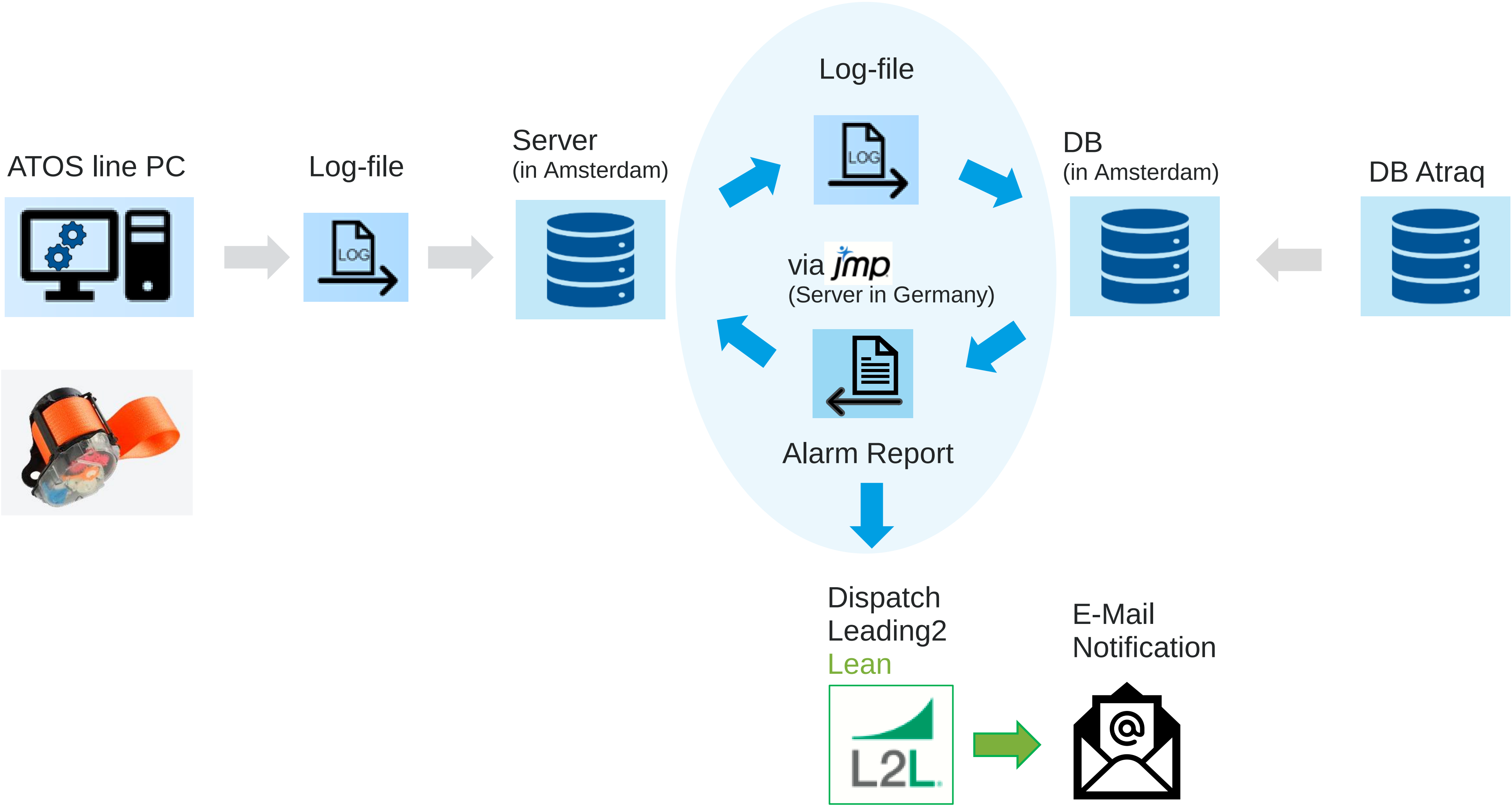
Automated Process of Collecting Product Test Data and Creating Alarm Reports for Root Cause Analysis

Dr. Astrid Ruck

Group Senior Specialist in Statistics

Autoliv

Content



ATOS line PC

Log-file

Server
(in Amsterdam)

Log-file

via jmp
(Server in Germany)

Alarm Report

DB
(in Amsterdam)

DB Atraq

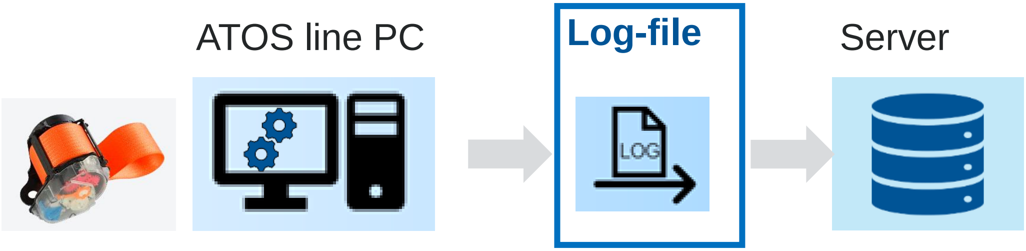
Dispatch Leading2 Lean

E-Mail Notification

Log-Files

Log-Files are text files, including **information** about …

- Machine ID
- Test Date/Time
- Seatbelt (full assembly)
- **Test Type**
  - **Tilt-Right, Tilt-Left, Tilt-Forward, Tilt-Backward [°]**
    To ensure blocking of seatbelt extraction in the case of rollover scenario
  - **Webbing Length [mm]**
    To ensure correct webbing length based on vehicle type, due to its dependency of blocking
  - **WebSenseLock [g]**
    To ensure blocking of webbing extraction at specific webbing acceleration
  - **WebSenseNoLock [g]**
    To ensure free wheeling of webbing extraction at low webbing acceleration
  - **KiSi (Kindersicherung/Child restrain system) [pass/fail]**
    To ensure that automatic locking retractor (ALR) is activated or not
- **Test Value** (continuous)
- **Result overall Tests** and **per Test (Pass/Fail)**



