

Rapid Deployment of Data Mining in Engineering Applications

IEEE STC 2014

Nikhil Dakwala

ndakwala@broadcom.com

Broadcom Corporation, Irvine, CA

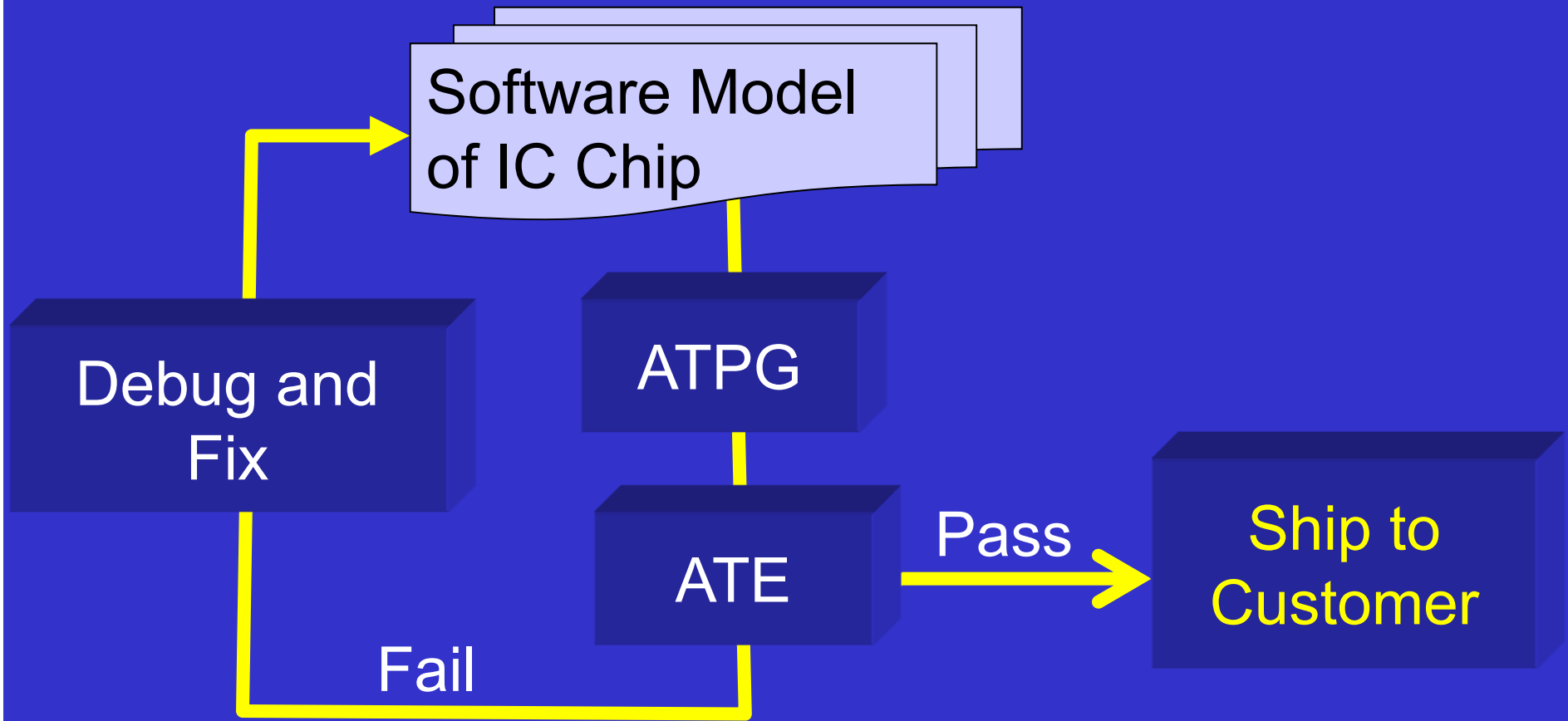
Presentation Outline

- Terms and acronyms
- Need for data mining
- Supervised learning-based data mining
- Cluster analysis through Fuzzy clustering
- Decision trees
- Rapid deployment in Perl
- Conclusion

