An Agile Cause Analysis Using JMP®

Georgia Z. Morgan, Sr. Statistician, Retired

Abstract

Cause analysis is a general term applied to the tasks used to investigate, analyze and eventually determine the causal factors of a single, or multiple, anomalies. Advanced machine learning methods and other data mining tools have reduced the time and effort to analyze data. Data prep, the steps to acquire data and transform it into information, require domain expertise. To root cause a complex, manufacturing process, it is likely that the domain experts are from different organizations and the data from different sources. A simulated, but representative, conditional anomaly will be used to demonstrate the benefit of using the JMP® virtual join and other tools for a cause analysis that takes multiple iterations.

Keywords: anomaly**,** RCA (Root Cause Analysis), data mining tools, semiconductor manufacturing, JMP® virtual join, JSL, reentrant process. IC (integrated circuit), FEOL (Front End Of Line), BEOL (Back End Of Line), EOL (End Of Line, typically, yield and performance testing and Die Prep), PM (Preventive Maintenance), BKM (Best Known Method), die (a computer chip).

Introduction

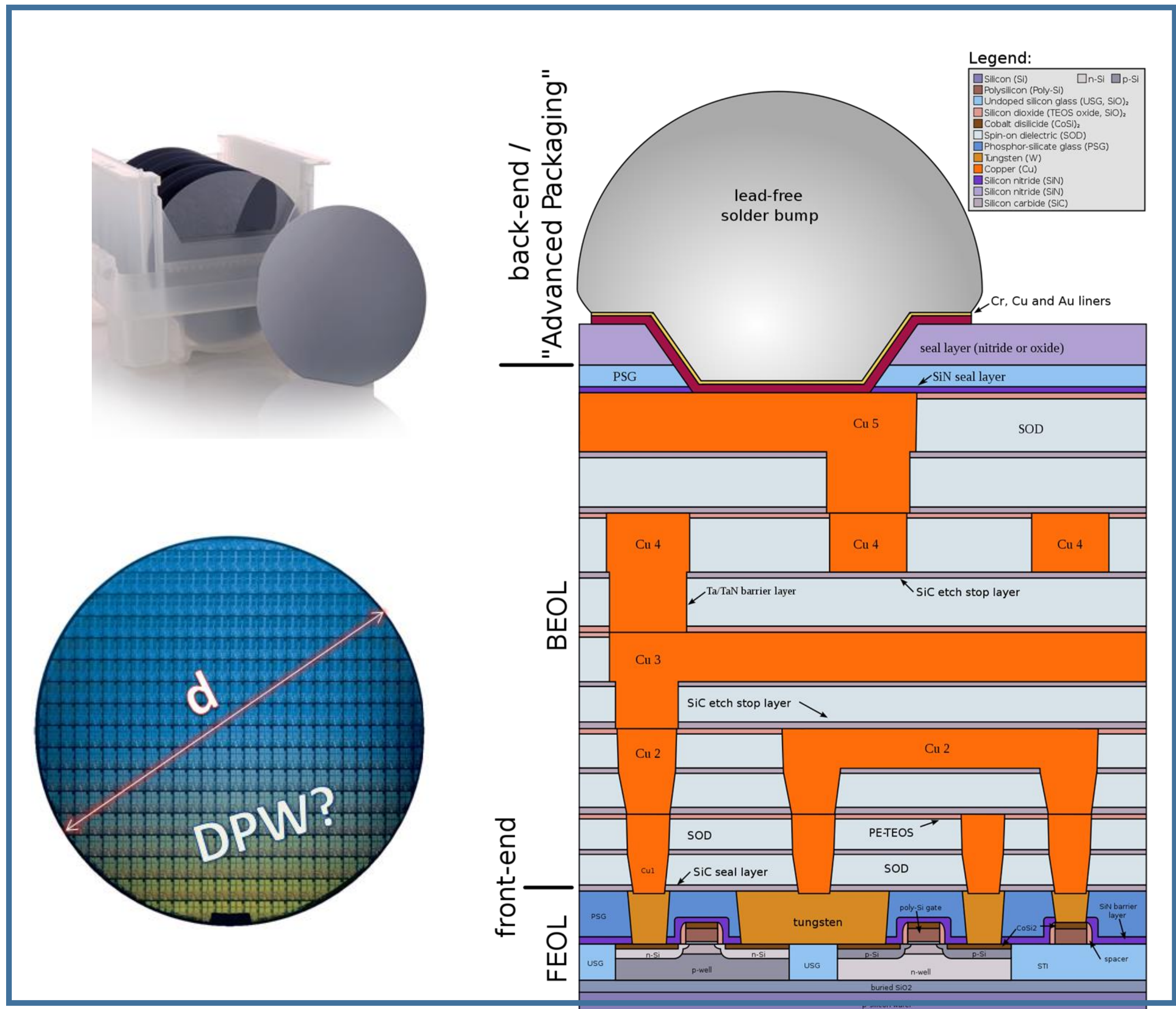
The best protection against failures and unexpected results are the development of robust products and procedures through experimentation and characterization. This applies to pharmaceuticals, medical protocols and manufacturing. An unexpected reaction to a treatment on humans or an abnormality in a complex microchip can be due to a single factor or the interaction of multiple factors. Due to the complexity of the human species, or semiconductor manufacturing, characterization never encompasses all that can go wrong. A single abnormality can trigger a troubleshooting investigation, but most often, a root cause analysis is initiated after multiple instances of what appears to be the same anomaly.

This paper will present two different scenarios, both are fictional but constructed from semiconductor manufacturing. The first is a simple example to support the thesis that root cause analysis often requires multiple data restructuring. After an overview of the JMP virtual join, the second example demonstrates the value and agility provided by this feature.

Integrated Circuits, ICs, are created from numerous processing steps of: oxidation, implantation, deposition of films and sputtered metals with different conductivity and barrier (non-conductivity) characteristics, patterning, planarization, and of course cleaning and measurement. It starts with a bare, doped silicon wafer [1] and ends up with multiple die (computer chips) per wafer. The maximum number of die per wafer depends upon the wafer size (diameter) and die size[2]. See Figure 1(L)

Figure 1(R) is a cartoon depicting a very small segment of a chip with a 5-layer Cu metal interconnect BEOL.[3] Even though the architecture, metal thickness, layout and possibly the recipe, might be different for each Cu layer, typically, they are processed in the same factory area (fab bay) and on the same tools.

Figure 1 (L) Carrier of 25 bare silicon wafers with flat and fully processed wafer with notch (R) An IC graphic of a five layer Cu interconnect BEOL.

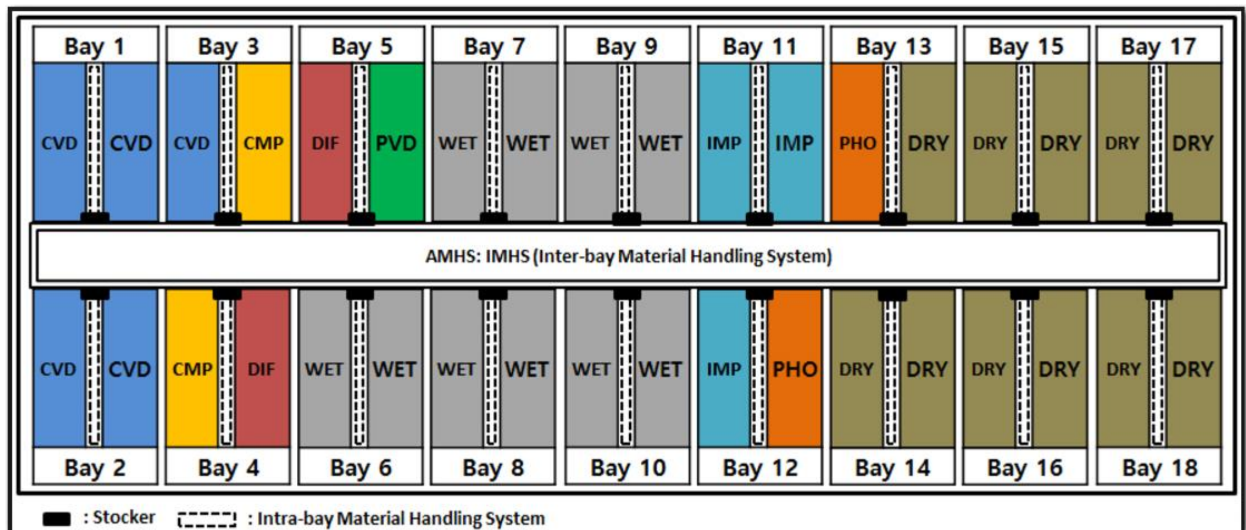


The FEOL and BEOL are segmented. Even a minute amount of Cu quickly diffuses through oxides creating defects and degrading the gate oxide quality.