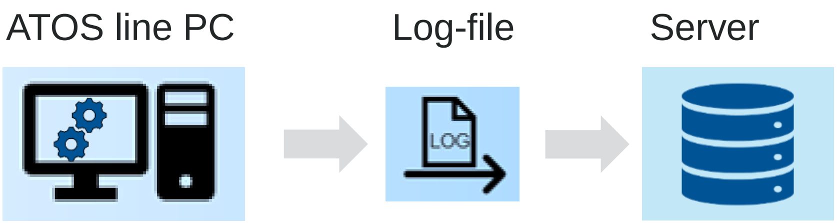
final_check_test_autoliv

- **Machine Parameter Settings** per Test Type
- **Internal Barcode** (unique per retractor) based on
  - Retractor Number
  - Global Line ID for Retractor Line
  - Production day of Retractor
  - Key Index
- Product Family
- Confection Line
- Customer
- Car Position

Autoliv

Automated Log-File Transfer to a Server

- **Log-files** are transferred **once per day to a server.**
- In order **not to disturb the testing of products**, the log-files are transmitted …
  - **between night** and **morning shift,** if the product tests are performed in a **three-shift system**
  - **after** the **early shift,** if the product tests are performed in a **two-shift system**
- These **transfer times** are **synchronized within an Autoliv facility**, but are **different between Autoliv's facilities**, due to **different time zones**
  - **Examples** based on Berlin-Time
    - **06:05 Autoliv Hungary (ALH)** after night shift (3 shifts)
    - **07:15 Autoliv Rumania (ARO)** after night shift (3 shifts)
    - **….**



Script transfers log-files from ATOS line PCs to a server

Folder structure on the server:

- Plant (ALH, ARO, …)
  - Machine Number (FCS-M-xxxxxxx)
    - YearMonth (YYYYMM)

    → Daily log-file

Automated Log-File Transfer into a SQL-DB via JMP 16.1.0

- **Multiple File Import** (available since JMP15)
- **Add/create relevant Columns**
- **Select relevant Columns**
  - via column name, which is **independent of the column order**
  - **delete irrelevant columns**
- **Recode** – clean data

```
dt << Begin Data Update;
dt << Recode Column(:ProductFamily,{Map Value( _rcOrig, {" Rxxx", "Rxxx"},
Unmatched( _rcNow ) )},Target Column( :ProductFamily ));
dt << End Data Update;
```

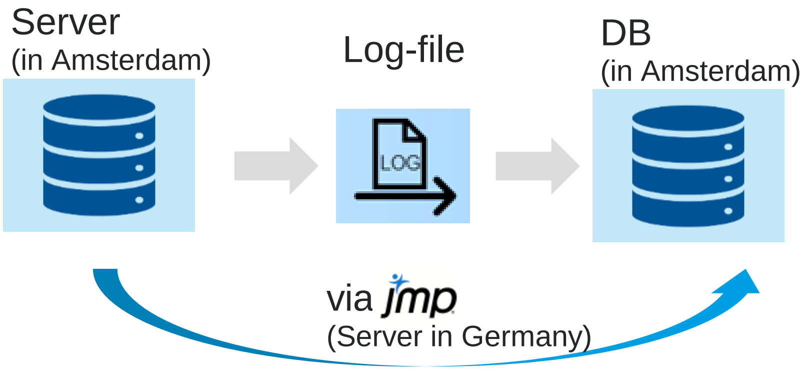
- **Transfer into DB**

```
   New SQL Query(Connection( connection_string ), CustomSQL(sql))

   <<  Run Foreground();
```

  - **Run Foreground()** ensures, that the transfer into the database will be completed before the next procedures will run

- **Avoid background processes …**

```
      ▪  Close All( Data Tables, NoSave );
      ▪  Exit(No Save)
```



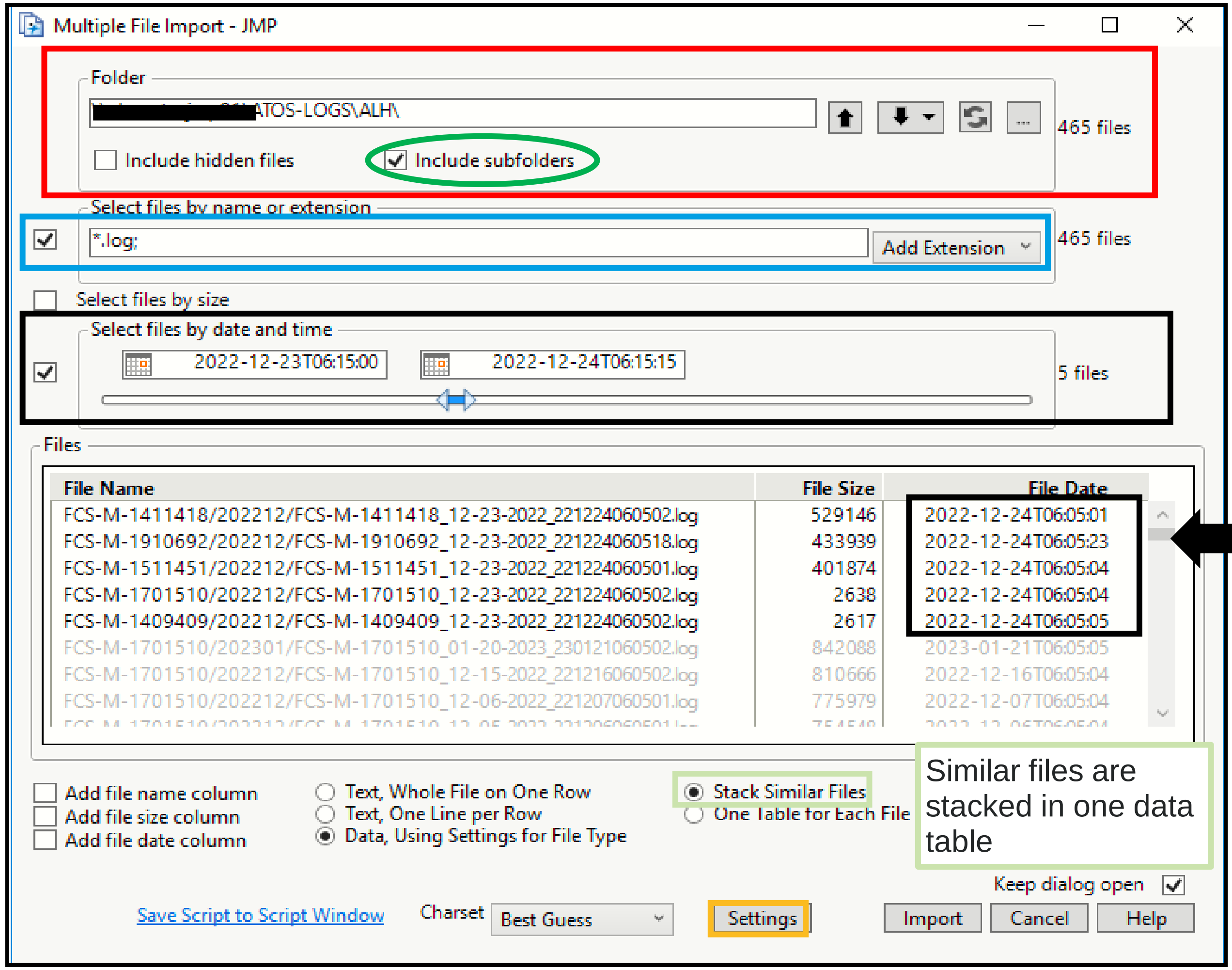
Since **JMP16, each action** in JMP is **recorded** in the **Enhanced Log**. Data work can now be saved as a **playable script per point-and-click.**

