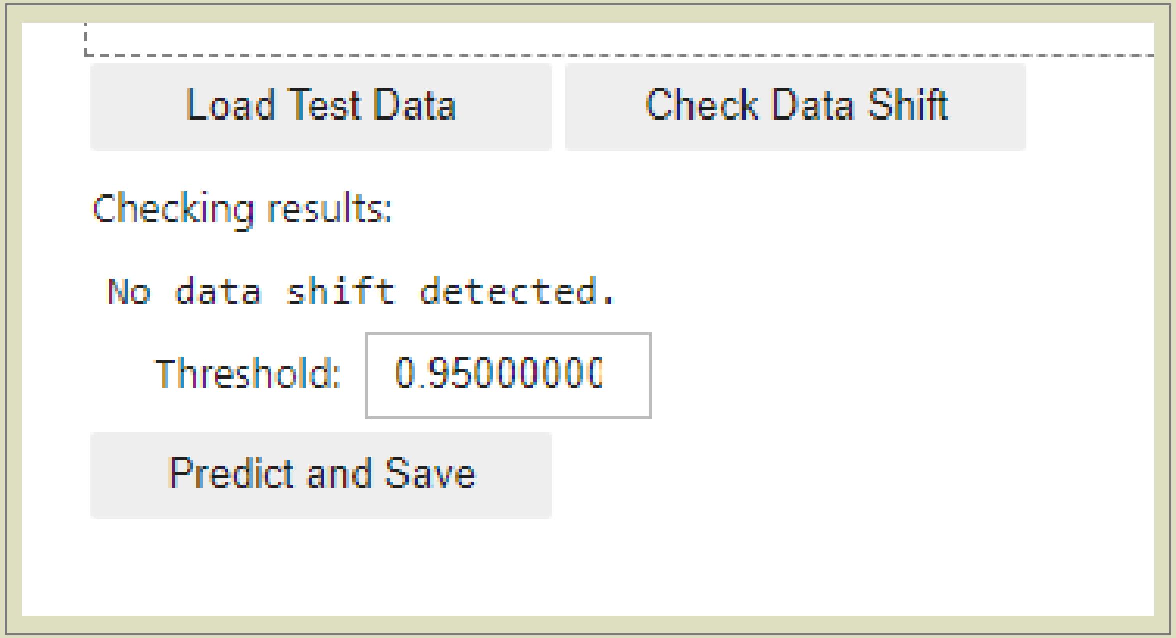
Up to 1.0% 1

```
Proposed threshold: 0.95 -- cost reduction: 40.16% -- overkill rate: 0.91%
```

Monitoring

First stage monitoring to check the model continues to perform as required

Save and use real time in production



Checking results:

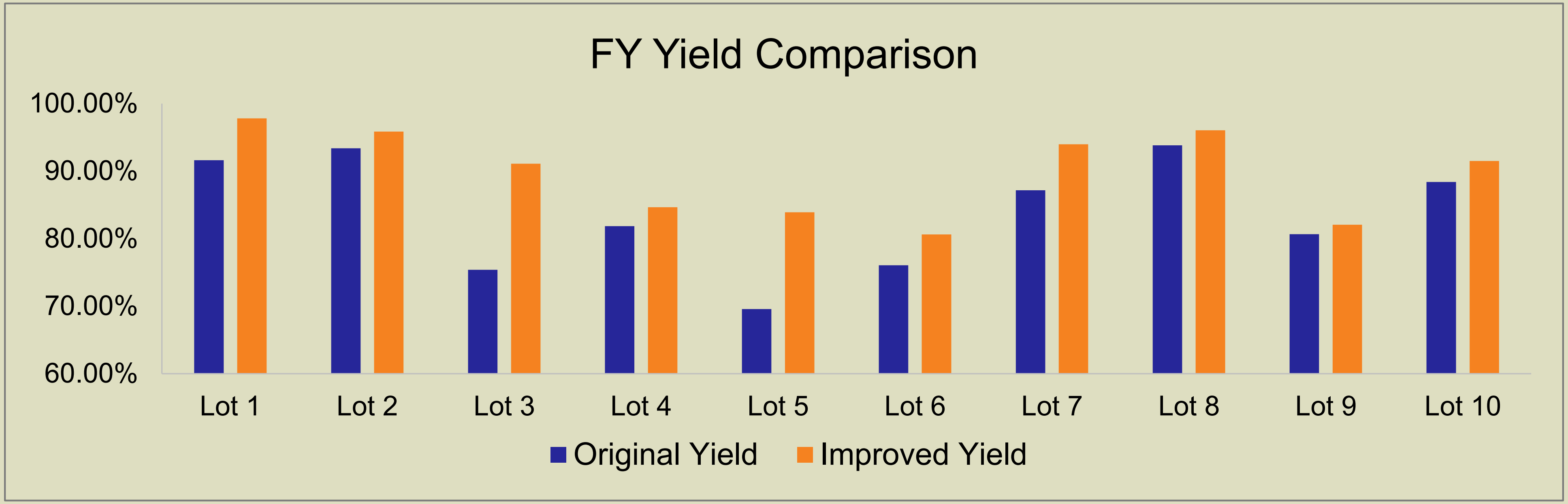
```
No data shift detected.
```

Threshold: 0.9500000(

Setup

- **1D Convolutional Neural Network was chosen as the backbone architecture.**
- **Our method was trained on wafers from 11 lots and evaluated on 10 different lots.**
  - Corresponds to approximately 160k training samples and 155k evaluation samples
- **Training took approximately 5 hours on a T4 GPU and prediction (inference) was done in real time.**
- **This work was on a different ASIC to the previously presented VOICE work. (A complex ASIC going into a multichip BGA package)**

Main Results



FT yield was **improved** by more than **5.4%** in average.

In total, the cost at FT was correspondingly **reduced** by approximately **40%.**

Conclusion - Advancing from WS to FT with Efficiency

Implemented ML to **predict** FT pass/fail from WS data, achieving **significant cost** and **time reductions**.

Integrated state-of-the-art techniques into a **user-friendly** GUI, **allowing** model **customization**.

Detected over **50%** of **test escapes**, yielding **40% cost savings** at the FT stage.

Delivered as a portable Python package for seamless production integration.

Initial Success

From training data

	Predicted PASS	Predicted FAIL
True PASS	98.05%	**1.95%**
True FAIL	55.30%	**44.70%**

From fresh data/verification

	Predicted PASS	Predicted FAIL
True PASS	94.63%	**5.37%**
True FAIL	63.84%	**36.16%**

- Selected top 10 WS predictors from 227 metrics to forecast FT results across 9 lots//~47 wafers/3800 devices.
- Model application indicates a potential 36% reduction in test escapes with <6% overkill.
- Adjustable model tuning to balance cost and failure probability.
- Achieved a substantial increase in FT yield to ~95%, optimizing costs with the high ratio of packaged part expense.

Continued Work Advancing ML Predictions

Integrated ML for robust outlier detection.

03

02

Use case flexibility with optional variable selection enhancement.

04

Seamless data integration with existing workflows.

01

Developed complex neural networks for advanced predictive modelling.

05

Demonstrated success on a secondary ASIC, predicting FT pass/fail from over 700 WS tests.

COPYRIGHT NOTICE

www.testconx.org