Figure 2 depicts a possible fab (factory) layout for a FEOL [4]. Processing equipment of the same type and function are located in bays. Their layout is optimized for plumbing to bulk chemical delivery, exhausts and drains. An overhead rail system, called an automatic material handling system (AMHS) carries fab lots from one station to the next, or temporarily stores them in a remote stocker until a tool becomes available. There is no standard layout, each semiconductor corporation has its own factory planning group. Most of the factory queues are reentrant. For example, the four tools in Bays 7 and 9 might perform seven different clean steps. Knowledge of the factory flow and equipment usage is essential for root cause analysis. Using this example, a single tool could potentially affect a fab lot up to seven times, and some fab lots might never have been processed on that piece of equipment.

See the references [7] or do a quick web search for videos and other sources to learn more about semiconductor manufacturing.

Figure 2 Possible fab layout for a FEOL process



Example #1- Broken Cu Metal Lines

Scenario: A 5 metal Cu layer BEOL is being processed in a factory without state-of the-art visual inspection nor inline electrical test (E-test). Defects are found at EOL (end of line) with tests, strip back and visual inspection. Two lots are found to have a slight degradation for yield (die per wafer). Strip back points to Metal 1 broken lines. Soon intermittent lots with varying degrees of yield loss appear with one or several layers of broken metal lines. The BEOL is shut down. A root cause analysis team, RCA team, is convened. The team has a BEOL leader, and a key representative from lithography, wet cleans, metal deposition, metal etch, polisher, spin on dielectrics (SOD) and AHMS, wafer handling. They meet at each shift, or when new data is available.

At the first meeting, each functional rep is told to gather information and assess if their area might be the source. A file of lot number and yield is posted. A lot level file was chosen since the wafers showed similar yield for both good and bad lots. They will meet again at 4pm to review the data. Note broken lines can occur due to bad chemistry, defects, or any vibration in a tool or fab construction. Each fab area rep should look for:

- changes in bulk chemicals,
- new replacement parts,
- changes in preventative maintenance (PM) procedures or scheduling,
- any unusual SPC monitors,
- whether the tools are calibrated and do they all meet operational specifications for gas flow, pressures, temperatures, etc.
- also, post your area’s tool history data for the posted lots.

The wet clean rep with some statistical help pulled together a file and did some preliminary work using JMP. The engineer flags the RCA lead and the team convenes early at 2pm. The BEOL has 7 wet cleans: two cleans for metal 1 and one for each of the other 4 layers, and a final clean when it leaves the area. RSD is a single chamber toolset and there are 5 tools supporting the BEOL clean area (BECLN). A snippet of the joined data is seen in Figure 3. A variability chart (Figure 4), and 1st best split by the Partition platform (Figure 5L) indicate tool RSD03 is the potential culprit.

Figure 3 RCA Tool History Joined with Lot Yield Data (157 lots x 7 oper = 1099 rows)

Example1 LowYield - JMP Pro

File Edit Tables Rows Cols DOE Analyze Graph Tools Add-Ins View Window Help

Example1 LowYield

Notes Simulated data fo

Variability Chart

Decision Tree of yield

Columns (6/0)

Fab Lot

Yield (DPW) Lot Avg

Tool

Tool ID

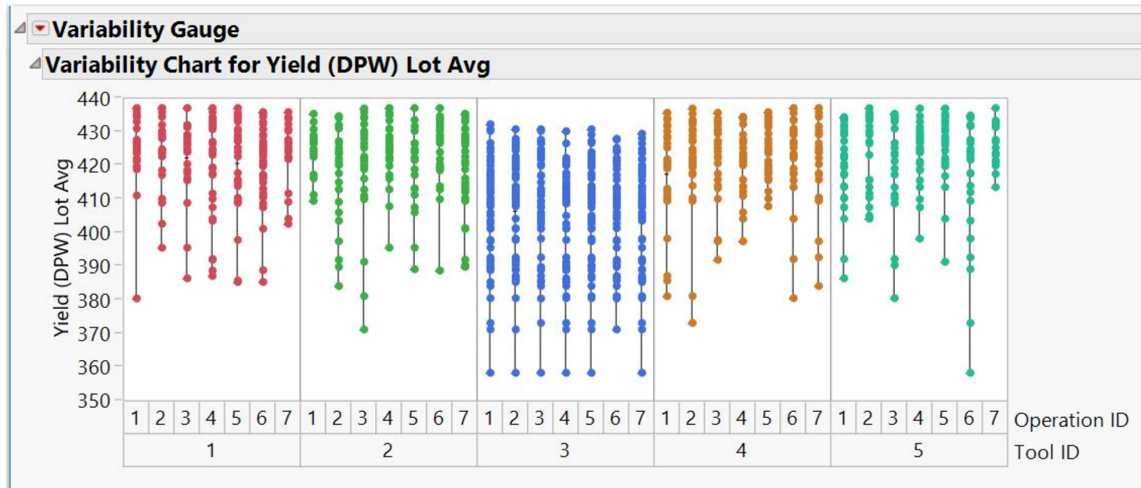
Operation

Operation ID

Rows	
All rows	1,099
Selected	0
Excluded	0
Hidden	0
Labelled	0

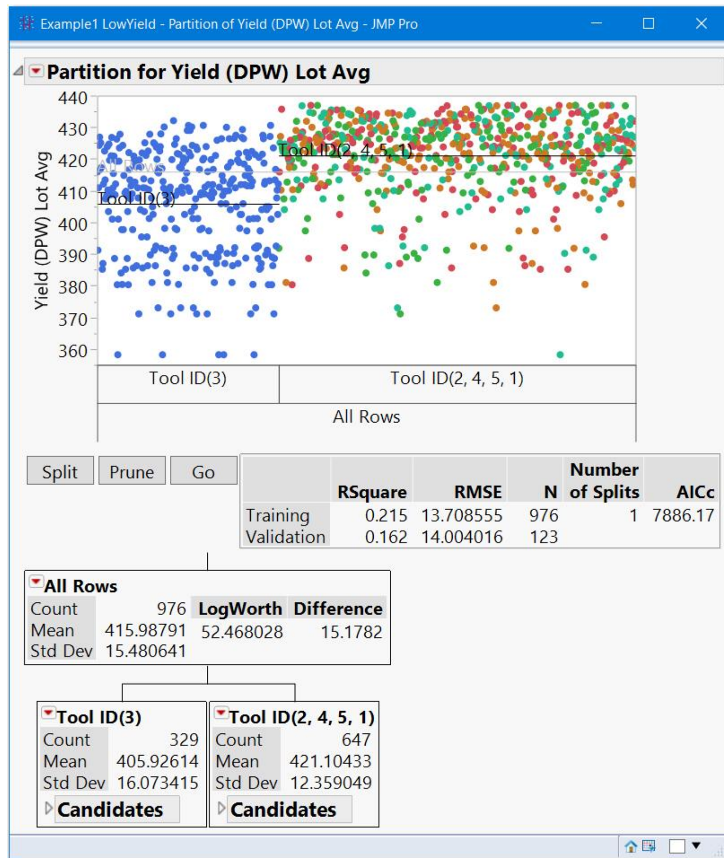
	Fab Lot	Yield (DPW) Lot Avg	Tool	Tool ID	Operation	Operation ID
1	L00043	436.9	RSD01	1	BECLN01	1
2	L00043	436.9	RSD01	1	BECLN02	2
3	L00043	436.9	RSD01	1	BECLN03	3
4	L00043	436.9	RSD02	2	BECLN04	4
5	L00043	436.9	RSD01	1	BECLN05	5
6	L00043	436.9	RSD04	4	BECLN06	6
7	L00043	436.9	RSD05	5	BECLN07	7
8	L00098	415.4	RSD03	3	BECLN01	1
9	L00098	415.4	RSD05	5	BECLN02	2
10	L00098	415.4	RSD01	1	BECLN03	3
11	L00098	415.4	RSD05	5	BECLN04	4
12	L00098	415.4	RSD04	4	BECLN05	5
13	L00098	415.4	RSD04	4	BECLN06	6
14	L00098	415.4	RSD03	3	BECLN07	7
15	L00166	418.9	RSD04	4	BECLN01	1
16	L00166	418.9	RSD03	3	BECLN02	2
17	L00166	418.9	RSD05	5	BECLN03	3
18	L00166	418.9	RSD01	1	BECLN04	4

Figure 4 Variability Chart of Yield with Tool, Operation Groups



RSD03 has no high yielding lots and most of the low yielding lots. Figure 5 (R) shows a low-tech sequence analysis. It is created by splitting the table from Figure 3, splitting column Tool ID by Operation ID. The new columns are combined, and the file is sorted by Yield descending. The high yielding lots never passed though RSD03 and the low yielding lots had passed through RSD03 multiple times.