“JMP writes 90% of your code – the skeleton, …”

Jordan Hiller, JMP Senior Systems Engineer

=> the other 10% is learning by doing

Multiple File Import



Multiple File Import - JMP

Folder

[redacted]ATOS-LOGS\ALH\

465 files

□ Include hidden files ☑ Include subfolders

Select files by name or extension

☑ *.log; Add Extension

465 files

□ Select files by size

☑ Select files by date and time

2022-12-23T06:15:00 2022-12-24T06:15:15

5 files

Files

File Name	File Size	File Date
FCS-M-1411418/202212/FCS-M-1411418_12-23-2022_221224060502.log	529146	2022-12-24T06:05:01
FCS-M-1910692/202212/FCS-M-1910692_12-23-2022_221224060518.log	433939	2022-12-24T06:05:23
FCS-M-1511451/202212/FCS-M-1511451_12-23-2022_221224060501.log	401874	2022-12-24T06:05:04
FCS-M-1701510/202212/FCS-M-1701510_12-23-2022_221224060502.log	2638	2022-12-24T06:05:04
FCS-M-1409409/202212/FCS-M-1409409_12-23-2022_221224060502.log	2617	2022-12-24T06:05:05
FCS-M-1701510/202301/FCS-M-1701510_01-20-2023_230121060502.log	842088	2023-01-21T06:05:05
FCS-M-1701510/202212/FCS-M-1701510_12-15-2022_221216060502.log	810666	2022-12-16T06:05:04
FCS-M-1701510/202212/FCS-M-1701510_12-06-2022_221207060501.log	775979	2022-12-07T06:05:04

□ Add file name column
□ Add file size column
□ Add file date column

○ Text, Whole File on One Row
○ Text, One Line per Row
● Data, Using Settings for File Type

● Stack Similar Files
○ One Table for Each File

Similar files are stacked in one data table

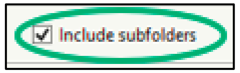
Keep dialog open ☑

Folder structure on the server:

- Plant (ALH, ARO, …)
  - Machine Number (FCS-M-xxxxxxx)
    - YearMonth (YYYYMM)
    - → Daily log-file

Server
(in Amsterdam)

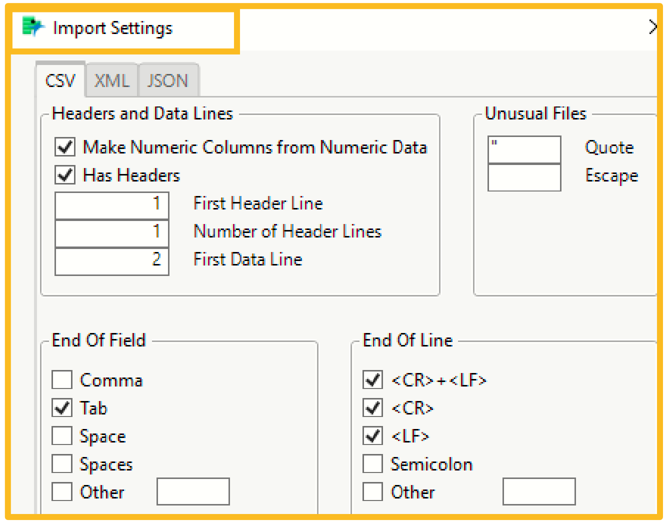
Folder: \\server name\ATOS-LOGS\Plant



☑ Include subfolders imports all files from subdirectories

At 2022-12-24 the log-files from 2022-12-23 6:15:00 to 2022-12-24 06:15 shall be imported into JMP

ALH transfers log files at 6:05 after night shift



Import Settings

CSV XML JSON

Headers and Data Lines

☑ Make Numeric Columns from Numeric Data
☑ Has Headers
1 First Header Line
1 Number of Header Lines
2 First Data Line