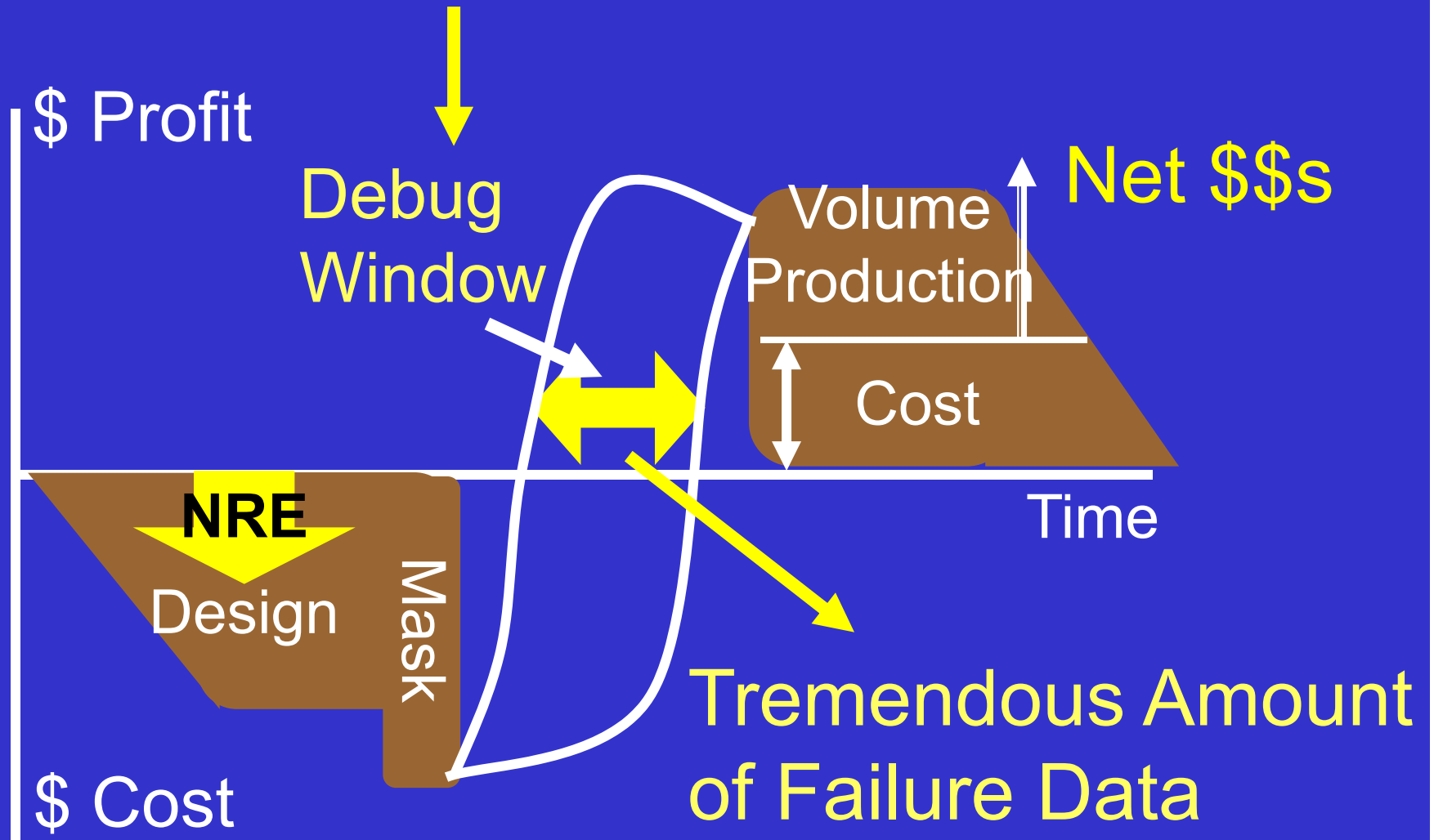
Terms and Acronyms

- Silicon: Integrated circuit (IC) chip
- ATE: Automatic Test Equipment (testers)
- ATPG: Automatic Test Pattern Generation
- TVF: Three operating corners: temperature, voltage, and frequency
- SA: Short at VDD or short at ground faults
- Shmoo: Steady ramp up/down of voltage vs. frequency in regular steps, for a pattern or group of patterns, on ATE
- CDC: Clock Domain Crossing
- DM: Data Mining
- DT: Decision Tree