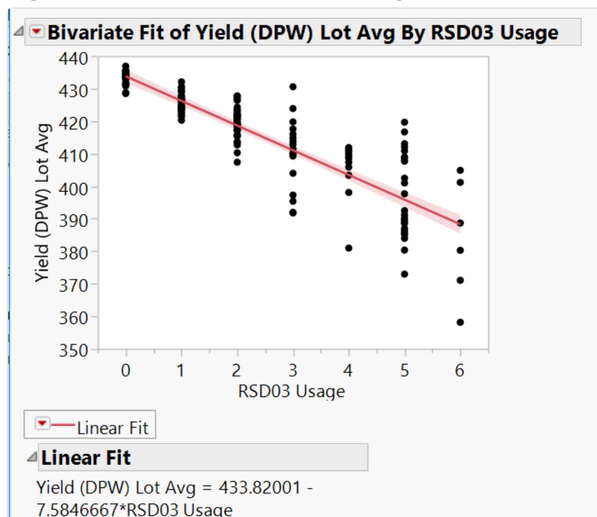
Figure 5 (L) Partition for Yield with Factors Tool ID and Operation ID and **(R)** Yield by Tool Sequence



	Fab Lot	Yield (DPW) Lot Avg	ToolSequence
1	L00043	436.9	1112145
2	L00359	436.9	1511224
3	L00370	436.8	1425524
4	L00592	435.7	1121411
5	L01995	435.5	4542144
6	L00167	435.2	2452512
7	L00428	434.7	1552152
8	L00611	434.5	1255422
9	L00816	434.3	1124514
10	L00319	434.2	5551151
11	L00890	433.9	4224151
12	L01592	433.7	5544421
13	L00946	433.4	4451242
14	L00814	433.2	5552545
15	L00067	432.9	1451122
143	L00853	389.7	3233332
144	L00550	389	3333253
145	L00243	388.7	3333313
146	L00787	388.6	3331323
147	L00228	387	4331333
148	L00262	386.3	5313333
149	L00648	385.6	4333133
150	L00997	385.2	3333113
151	L00790	384	3233334
152	L00372	381	4423333
153	L00019	380.4	3353343
154	L00820	380.3	1333333
155	L00542	373	3433353
156	L00265	371.1	3323333
157	L00637	358.2	3333353

The RCA team takes RSD03 offline, and turns the BEOL on. More is needed. A request to plot yield by the number of passes through RSD03 is requested for the upper management review. The Wet Clean area needs to look for the root cause. The requested management plot is seen in Figure 6.

Figure 6 Yield versus RSD03 Usage



Further investigation shows the spin speed on RSD03 to be within spec, but faster than the other 4 tools. Further experimentation shows that faster spin speed on any RSD tool causes tool vibration that results

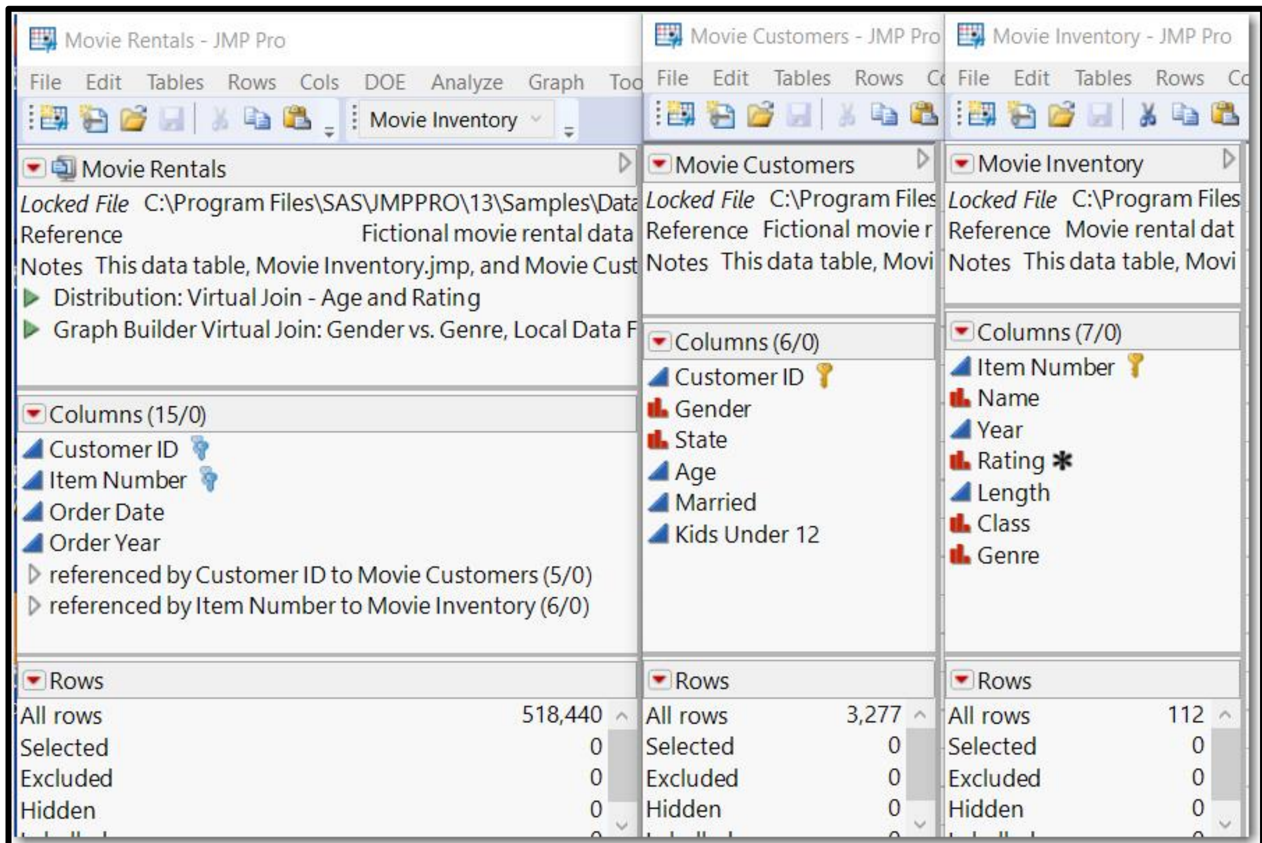
in line breakage. Vibration is the root cause, caused by spin speed. With excellent engineering, and experimentation, the wet clean recipe is optimized reducing spin speed without a significant increase in clean time. The learning from this RCA team results in an overall 8 DPW improvement; from an average of 433.8 DPW to 442.1 DPW.

This simulation is a simple example, since the root cause was contained to one tool in one toolset and one fab area. When the root cause is a the much more difficult problem of cross contamination involving multiple toolsets, time between events or complex event sequences, data management can be cumbersome. Consider the table in Figure 3. Each toolset needs columns for Tool ID and Operation ID, so unique names need to be agreed upon before combining the data. Now, look at column ToolSequence in the data table shown in Figure 5R. It represents the tools used for the 7 clean steps. This side-by-side (unstacked) layout is conducive for data mining. The next section will provide an overview the JMP Virtual Join, and how it might be beneficial for root cause analyses.

JMP Virtual Join

The JMP Virtual Join feature allows a main table to link to multiple auxiliary tables without physically joining them. The virtual join is defined by two column properties, the Link Reference for the main table and the Link ID for the auxiliary table. The JMP book, ***Using JMP***, contains a Netflix like movie rentals example. The data table **Movie Rentals.jmp** contains numerous movie rental transactions. Each transaction includes a Customer ID and an Item Number that is associated with a movie CD in the inventory. The data table **Movie Customers.jmp** contains customer demographics. It is virtually joined to rentals by Customer ID. The data table **Movie Inventory.jmp** contains the movie name, ratings, genre, etc. It is virtually joined to rentals by Item Number.