Unusual Files

" Quote
Escape

End Of Field

□ Comma
☑ Tab
□ Space
□ Spaces
□ Other

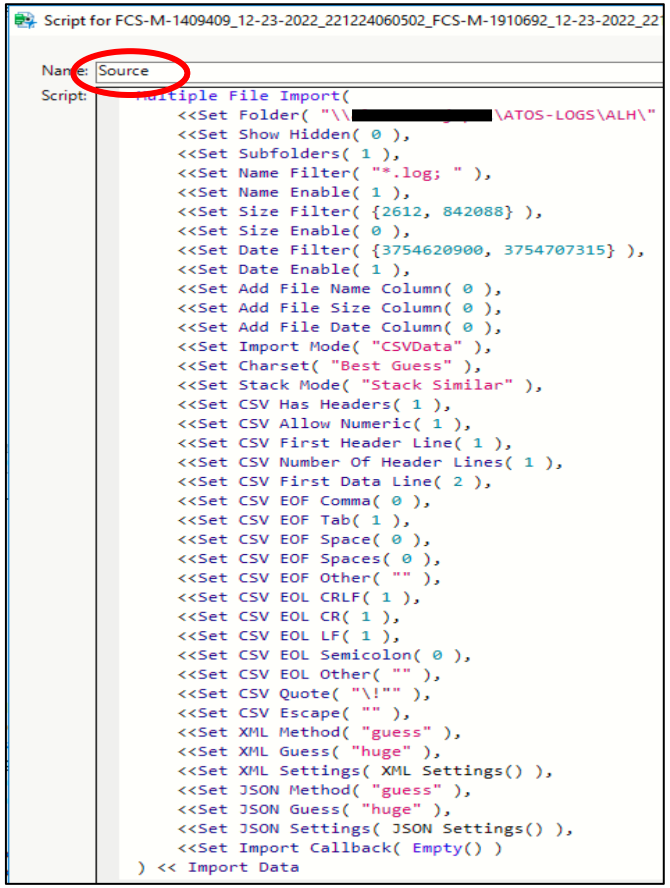
End Of Line

☑ <CR>+<LF>
☑ <CR>
☑ <LF>
□ Semicolon
□ Other

CSV settings can be changed manually

Autoliv

Multiple File Import: Resulting Data Table and Playable Scripts

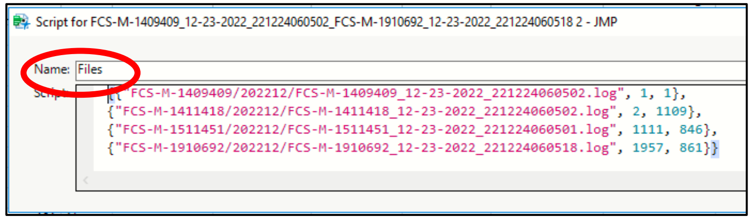


Script for FCS-M-1409409_12-23-2022_221224060502_FCS-M-1910692_12-23-2022_22

Name: Source

Script:

```
Multiple File Import(
      <<Set Folder( "\\[redacted]\ATOS-LOGS\ALH\" ),
      <<Set Show Hidden( 0 ),
      <<Set Subfolders( 1 ),
      <<Set Name Filter( "*.log; " ),
      <<Set Name Enable( 1 ),
      <<Set Size Filter( {2612, 842088} ),
      <<Set Size Enable( 0 ),
      <<Set Date Filter( {3754620900, 3754707315} ),
      <<Set Date Enable( 1 ),
      <<Set Add File Name Column( 0 ),
      <<Set Add File Size Column( 0 ),
      <<Set Add File Date Column( 0 ),
      <<Set Import Mode( "CSVData" ),
      <<Set Charset( "Best Guess" ),
      <<Set Stack Mode( "Stack Similar" ),
      <<Set CSV Has Headers( 1 ),
      <<Set CSV Allow Numeric( 1 ),
      <<Set CSV First Header Line( 1 ),
      <<Set CSV Number Of Header Lines( 1 ),
      <<Set CSV First Data Line( 2 ),
      <<Set CSV EOF Comma( 0 ),
      <<Set CSV EOF Tab( 1 ),
      <<Set CSV EOF Space( 0 ),
      <<Set CSV EOF Spaces( 0 ),
      <<Set CSV EOF Other( "" ),
      <<Set CSV EOL CRLF( 1 ),
      <<Set CSV EOL CR( 1 ),
      <<Set CSV EOL LF( 1 ),
      <<Set CSV EOL Semicolon( 0 ),
      <<Set CSV EOL Other( "" ),
      <<Set CSV Quote( "\!"" ),
      <<Set CSV Escape( "" ),
      <<Set XML Method( "guess" ),
      <<Set XML Guess( "huge" ),
      <<Set XML Settings( XML Settings() ),
      <<Set JSON Method( "guess" ),
      <<Set JSON Guess( "huge" ),
      <<Set JSON Settings( JSON Settings() ),
      <<Set Import Callback( Empty() )
) << Import Data
```

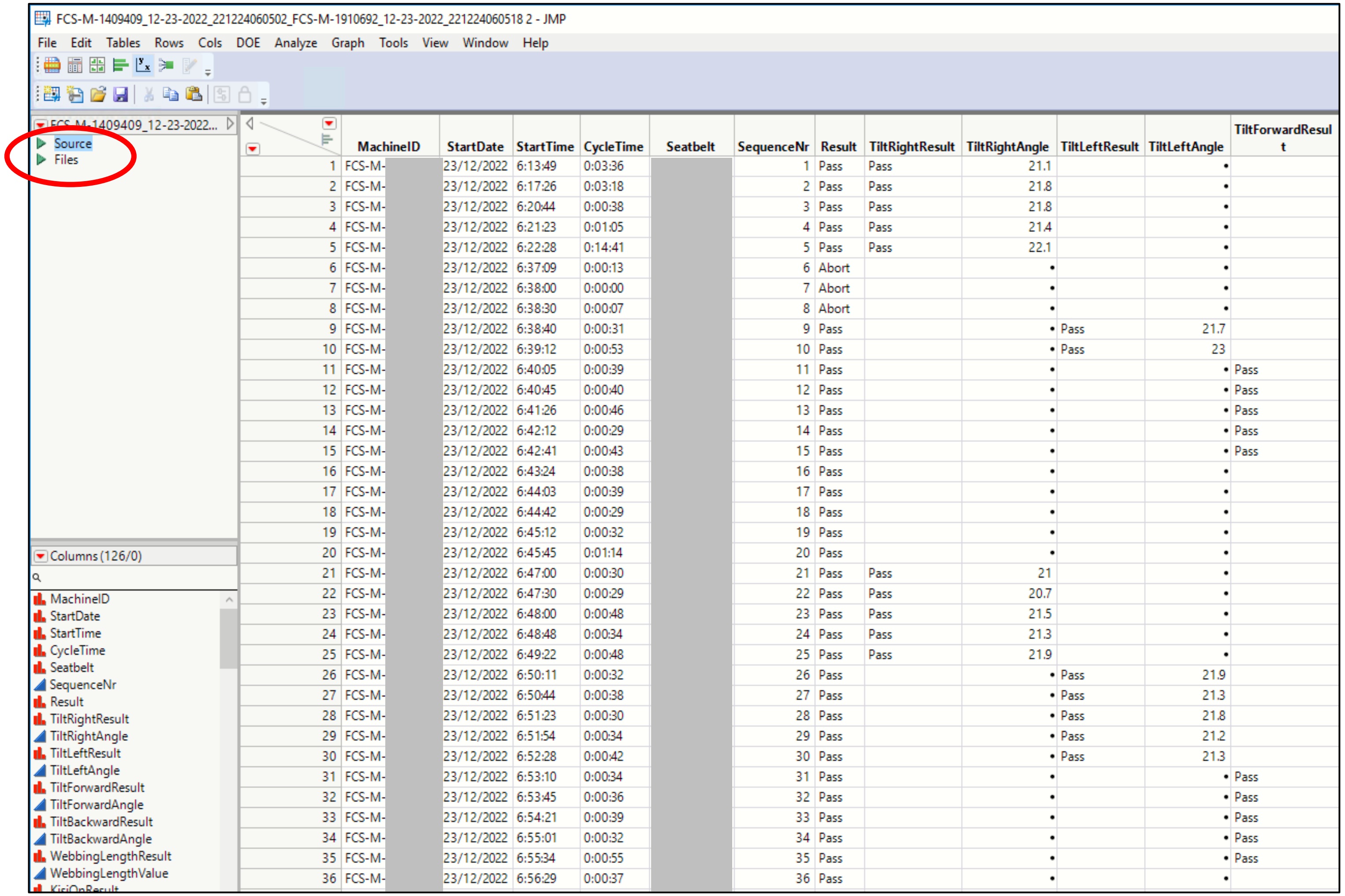


Script for FCS-M-1409409_12-23-2022_221224060502_FCS-M-1910692_12-23-2022_221224060518 2 - JMP