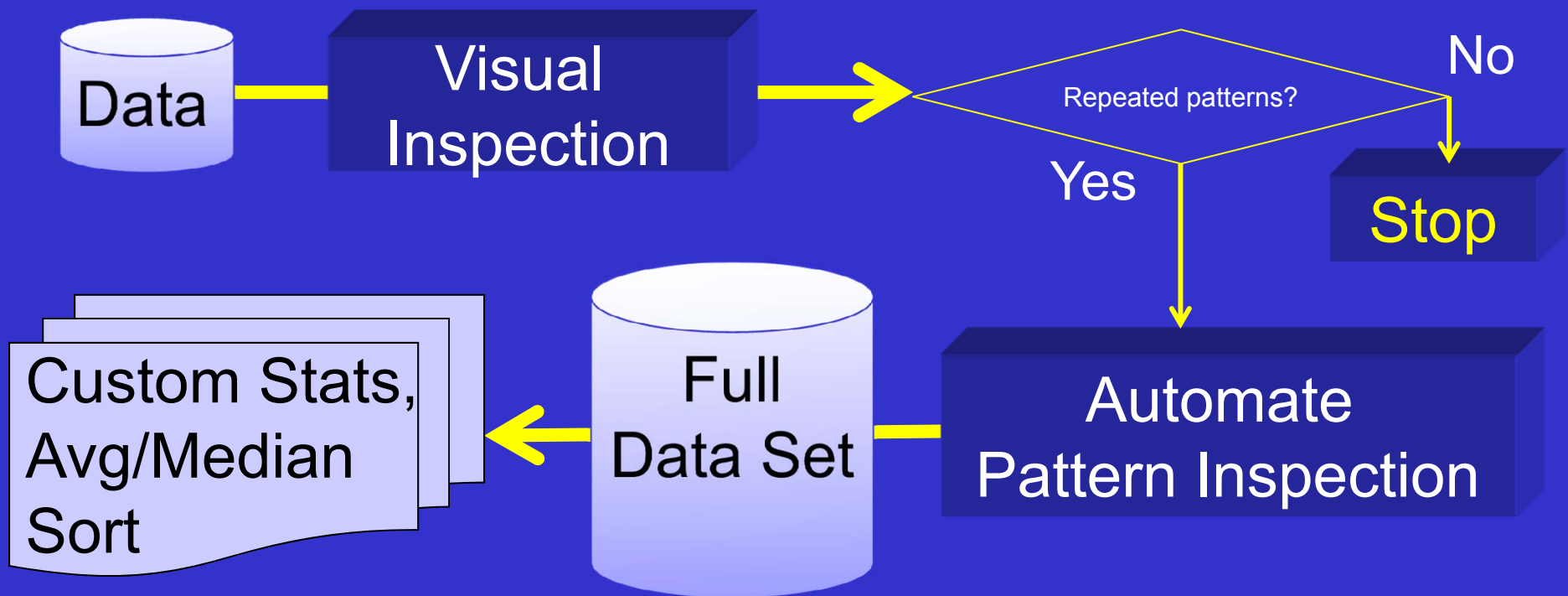
IC Test and Debug Flow



Need for Data Mining



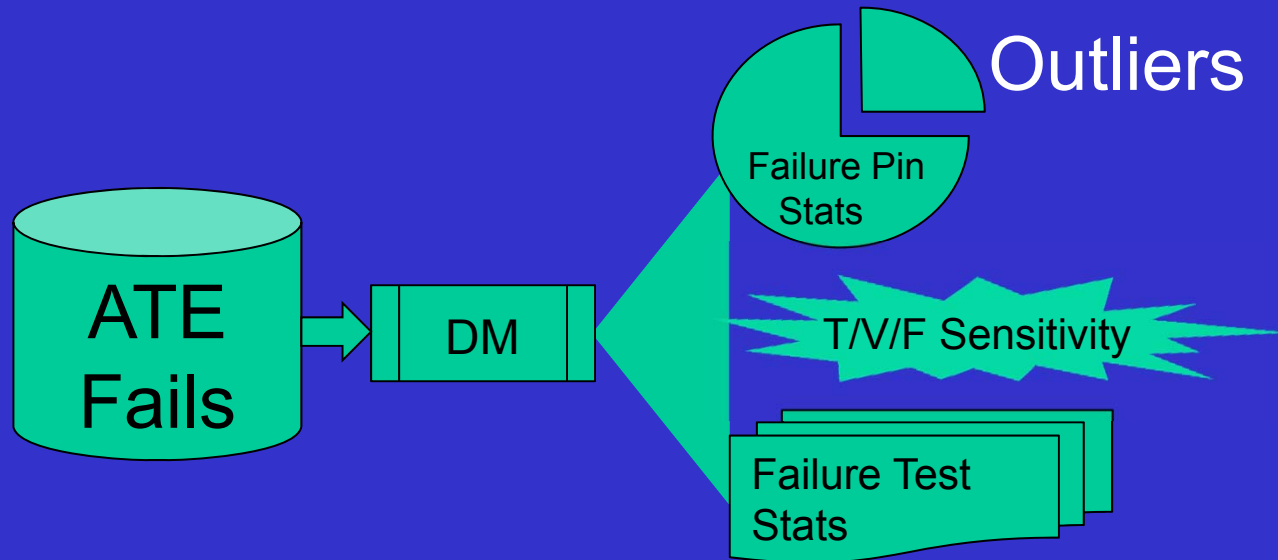
Data Mining: Supervised Knowledge Discovery