Figure 7 JMP book, ***Using JMP***, Virtual Join Example for Movie Rentals



Items to Note for Virtual Joins

- The Link Reference column (main table) and the Link ID (auxiliary table) must have the same column name and the same data type.
- The Link Reference appears as a blue shadowed key and Link ID is a gold key when they are linked.
- Columns of the referenced (linked) data table are displayed as grouped variables in the main table. Click on the disclosure icon to display the columns. These columns can be used in formulas, graphs, analyses and table commands.
- Open these 3 movie files, from the JMP Sample Data directory. Since the column properties are defined once opened the tables are automatically linked.
- If multiple auxiliary tables are to be virtually joined, each table requires a unique column to define the Link Reference.
- Link ID values must be unique. For the movie example, a customer id can appear in many rows of the main table, Movie Rentals, but the matching customer id must only appear in one row of the Movie Customer data table.
- Columns with data type Expression or Row State cannot be used for linking.

Unlike the movies example, a cause analysis in a semiconductor factory will likely join multiple auxiliary files by the same identifier, lot, or wafer number.

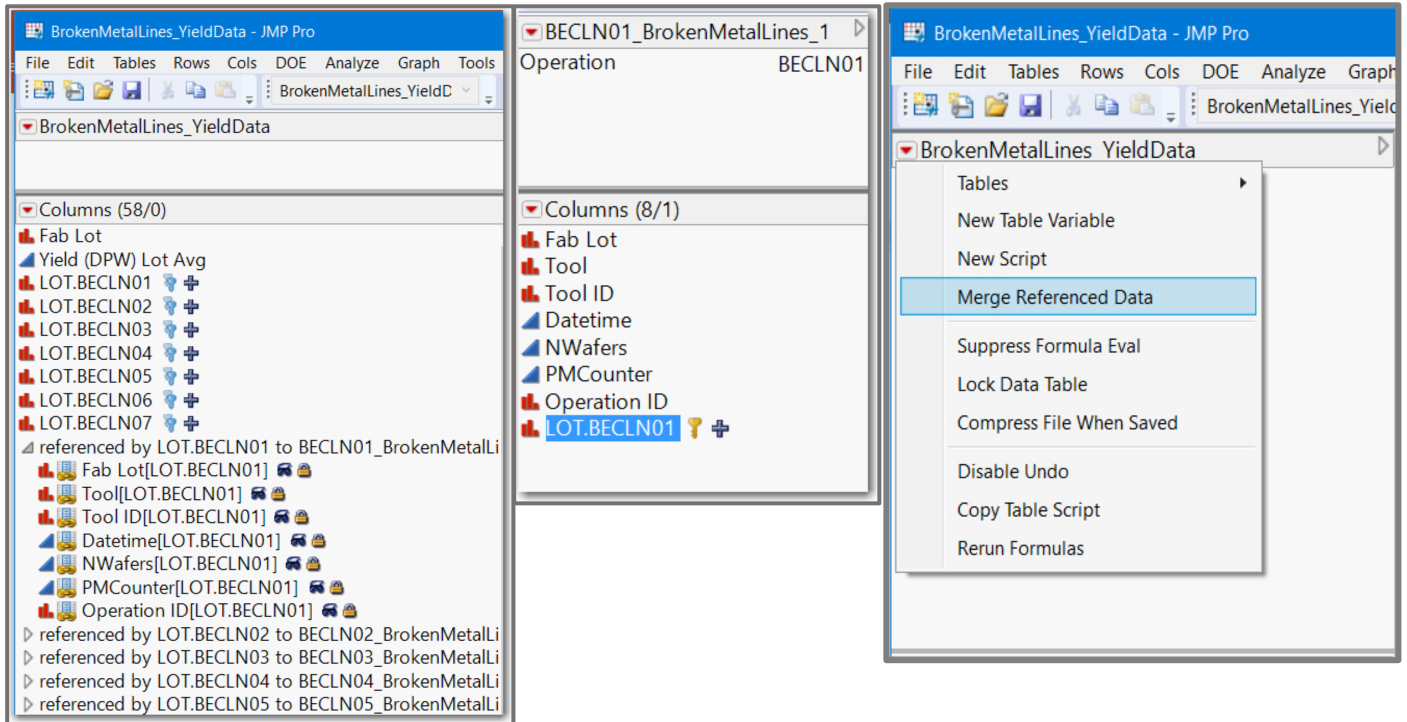
Consider the simulated scenario of Example 1. The root cause analysis team wants to apply some data mining tools to a compilation of tables from multiple fab areas. Each fab area will create a table for each manufacturing step (operation), that belongs to their area. The BE clean engineer creates 7 tables, one for each of the 7 clean steps. Since this tool event data is extracted from the same factory manufacturing system database, the tables will likely have the same column names. Figure 8 displays 3 of the 7 tables for the back-end (BE) clean steps.

Figure 8 Fab Equipment Data for 3 of 7 Operations, Same Column Headers,

The BECLN engineer decides to use the virtual join method on his files joined with the yield data. Figure 9 (L) depicts the virtually joined main table for Example 1. The first two columns, Fab Lot and Yield

(DPW) Lot Avg, represent the anomaly data. For each of his tables, a character column named LOT.BECLN0x is created with the formula(:Fab Lot) and assigned the column property **Link ID**. Figure 9M shows the table panel for file BECLN01_BrokenMetalLines_1.jmp.

Figure 9 **(L)** EOL Yield/Performance/Anomaly Data Virtually Joined with Equipment Data for 7 Operations, **(M)** BECLN01 Auxiliary Table Panel with Link ID, **(R)** Table Menu, Merge Referenced Data.



More Items to Note for Virtual Joins

- If multiple auxiliary tables are to be virtually joined at the same time, each table requires a Link Reference. Note that using a formula to create each linked column (Link Reference and Link ID) ensures that the links are updated if new data is added. If a table is created for each processing step, this could represent hundreds of tables. Below are commands, useful for getting and setting the Link Reference using JSL.

```
//run one line at a time, and review the log
column("LOT.BECLN01") << get column properties("Link Reference");
column("LOT.BECLN01")) << get properties list;
column("LOT.BECLN01")) << get property("Link Reference");

/*: Log Results

{Formula( :Wafernum ),
   Set Property( "Link Reference", Reference Table( "<path>\BECLN01.jmp" ) )}

{Formula, Link Reference}

Reference Table( ",path>\BECLN01.jmp" )
```

- If the auxiliary tables are in the same directory folder as the main table, a path name is not required. The path for the link reference is <u>case sensitive</u>. If the RCA team is using both Windows

and Macintosh environments, use posix format for the path, forward slashes. For example, c:/temp/name vs. c:\temp\name. Posix works on both operating systems

- If creating virtual links with the JMP UI, open the auxiliary tables and set the Link ID first. Then open the main table. Right click on the column header to be used as the Link Reference, scroll to the item Link Reference. A list of tables is available to be selected.
- When scripting virtual joins, the linked (referenced) data table (the auxiliary table) must be saved, and the path specified.
- The saved script for a JMP analysis using a virtually joined column produces verbose syntax. So far, I have not seen issues with using the more concise syntax: **:Name(**).

```
Referenced Column( "Tool[LOT.BECLN01]",
      Reference( Column( :LOT.BECLN01 ), Reference( Column( :Tool ) ) ) )

:Name( "Tool[LOT.BECLN01]" ) //concise
```

- If an auxiliary table is no longer needed, close the auxiliary table. The linked columns no longer appear in the main table. I recommend keeping the link reference if there is a possibility of reusing it at some later date, even for a different table.
- Figure 9(R) shows the table option to merge the auxiliary data. If you only wanted tables BECLN01 and BECLN05, highlight those Link Reference columns and select Merge Referenced Data. The default is to merge all.

Managing links for 150 (or more) operations is best done by JSL. See the attached add-in to create and manage links.

Two Main Benefits for Virtual Joins: Naming and Unlink/Relink

Naming

JMP joins, found in menu item Tables → Join, add a phrase "of tablename1" and "of tablename2" to each commonly named column to distinguish the data source. Descriptive table names to distinguish one query from another typically are long; the names often include a datetime or a version number.

A Virtual Join allows you to create a shortened name for the Link Reference and Link ID. Whether you are using an inhouse custom query, SQL, or creating smaller tables and joining, name management usually requires a script.

IC manufacturing RCA teams usually combine EOL process step data or tool data. So the join will be LOT.<operation name or number> or LOT.<tool name>. [See Aggregation for more discussion.]