Name: Files

Script:

```
{{"FCS-M-1409409/202212/FCS-M-1409409_12-23-2022_221224060502.log", 1, 1},
{"FCS-M-1411418/202212/FCS-M-1411418_12-23-2022_221224060502.log", 2, 1109},
{"FCS-M-1511451/202212/FCS-M-1511451_12-23-2022_221224060501.log", 1111, 846},
{"FCS-M-1910692/202212/FCS-M-1910692_12-23-2022_221224060518.log", 1957, 861}}
```



FCS-M-1409409_12-23-2022_221224060502_FCS-M-1910692_12-23-2022_221224060518 2 - JMP

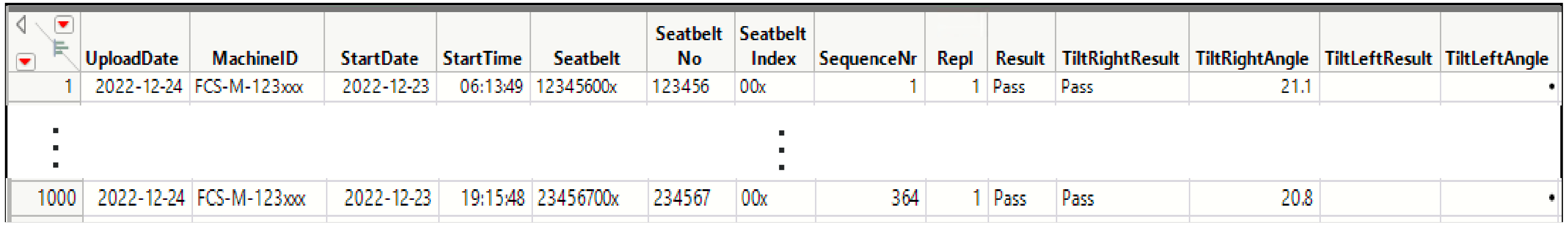
FCS-M-1409409_12-23-2022...
- Source
- Files

Columns (126/0)

- MachineID
- StartDate
- StartTime
- CycleTime
- Seatbelt
- SequenceNr
- Result
- TiltRightResult
- TiltRightAngle
- TiltLeftResult
- TiltLeftAngle
- TiltForwardResult
- TiltForwardAngle
- TiltBackwardResult
- TiltBackwardAngle
- WebbingLengthResult
- WebbingLengthValue
- KisiOnResult

	MachineID	StartDate	StartTime	CycleTime	Seatbelt	SequenceNr	Result	TiltRightResult	TiltRightAngle	TiltLeftResult	TiltLeftAngle	TiltForwardResult
1	FCS-M-[redacted]	23/12/2022	6:13:49	0:03:36	[redacted]	1	Pass	Pass	21.1		•	
2	FCS-M-[redacted]	23/12/2022	6:17:26	0:03:18	[redacted]	2	Pass	Pass	21.8		•	
3	FCS-M-[redacted]	23/12/2022	6:20:44	0:00:38	[redacted]	3	Pass	Pass	21.8		•	
4	FCS-M-[redacted]	23/12/2022	6:21:23	0:01:05	[redacted]	4	Pass	Pass	21.4		•	
5	FCS-M-[redacted]	23/12/2022	6:22:28	0:14:41	[redacted]	5	Pass	Pass	22.1		•	
6	FCS-M-[redacted]	23/12/2022	6:37:09	0:00:13	[redacted]	6	Abort		•		•	
7	FCS-M-[redacted]	23/12/2022	6:38:00	0:00:00	[redacted]	7	Abort		•		•	
8	FCS-M-[redacted]	23/12/2022	6:38:30	0:00:07	[redacted]	8	Abort		•		•	
9	FCS-M-[redacted]	23/12/2022	6:38:40	0:00:31	[redacted]	9	Pass		•	Pass	21.7	
10	FCS-M-[redacted]	23/12/2022	6:39:12	0:00:53	[redacted]	10	Pass		•	Pass	23	
11	FCS-M-[redacted]	23/12/2022	6:40:05	0:00:39	[redacted]	11	Pass		•		•	Pass
12	FCS-M-[redacted]	23/12/2022	6:40:45	0:00:40	[redacted]	12	Pass		•		•	Pass
13	FCS-M-[redacted]	23/12/2022	6:41:26	0:00:46	[redacted]	13	Pass		•		•	Pass
14	FCS-M-[redacted]	23/12/2022	6:42:12	0:00:29	[redacted]	14	Pass		•		•	Pass
15	FCS-M-[redacted]	23/12/2022	6:42:41	0:00:43	[redacted]	15	Pass		•		•	Pass
16	FCS-M-[redacted]	23/12/2022	6:43:24	0:00:38	[redacted]	16	Pass		•		•	
17	FCS-M-[redacted]	23/12/2022	6:44:03	0:00:39	[redacted]	17	Pass		•		•	
18	FCS-M-[redacted]	23/12/2022	6:44:42	0:00:29	[redacted]	18	Pass		•		•	
19	FCS-M-[redacted]	23/12/2022	6:45:12	0:00:32	[redacted]	19	Pass		•		•	
20	FCS-M-[redacted]	23/12/2022	6:45:45	0:01:14	[redacted]	20	Pass		•		•	
21	FCS-M-[redacted]	23/12/2022	6:47:00	0:00:30	[redacted]	21	Pass	Pass	21		•	
22	FCS-M-[redacted]	23/12/2022	6:47:30	0:00:29	[redacted]	22	Pass	Pass	20.7		•	
23	FCS-M-[redacted]	23/12/2022	6:48:00	0:00:48	[redacted]	23	Pass	Pass	21.5		•	
24	FCS-M-[redacted]	23/12/2022	6:48:48	0:00:34	[redacted]	24	Pass	Pass	21.3		•	
25	FCS-M-[redacted]	23/12/2022	6:49:22	0:00:48	[redacted]	25	Pass	Pass	21.9		•	
26	FCS-M-[redacted]	23/12/2022	6:50:11	0:00:32	[redacted]	26	Pass		•	Pass	21.9	
27	FCS-M-[redacted]	23/12/2022	6:50:44	0:00:38	[redacted]	27	Pass		•	Pass	21.3	
28	FCS-M-[redacted]	23/12/2022	6:51:23	0:00:30	[redacted]	28	Pass		•	Pass	21.8	
29	FCS-M-[redacted]	23/12/2022	6:51:54	0:00:34	[redacted]	29	Pass		•	Pass	21.2	
30	FCS-M-[redacted]	23/12/2022	6:52:28	0:00:42	[redacted]	30	Pass		•	Pass	21.3	
31	FCS-M-[redacted]	23/12/2022	6:53:10	0:00:34	[redacted]	31	Pass		•		•	Pass
32	FCS-M-[redacted]	23/12/2022	6:53:45	0:00:36	[redacted]	32	Pass		•		•	Pass
33	FCS-M-[redacted]	23/12/2022	6:54:21	0:00:39	[redacted]	33	Pass		•		•	Pass
34	FCS-M-[redacted]	23/12/2022	6:55:01	0:00:32	[redacted]	34	Pass		•		•	Pass
35	FCS-M-[redacted]	23/12/2022	6:55:34	0:00:55	[redacted]	35	Pass		•		•	Pass
36	FCS-M-[redacted]	23/12/2022	6:56:29	0:00:37	[redacted]	36	Pass		•		•	