Data Mining Techniques

- Cluster analysis
 - Fuzzy clustering
- Decision trees

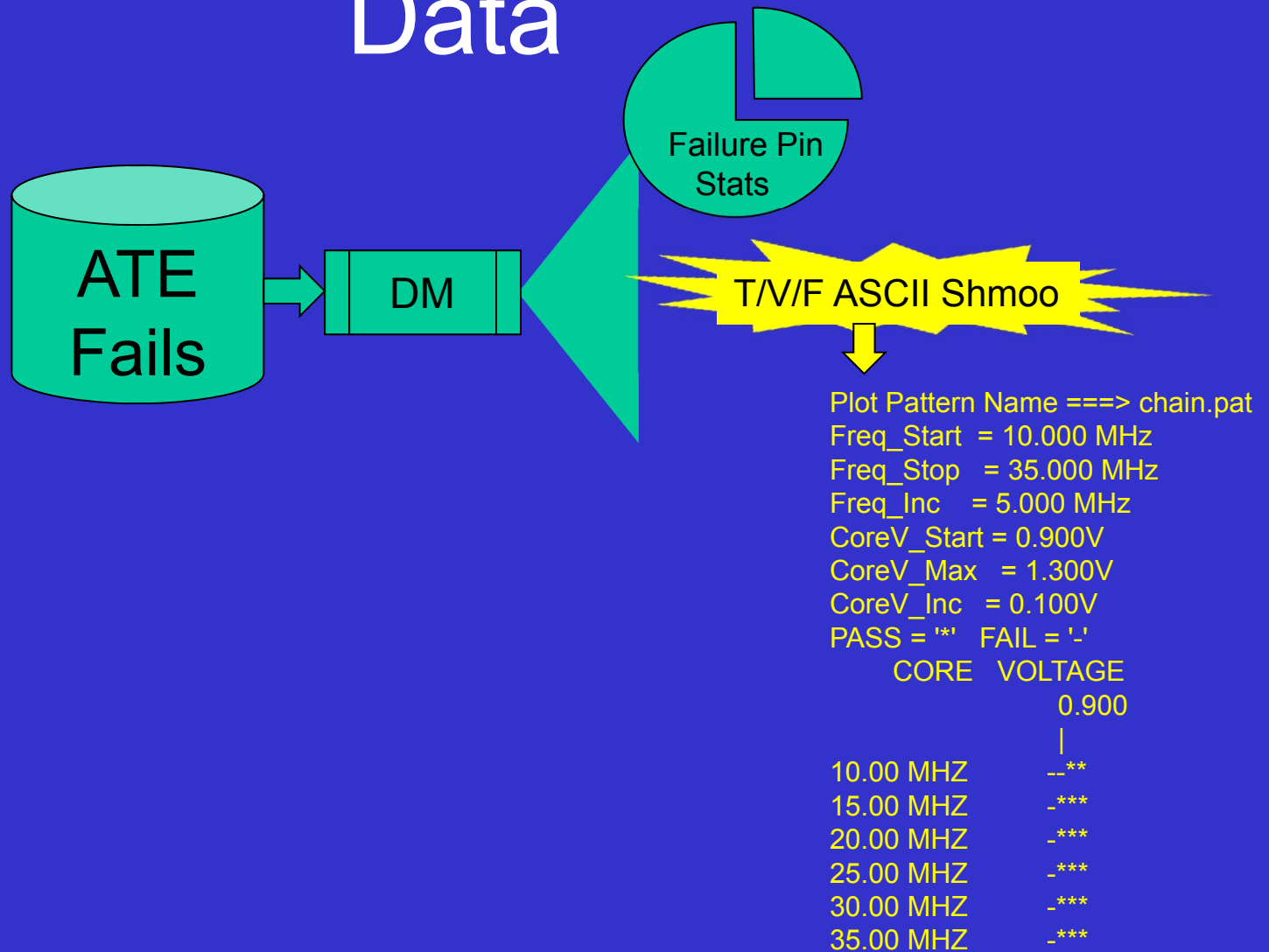
Clustering: Mine ATE Logs



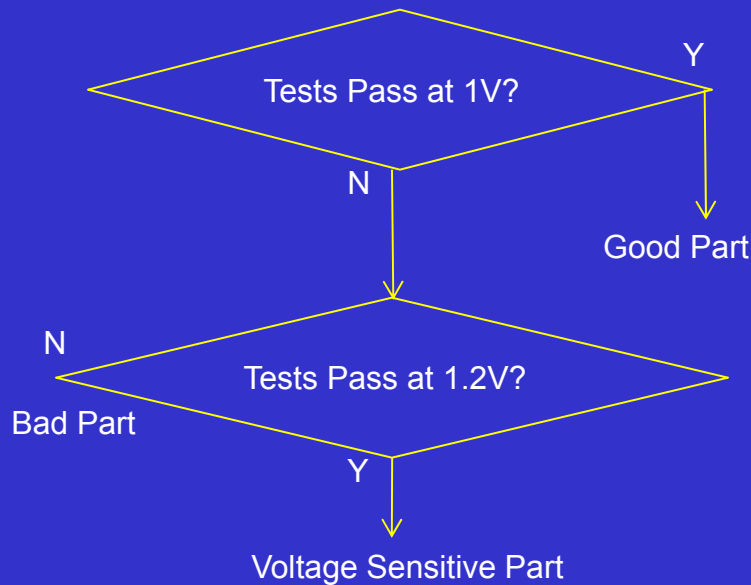
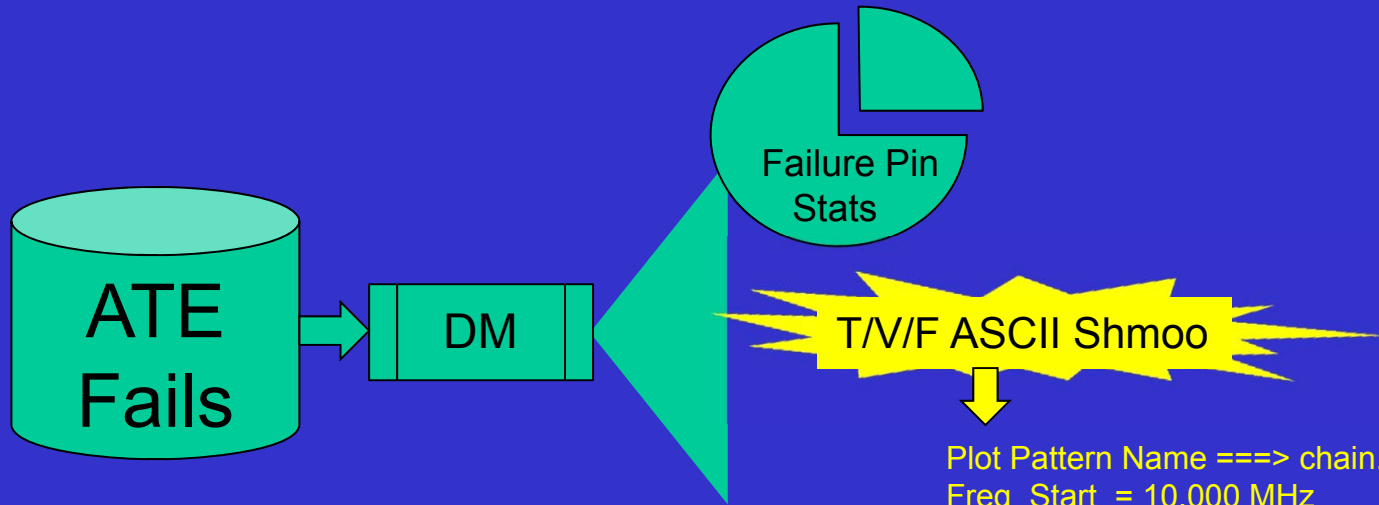
Fuzzy Clustering

- /DATA_FROM_MEM_PIPELINE
- /DATA_FROM_MEM_PIPELINE
- /DATA_FROM_MEM_PIPELINE
- /DATA_FROM_MEM_PIPELINE
- /DATA_FROM_MEM_PIPELINE
- /DATA_FROM_MEM_PIPELINE
- /DATA_FROM_MEM_PIPELINE
- /DATA_FROM_MEM_PIPELINE
- /DATA_FROM_MEM_PIPELINE