Control of the link names enables concise and consistent joined column naming, and only the links are named, JMP adds the context names: Tool[LOT.BECLN01]

Unlink/Relink

It is also my experience that most root cause analysis requires multiple iterations of data reshaping, like adding additional data or changing the aggregation. For a large monolithic file, this task is often assigned to the "next meeting," and data from the different fab areas are not ready to join until moments before that meeting begins. Suppose the BECLN engineer noticed the BECLN05 had many rows of missing information. Then he remembers, he did not use the updated lot list when he ran that query. All that he has to do is close the BECLN05 data table. Fix the table and open it. If it has the same file name and stored in the same directory, the main table automatically displays the corrected table. If the file name changed, right click on the LOT.BECLN05 header in the main table and select Link Reference and a list of files will appear, select the new file. [Or use the Link Manager that makes it easier to search.]

Consider another scenario: for the next meeting, the RCA team leader asks the BECLN engineer to add maximum and mean flow rate, temperature and agitation speed to the 7 tables. If the engineer adds these columns to each of the 7 tables and keeps the same table name. During the next meeting the RCA lead can open the main table and 7 auxiliary tables and the new columns are immediately available.

The benefit of the virtual join unlink, by closing the table and automatically (or rapidly) relinking, facilitates area ownership for their data, and distributes the workload.

The JMP add-in can reduce the tasks of creating links, opening tables defined by the Link Reference, and selecting columns and unlinking. Due to the complexity of IC manufacturing, a data mining-ready table might have many hundreds of auxiliary tables. An add-in can save the ergonomic burden of numerous point and click tasks.

It is my opinion that, with a little training, a virtual join add-in, and establishing an RCA team standard for link naming, the JMP Virtual Join provides an agile method for cause analysis teams to iteratively investigate their data without the added burden to get the data joined before the meeting.

For this example, each row represents a fab lot. This naturally joined side-by side format establishes tool sequencing, which tool the fab lot went through for each step. Also, queue times between steps can be easily calculated, etc. To reiterate this side-by-side table layout is the most useful and successful when using tree methods, looking for processing features (patterns) that align with varying quality metrics aggregated by lot.

Aggregation [*optional notes, can be skipped*]

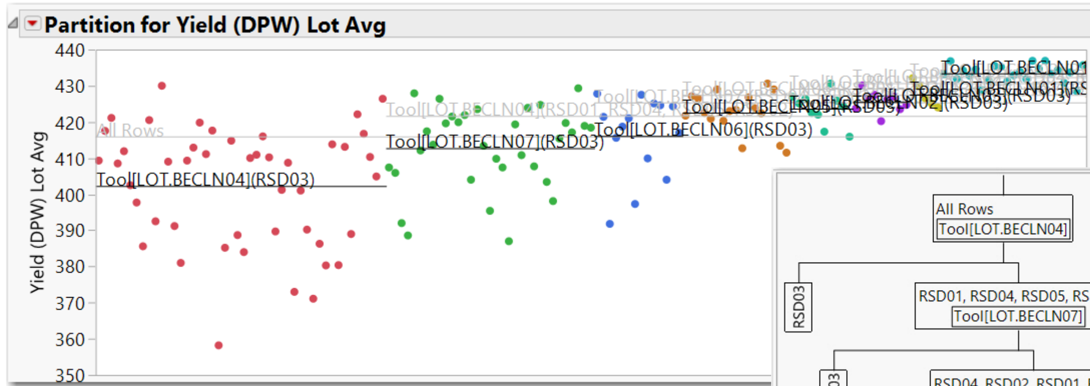
This example is using fab lot aggregation: the yield loss is aggregated as fab lot average die per wafer (DPW). If the yield loss had been seen on intermittent wafers within the fab lot, then the yield data (DPW) would have been aggregated by wafer (Wafernum), and the operation data would be extracted using a wafer number query. Note many semiconductor tools have multiple chambers that perform different functions. So, wafer processing order might have an effect: for example, the last wafer in the lot has a longer pre-process queue time and the first wafer in the lot has a longer post-process queue time.

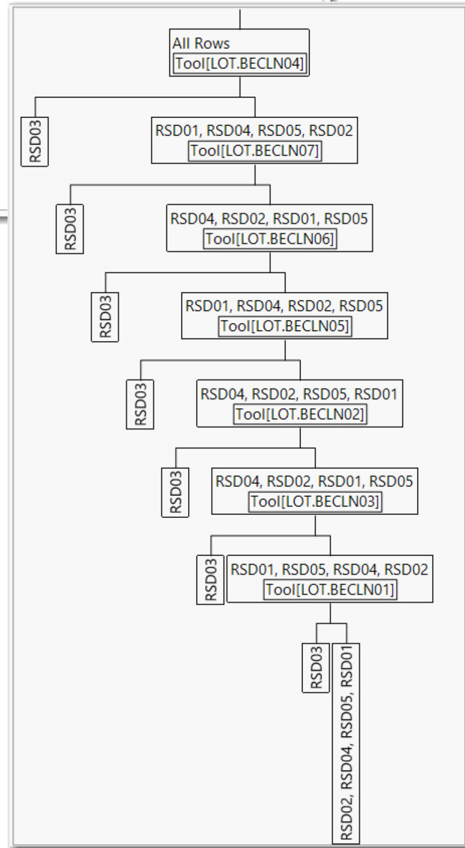
Often an IC manufacturing RCA team starts with the main table, a file organized by lot, or lot and wafer number, with columns for yield, reliability, EOL performance metrics that include both typical and anomalous (out of control) data. A real dataset likely would contain columns for product and process and multiple quality metric columns. If there are products (chips) with different die sizes, then yield and other metric “normalization” must be done on the main table. In cases where the anomaly is seen only on a specific region of the wafer, the main file would contain die level data: the values would represent the die test results and a column for the die location, XColumn and YRow on the wafer would be included. This is called, Lot, wafer or die level aggregation.

Example 1: Side-by-Side Analysis

Figure 5L contains the results of running the Partition platform on the stacked data. It merely hints that RSD03 seems to have lower yield. The subsequent analyses required sorting and restructuring the data to create the column Tool Sequence (or pattern). Figure 10 shows the result of running the Partition platform on the virtually linked table. This shows that for each operation, if RSD03 was used, there is a yield loss. This is much stronger support that RSD03 is the source of the yield loss. Columns Pattern and RSD03 Usage are easy to create and do not require restructuring the main table.

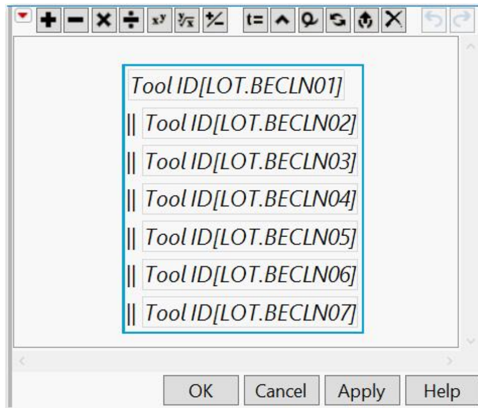
Figure 10 Partition Small Tree View for Yield (DPW) vs. Tool for 7 Back End Clean Steps





No table restructuring required:

Formula to create the column Pattern (Tool Sequence)



Formula to create column RSD03 Usage:

```
N Row( Loc( Words( :Pattern, "" ), "3" ) )
```

The source of the yield loss in Example 1 was RSD03, one tool, in one toolset, in one function area.