Creating a Value List for a SQL Command



	UploadDate	MachineID	StartDate	StartTime	Seatbelt	Seatbelt No	Seatbelt Index	SequenceNr	Repl	Result	TiltRightResult	TiltRightAngle	TiltLeftResult	TiltLeftAngle
1	2022-12-24	FCS-M-123xxx	2022-12-23	06:13:49	12345600x	123456	00x	1	1	Pass	Pass	21.1		•
⋮						⋮								
1000	2022-12-24	FCS-M-123xxx	2022-12-23	19:15:48	23456700x	234567	00x	364	1	Pass	Pass	20.8		•

…

…

We transfer **1000 rows** of the data table **per loop** into the database

```
r = dt << Get Rows Where(Row() >= 1 & Row() <= 1000);
    MyList = dt[r,0];
```

MyList in JMP

Empty character cell

```
{{"2022-12-24", "FCS-M-123xxx", "2022-12-23", "06:13:49", “12345600x", “123456", "00x",   1, 1, "Pass", "Pass", 21.1, "", ., …},…,

 {"2022-12-24", "FCS-M-123xxx", "2022-12-23", “19:15:48", “23456700x", “234567", "00x", 364, 1, "Pass", "Pass", 20.8, "", ., …}}
```

Empty numeric cell

Tip: Use Data Type “Character” for Date/Time Columns

Using “Substitute” to create values for a SQL command

```
x=Substitute(Char(MyList), Items(  "\!"", "" ), "'");
x=Remove(x,1);
x=Remove(x,Length(x));
x=Substitute(x, Items(  "\{","" ), "(" );
x=Substitute(x, Items(  "\}","" ), ")" );
x=Substitute(x, ".," , " NULL,")
```

```
" -> '
Remove   1st list element: {
Remove last list element: }
{ -> (
} -> )
. -> NULL
```

```
"('2022-12-24', 'FCS-M-123xxx', '2022-12-23', ‘06:13:49', ‘12345600x', ‘123456', ‘00x’,   1, 1, 'Pass', 'Pass’, 21.1, '',  NULL, …),…,
 ⋮
 ('2022-12-24', 'FCS-M-123xxx', '2022-12-23', ‘19:15:48', ‘23456700x', ‘234567', ‘00x’, 364, 1, 'Pass', 'Pass’, 20.8, '',  NULL, …) "
```

Transfer into DB via CustomSQL Function in JMP

DB
(in Amsterdam)

“x” is the value list …

```
"('2022-12-24', 'FCS-M-123xxx', '2022-12-23', ‘06:13:49', ‘12345600x', ‘123456', ‘00x’,   1, 1, 'Pass', 'Pass’, 21.1, '',  NULL, …),…,

 ('2022-12-24', 'FCS-M-123xxx', '2022-12-23', ‘19:15:48', ‘23456700x', ‘234567', ‘00x’, 364, 1, 'Pass', 'Pass’, 20.8, '',  NULL, …) "
```

The **SQL-Command**, which is **used in CustomSQL is a plain string**, that goes directly into the database

```
sqlTemplate ="INSERT INTO [ATOS-Results].[dbo].[SPC]
         ([UploadDate],[MachineID],[StartDate],[StartTime],[Seatbelt],[SeatbeltNo],[SeatbeltIndex],
          [SequenceNr],[Repl],[Result],[TiltRightResult],[TiltRightAngle],[TiltLeftResult],
          [TiltLeftAngle], …)
     VALUES table;";
sql = Substitute(sqlTemplate, "table",x);
New SQL Query(Connection( connection_string ), CustomSQL(sql)) <<  Run Foreground();
```

Column names in DB table

Show(sql)