Decision Trees: Analyze Failure Data



Voltage Sensitivity Decision Tree



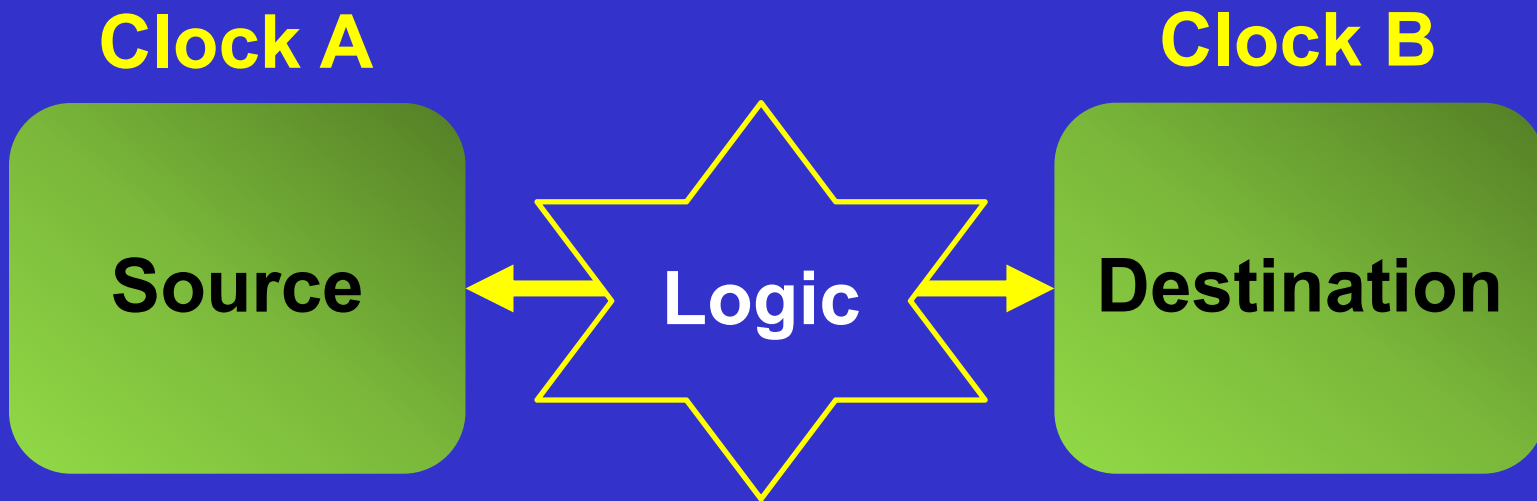
If-then-else parsing,
T/V/F failure statistics



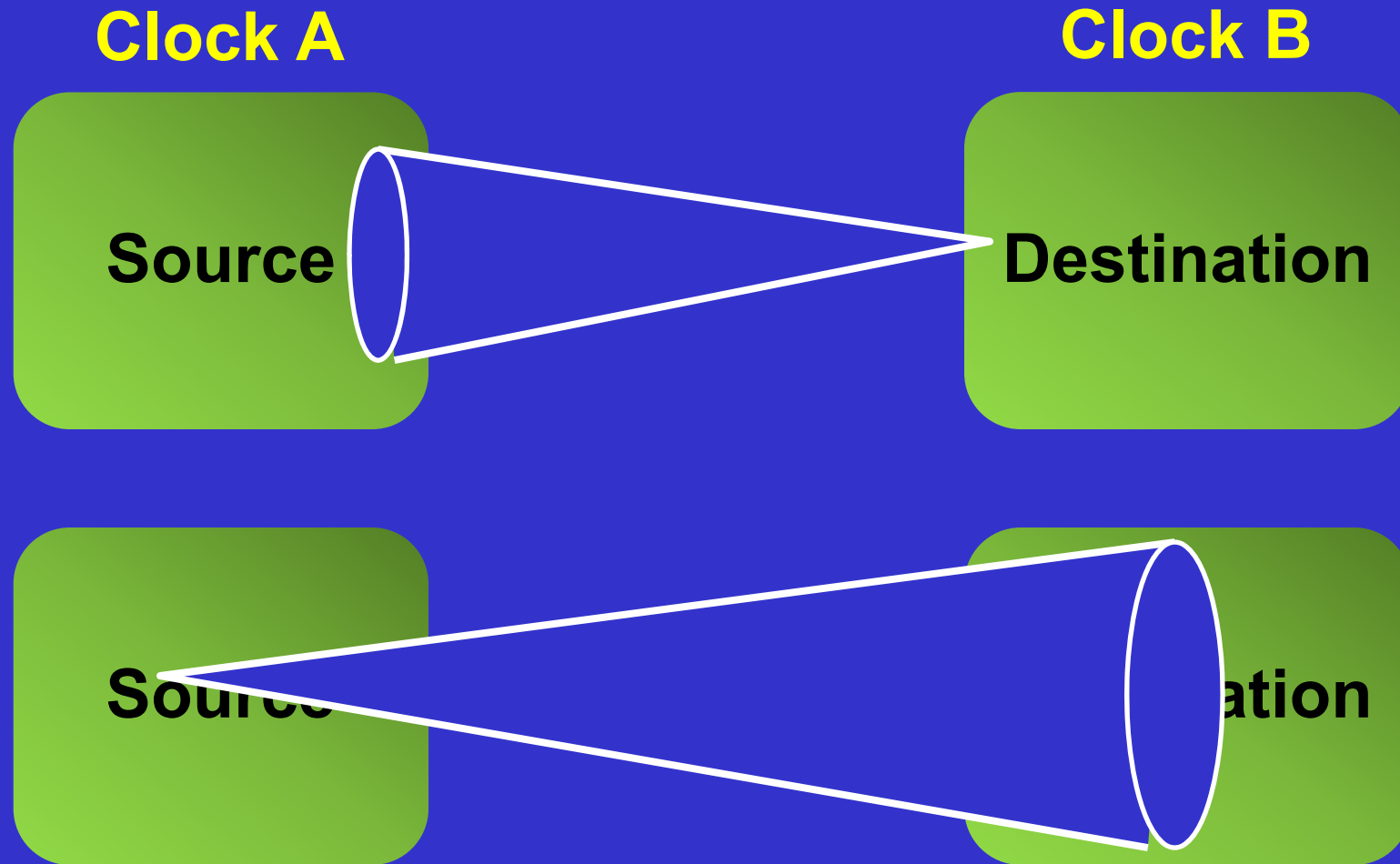
Plot Pattern Name ==> chain.pat
 Freq_Start = 10.000 MHz
 Freq_Stop = 35.000 MHz
 Freq_Inc = 5.000 MHz
 CoreV_Start = 0.900V
 CoreV_Max = 1.300V
 CoreV_Inc = 0.100V
 PASS = '*' FAIL = ' '