Example #2 Intermittent Failing EOL M3 Resistance

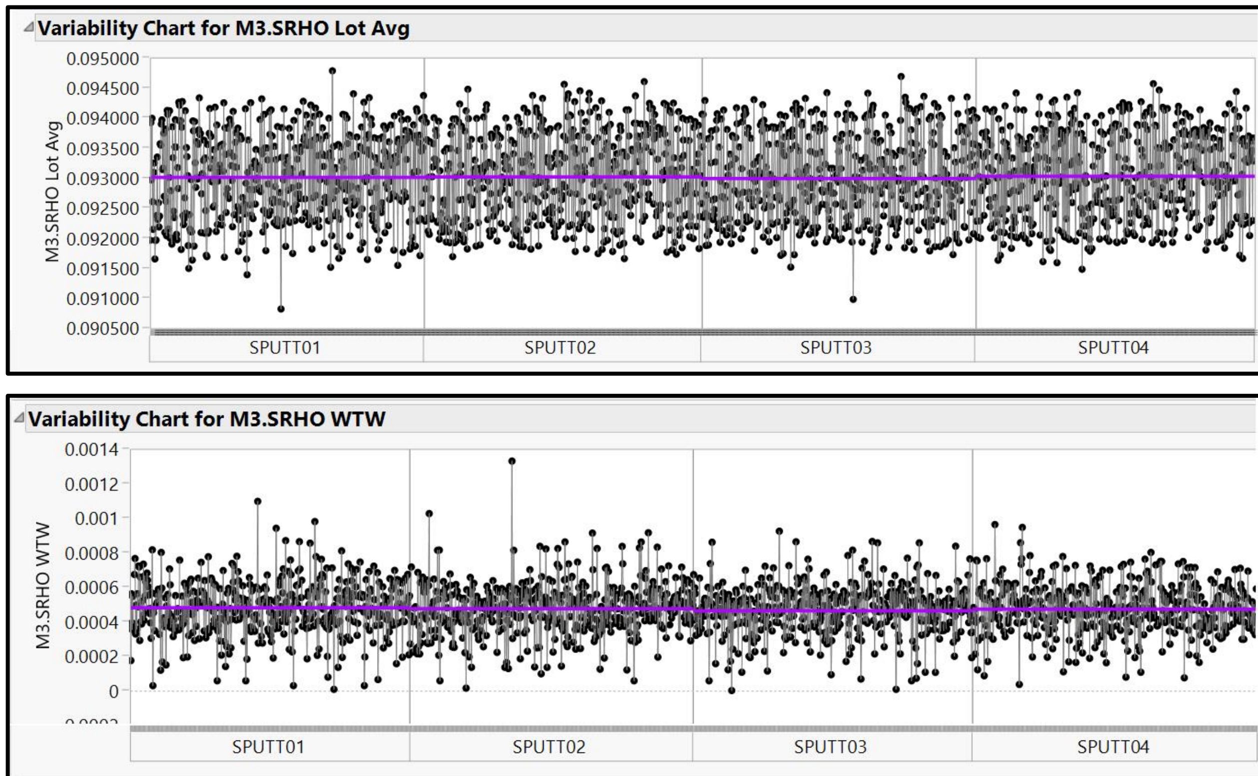
One of the more difficult causes to root out is a contamination problem, especially if the contamination is removed with scheduled maintenance (PM), dissipates with usage and if it is conditional. As stated earlier this simulated example is representative of semiconductor manufacturing. But it is reasonable to expect similar types of anomalies in epidemiology and drug studies and food or drug processing, etc. This type of problem is rare; from my experience less than 5% on a Pareto plot of anomalies. However, it requires an inordinate amount of time troubleshoot.

I have been researching both visual and numeric analytics, as well, as methods to quickly and easily organize the data to investigate and model contamination as a suspected cause. It's my hope this example will return numerous suggestions and leads.

Scenario: After a holiday shutdown, a few intermittent lots' EOL indicators for M3 resistance failed SPC. Some material was so far out of control, wafers had to be scrapped. There were a few anomalous lots, then none were seen. Then the "eventual rule" kicked in: small signals eventually become large signals. A flurry of bad lots appeared at the end-of-line.

The simulated data represents BEOL processing for a 7-layer interconnect metal, that starts with a clean queue (nothing in the queue during shutdown). Figure 11 depicts the baseline vs. the metal deposition tools (sputter tools) used at M3 deposition (dep), and Figure 12 is a similar plot with the post-holiday material.

Figure 11 Baseline M3.SRHO **(T)** Lot Avg **(B)** Wafer-to-Wafer, WTW, Std Dev



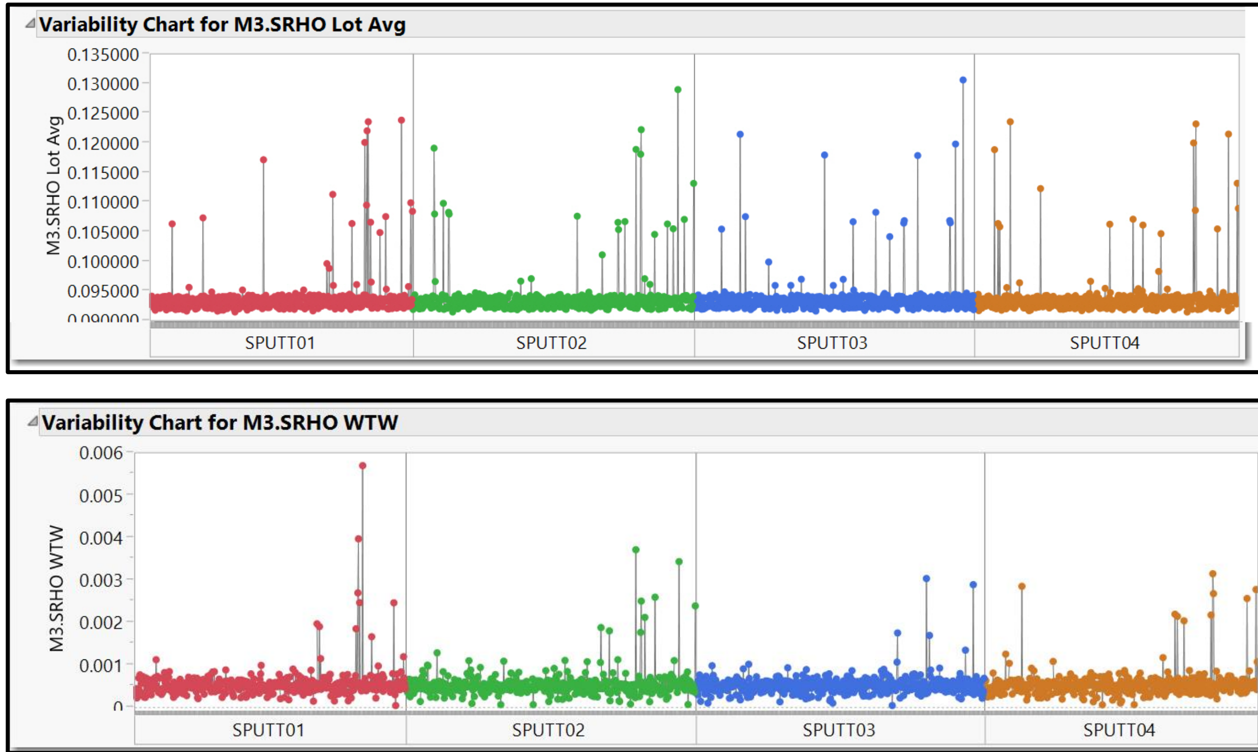
For this type of problem, there is no shortage of potential causes, even looking at just the M3 loop. If lines are too narrow resistance increases (pattern and etch). Defects and contamination can affect the metal conductive characteristics (almost any tool can cause defects).

The narrative for this example continues. At the first RCA meeting, the review of inline critical dimensions (line widths) and SEM visual inspections shows no indication of line size effect or defects. So, the team decides to drill down to the M3 sputter tools. The sputter engineer remarks that because of the holiday, all tools started up with a full PM (clean) and many of the early lots where half lots, 12 wafers per lot. Also, more than usual small quick turn lots of 6 wafers were run. The engineer speculates that recent flurry of bad material might be related to the fact that more 25 wafer lots have been running; in other words, wafer count could be a factor. For this simulation, the sputter tools have 2 chambers; the tools are cleaned after 5000 process wafers, or about 2500 per chamber. SPC wafers are collected every 500 wafers.

The simulation kept the contamination within BE Metal deposition, operations M1-M7, to reduce complexity. A first-in-first-out (FIFO) queue was simulated with the exception that the quick turn, 6

wafer, lots get priority. Random time between other steps, that accounted for wafers per lot, allowed for more realistic variation for time between M1 to M2, M2 to M3, etc. Other than PM and SPC monitors, no other unusual events were included in this simulation.