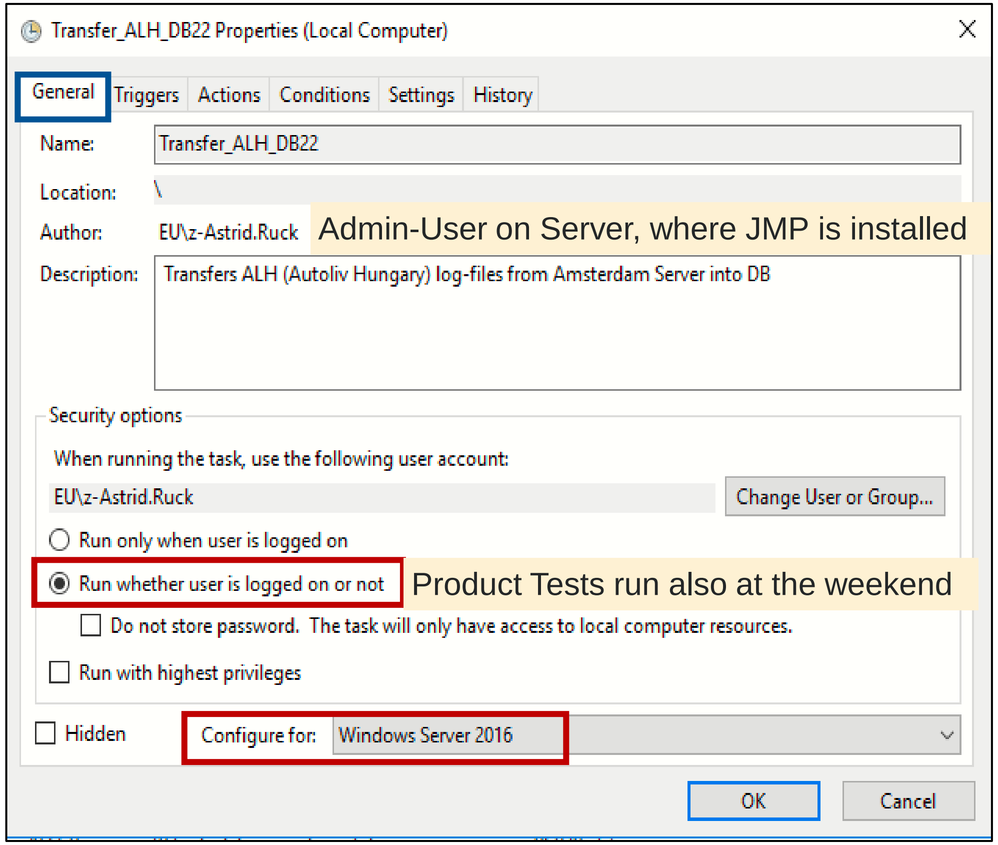
```
"INSERT INTO [ATOS-Results].[dbo].[SPC]
([UploadDate],[MachineID],[StartDate],[StartTime],[Seatbelt],[SeatbeltNo],[SeatbeltIndex],[SequenceNr],[Repl],[Result],[TiltRightResult],
 [TiltRightAngle],[TiltLeftResult],[TiltLeftAngle], …)
VALUES
('2022-12-24', 'FCS-M-123xxx', '2022-12-23', ‘06:13:49', ‘12345600x', ‘123456', ‘00x’,      1, 1, 'Pass', 'Pass’, 21.1, ‘ ', NULL,…),

('2022-12-24', 'FCS-M-123xxx', '2022-12-23', ‘19:15:48', ‘23456700x', ‘234567', ‘00x’,  364, 1, 'Pass', 'Pass’, 20.8, ‘ ',  NULL);"
```

Autoliv

Automatization of Data Transfer via Task Scheduler

DB
(in Amsterdam)

General



Transfer_ALH_DB22 Properties (Local Computer)

Name: Transfer_ALH_DB22

Location: \

Author: EU\z-Astrid.Ruck

Admin-User on Server, where JMP is installed

Description: Transfers ALH (Autoliv Hungary) log-files from Amsterdam Server into DB

Security options

When running the task, use the following user account:

EU\z-Astrid.Ruck

○ Run only when user is logged on

● Run whether user is logged on or not

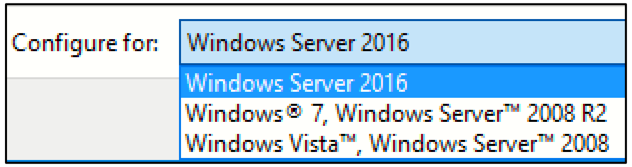
Product Tests run also at the weekend

□ Do not store password. The task will only have access to local computer resources.

□ Run with highest privileges

□ Hidden

Configure for: Windows Server 2016

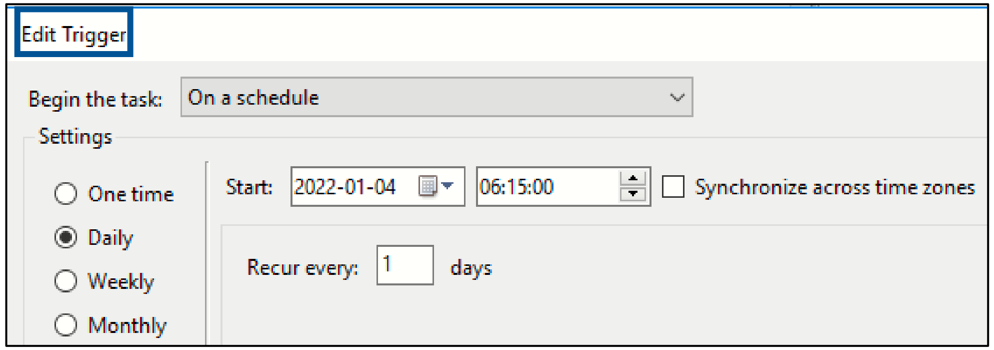


Configure for: Windows Server 2016

Windows Server 2016
Windows® 7, Windows Server™ 2008 R2
Windows Vista™, Windows Server™ 2008

If the configuration does not fit to the server, then background processes may not stop after completed task