	CORE VOLTAGE
	0.900
10.00 MHZ	_*_*
15.00 MHZ	_*_*_*
20.00 MHZ	_*_*_*
25.00 MHZ	_*_*_*
30.00 MHZ	_*_*_*
35.00 MHZ	_*_*_*

Rapid Deployment: Clock Domain Crossing (CDC)



Problem: Testing Huge Cones



Perl-Based Rapid Deployment

- Regular expressions.
- Pattern matching and substitutions are ideal for fuzzy clustering.
- Hash tables and sorting.

DM: Parse CDC Data

#extract clock

#extract end-point

#process_store_node (\$clock,\$end_point,"*end_point*")

→ 3rd argument is data-direction in hash data storage

#extract start-point

#process_store_node(\$clock, \$start_point,"*start_point*")

Cluster Subroutine

#store node according to data-direction, and update counts

Data structure to store full nodes

```
$dat_hash{full}{$direction}{$node}{count}++;
```

```
$dat_hash{full}{$direction}{$node}{clocks}
```

Fuzzify and store fuzzy statistics

```
$fuzed_node = &fuzzify_name($node);
```

```
$dat_hash{fuz}{$direction}{$fuzed_node}{count}++;
```

```
$dat_hash{fuz}{$direction}{$fuzed_node}{clocks}
```


Fuzzy Clustering in Perl

```
sub fuzzify_name {  
    local ($fn_name) = @_;  
    $fn_name =~ s/reg_*/ig;           #remove register string  
    $fn_name =~ s/inst_*/ig;        #remove instance string  
    $fn_name =~ s/_top//g;          #remove hierarchy  
    $fn_name =~ s/\d+_*/ig;         #remove bit indices  
    $fn_name =~ s/_+/_/g;           #remove hierarchy delimiter  
    return $fn_name;  
} #sub fuzzify_name
```

Conclusion

- Rapid deployment of data mining.
- Leverage engineer's knowledge.
- Extract intelligence, outliers.
- Higher-level programming languages like Perl, Python, TCL, etc.

Q&A

Chip Life-Cycle & Defect Costs