Figure 12 Post-Holiday M3.SRHO **(T)** Lot Avg **(B)** Wafer-to-Wafer, WTW, Std Dev

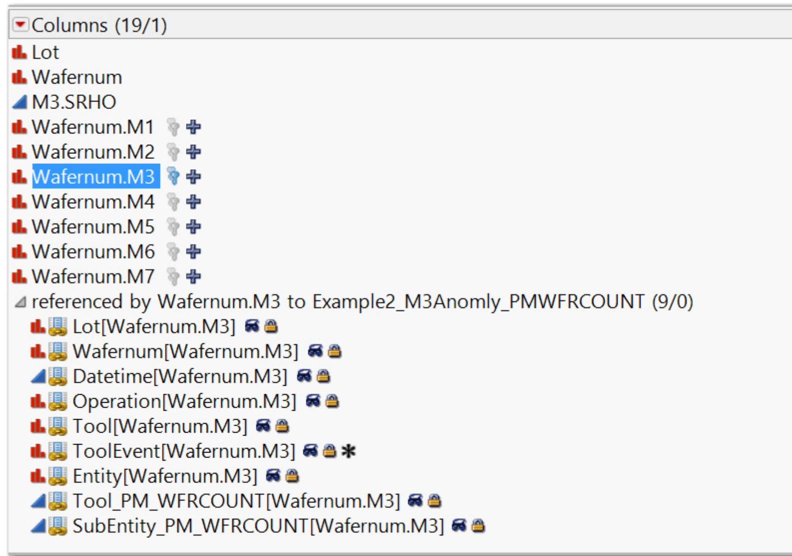


In the second RCA meeting, an analysis similar to Example 1 was run using the 7 files from metal deposition as well as other are tools. No strong or consistent pattern or sequence of tools was found to align with anomalous M3SRHO data. Each area also reviewed the result of their investigation of changes in their area's materials, methods or replacement parts. The metal sputter review highlighted there was a M5 chemistry change in mid-December, to make the coating more conformal to the IC architecture. The chemicals were the same, just the mixture amount for M5 changed slightly. The M5 change required a characterization to be done before full implementation. The test included each layer being run after the new M5 chemistry. No effect was seen. However, only a few wafers for each combination was run. Given this information and the fact the problem is seem only on M3, the RCA team leader decides the next meeting will focus on the tool state when M3 is running. The sputter rep needs to bring wafer count on the M3 sputter tools, how many total wafers were run. It was agreed that the engineer can have an extra day to get the data together the number of M5 wafers run before any M3 lot.

At the third RCA meeting, the anomaly data was opened. Even though files M1-M7, did not show anything useful, the Link References were not deleted. The other are file links were removed. The JMP file Example2_M3Anomaly_PMWFRCOUNT was opened. The engineer had already created the Link ID. Right click on the column Wafernum.M3 and select Link Reference. The PM data showed up as an option and was selected resulting in the file depicted in Figure 13.

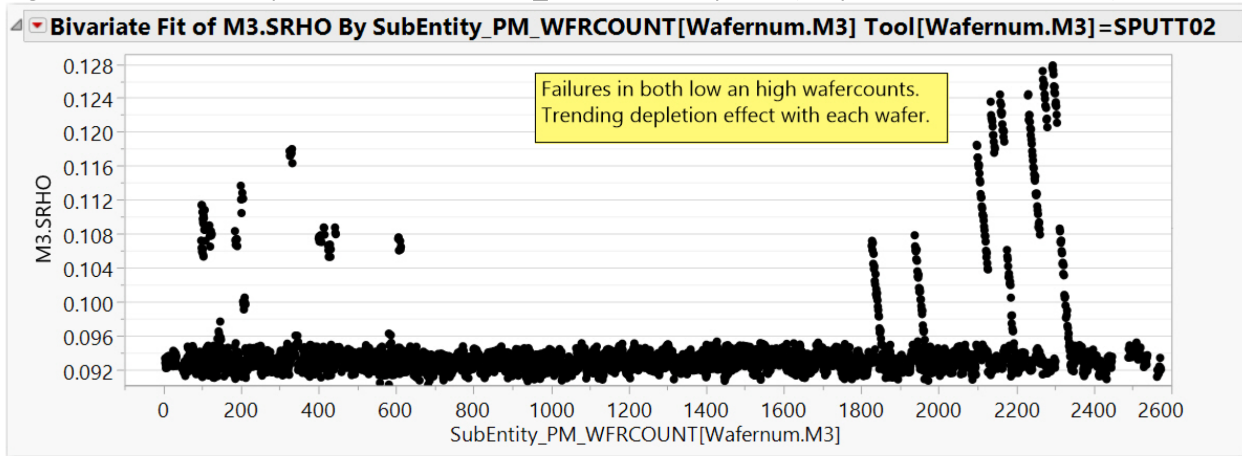
Fit Y by X was chosen for the analyses, using M3.SRHO for Y and the column Subentity_PM_Wfrcount from the table Example2_M3Anomaly_PMWFRCOUNT. See Figure 14.

Figure 13 Post-Holiday EOL M3column Anomaly Data Linked with M3 Tool PMCount



When there are intermittent failures, it is my experience to do less aggregation. If the Lot Avg versus wafer count at lot start had been plotted in Figure 14, the failures in both low and high wafers counts would be visible, however, the wafer dissipation (depletion) effect would not have been seen.

Figure 14 Post-Holiday M3.SRHO versus PM_WFRCOUNT by Tool (only SPUTT02 shown here)



The logical next step is to add the M5 wafer counts to a file and continue exploring. However, typically, this would be the time and place where the truly brilliant area engineers often propose excellent physical and chemical models of what might be happening. The proposed cause is a contamination effect that only affects M3 and it is reasonable to expect a M5 contamination of M3.

For past analyses, I have used event analyses for my next steps. (1)Create a table of all tool events. (2)Create columns representing 10 or more lags. This is easier to do by tool, create the lags, then join the tables. (3)Select the M3 operation data, since the investigation is on potential sources that affect M3. Once the tool state data is joined with the M3.SRHO data use tree methods (Partition) and feature selection to find a pattern. If this last paragraph is difficult to understand, it's best to look away and stare at Figure 15 for a few minutes.

Example 1 and the first iteration analysis of Example 2 looked for the sequence of the wafers, that is, wafer history. This analysis is different, it is looking at the tool history, what is the state of the tool when M3 was run. This last statement seems to be confusing to both IC manufacturing novices and experienced people. To help explain this difference, consider the example of a drug reaction seen on a small subset of people. The first analysis might be to look at the drug itself, where it was manufactured, how was it handled, how it was administered. And if nothing was found, then assume the drug was fine, but what was the state or characteristics of the person (ex. the reaction only occurs when eating both dairy and red licorice within an hour of the medication).

Event sequencing can be done, fairly quickly using the JMP function Lag(:col, n), which works on character columns.

Figure 15 displays a snippet of the table structure and Figure 16 displays the Partition history. Lags 1, 2 and 3 were chosen in order, and each time, splitting on the event M5LotMoveIn. So similar to the red licorice scenario, M3 has a bad reaction when M5 is run on the tool a few runs prior to M3.

Figure 15 Snippet of table M3 Last 20.jmp Structure with lags of Tool Events

Tool	ToolEvent	Entity	Event Lag 1	Event Lag 2	Event Lag 3	Event Lag 4	Event Lag 5	Patt5	num5
SPUTT01	M3LotMoveIn	SPUTT01	M1LotMoveIn	M4LotMoveIn	M6LotMoveIn	M5LotMoveIn	M2LotMoveIn	11151	11151
SPUTT01	M3LotMoveIn	SPUTT01	M4LotMoveIn	M6LotMoveIn	M2LotMoveIn	M6LotMoveIn	M4LotMoveIn	11111	11111
SPUTT01	M3LotMoveIn	SPUTT01	M5LotMoveIn	M6LotMoveIn	M2LotMoveIn	M7LotMoveIn	M5LotMoveIn	51115	51115
SPUTT01	M3LotMoveIn	SPUTT01	M6LotMoveIn	M2LotMoveIn	M6LotMoveIn	M7LotMoveIn	M2LotMoveIn	11111	11111

Columns Patt5 and num5 were created in the linked table and automatically appear in the main table. They were created to simplify the explanation of the effects seen so far.

Figure 16 Partition of M3.SRHO Lot Average vs. Last 20 Tool Events

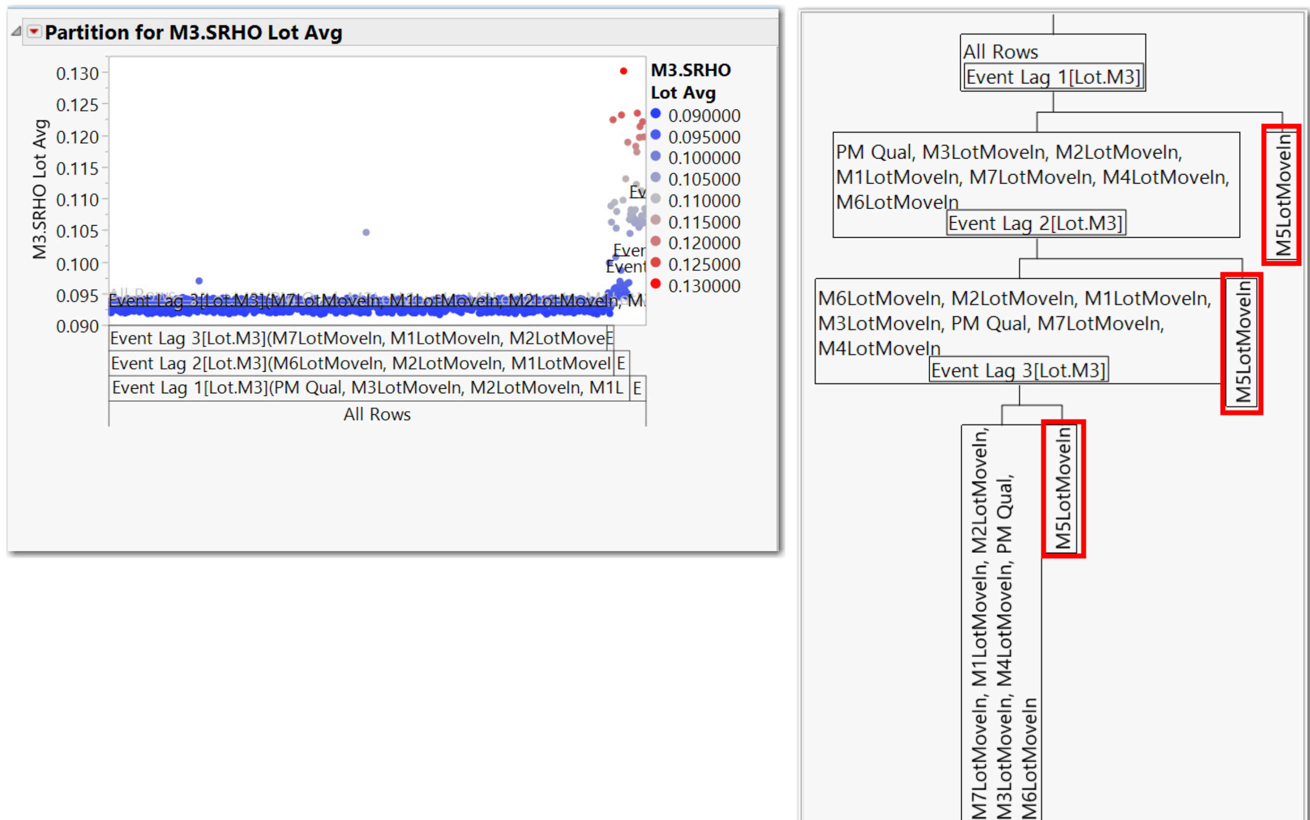
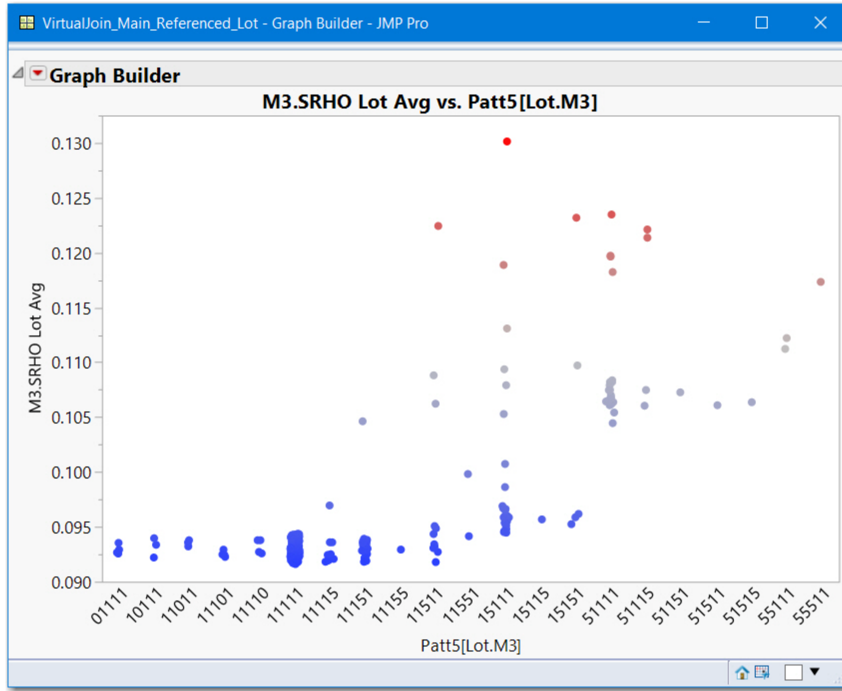


Figure 17 is a graph of this sequencing effect, where, a PM is coded as 0, any other step is coded as 1 and M5 as 5. Sometimes I refer to these as management graphs, however, a better name would be a

presentation graph. It tells the story quickly. (Note more context and better titles would be added before calling this a presentation graph. Here the default platform titles were maintained to indicate the JMP platform commands used to create it.)

Figure 17 Presentation Graph of M3SRHO Lot Processing Effect.



Next steps? When there are saturation and depletion effects, standard modeling is not very successful. To date, my BKM (best known method) is to estimate specific effects. For example, find all instances where M5 was run just prior to M3; calculate delta to baseline; then compare the effect to the number of M5 wafers processed. Also, study the size of the wafer-to-wafer depletion effect seen in Figure 14.

This simulation emulates a problem where M5 was contaminating the tool chambers. However, all but one process (M3) had a pre-conditioning step. In the associated zip file, the folder SimulatedData contains a readme.txt file that explains the simulated effects. Comment: the number of “bad” lots is much larger than what would likely be seen before a problem would likely be resolved. This lack of complete realism is due to this hand built queue that needed to run for a while to get a good mixture of M5 run prior and in close proximity to M3.

Summary

Root cause analyses are time consuming, require domain expertise and much data manipulation. Virtual Joins provide the opportunity for the domain experts to create tables and use their own naming conventions, and if the concept of the Virtual Join is understood and Link IDs created in the area or toolset specific files, cross domain data sets can ready for analyses much faster.

Sequential, conditional effects that can saturate, are “repairable” or interact with usage are not frequent. However, I am looking for a protocol (standard format) and methods to investigate and determine these types of effects. I am currently researching reliability sequencing models.

References

[1] Figure 1(L) Bare Silicon Wafer, (n.a.), (n.d.), **Silicon Wafer**. Retrieved July 31, 2017, from TechInstro website, https://www.techinstro.com/silicon-wafer/

[2] Figure 1 (L) , (n.a.), April 09, 2013, **Die Per Wafer Formula and (free) Calculator**. Retrieved July 31, 2017, from anysilicon website, http://anysilicon.com/die-per-wafer-formula-free-calculators/

[3] Figure 1(R) (n.a.), (n.d.), **Back End of Line**. Retrieved July 31, 2017 from Wikipedia, The Free Encyclopedia, website, https://en.wikipedia.org/wiki/Back_end_of_line

[4] Figure 2, 유광재, January 20, 2015, **Optimal Fab SDM LAB, KAIST Industrial & Systems Engineering**, Figure 2. Optimal FAB layout design based on the material flow among the process types. Retrieved July 31, 2017, from http://sdm.kaist.ac.kr/wordpress/korean/%EC%9C%A0%EA%B4%91%EC%9E%AC/, http://sdm.k aist.ac.kr/wordpress/korean/wp-content/uploads/sites/2/2015/01/YGJ_2.png.

[5] Leetaru, K. (June 12, 2016) ***Why We Need More Domain Experts In The Data Sciences.*** Retrieved July 31, 2017 from the Forbes website https://www.forbes.com/sites/kalevleetaru/2016/06/12/why-we-need-more-domain-experts-in-the-data-sciences/#14ee761e3b50

[6] (n.a.), (n.d.), ***Data Mining from A to Z.*** Retrieved July 31, 217 from the SAS website https://www.sas.com/content/dam/SAS/en_us/doc/whitepaper1/data-mining-from-a-z-104937.pdf

[7] Other websites to learn more about semiconductor manufacturing
https://www.bloomberg.com/news/articles/2016-06-09/how-intel-makes-a-chip
https://www.youtube.com/watch?v=vmAyXWvLHeI (FOUP Load Movie)
https://www.youtube.com/watch?v=4Q_n4vdyZzc (Semiconductor Technology at TSMC, 2011)