Trigger



Edit Trigger

Begin the task: On a schedule

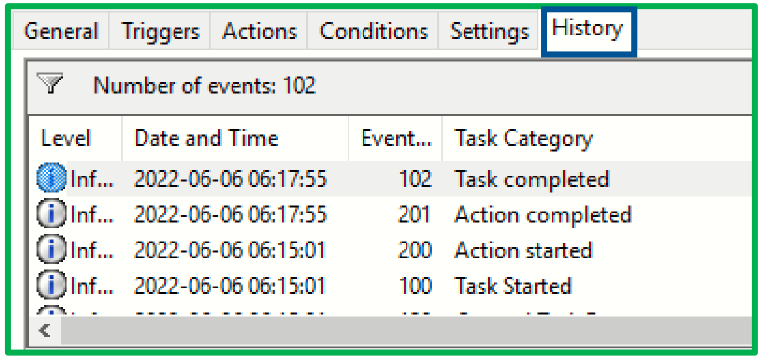
Settings

○ One time
● Daily
○ Weekly
○ Monthly

Start: 2022-01-04 06:15:00 □ Synchronize across time zones

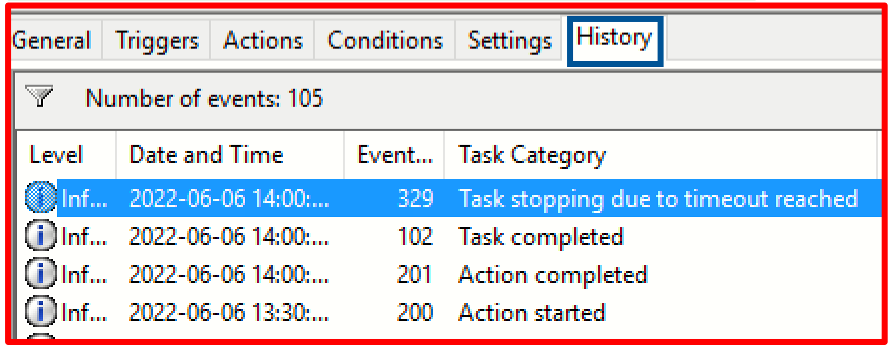
Recur every: 1 days

History



Number of events: 102

Level	Date and Time	Event...	Task Category
Inf...	2022-06-06 06:17:55	102	Task completed
Inf...	2022-06-06 06:17:55	201	Action completed
Inf...	2022-06-06 06:15:01	200	Action started
Inf...	2022-06-06 06:15:01	100	Task Started



Number of events: 105

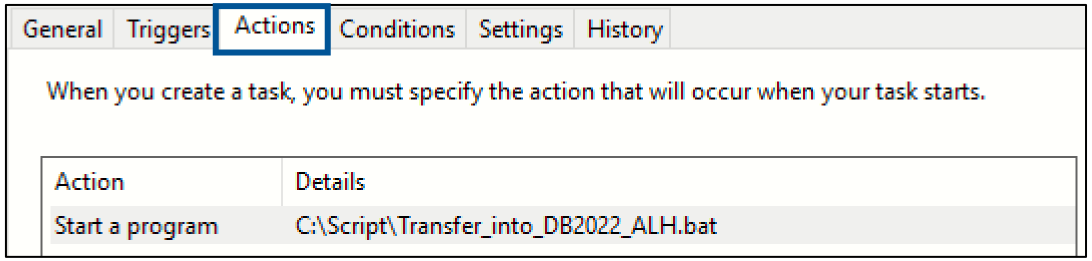
Level	Date and Time	Event...	Task Category
Inf...	2022-06-06 14:00:...	329	Task stopping due to timeout reached
Inf...	2022-06-06 14:00:...	102	Task completed
Inf...	2022-06-06 14:00:...	201	Action completed
Inf...	2022-06-06 13:30:...	200	Action started

To avoid background processes, check, if tables and programs are closed via script

Autoliv

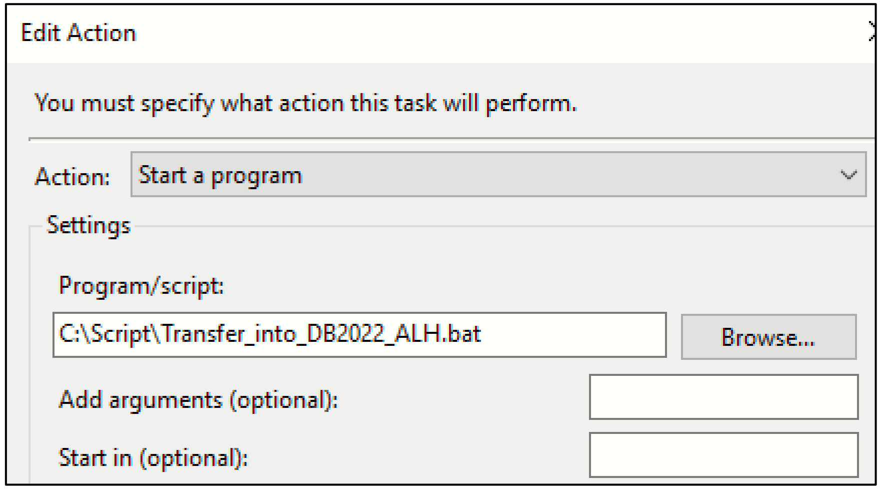
Automatization of Data Transfer via Task Scheduler

Action



When you create a task, you must specify the action that will occur when your task starts.

Action	Details
Start a program	C:\Script\Transfer_into_DB2022_ALH.bat



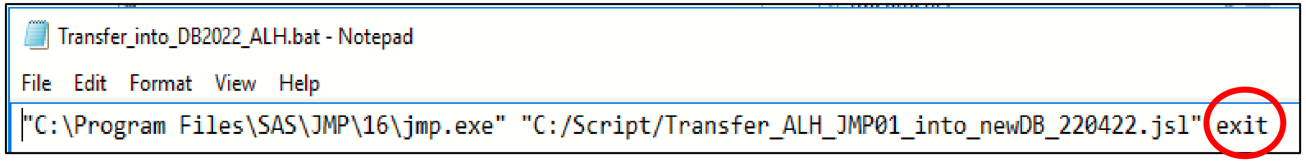
You must specify what action this task will perform.

Action: Start a program

Program/script: C:\Script\Transfer_into_DB2022_ALH.bat

Add arguments (optional):

Start in (optional):

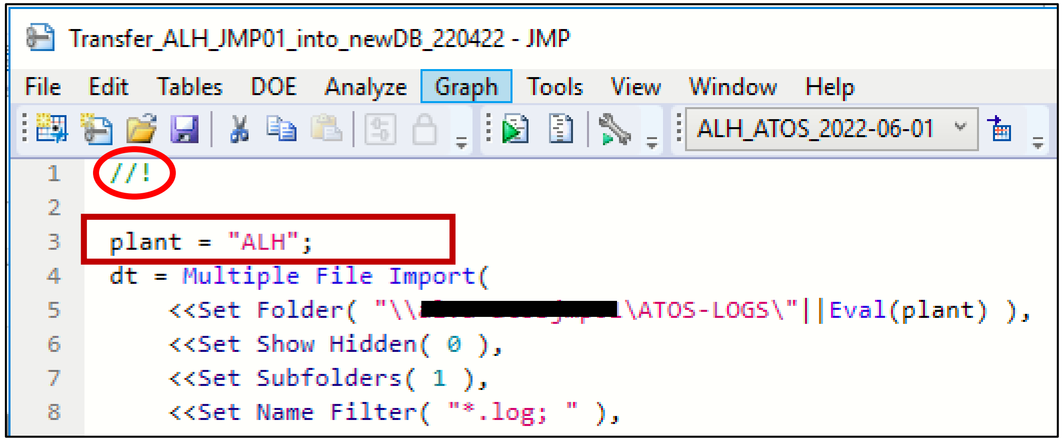


```
"C:\Program Files\SAS\JMP\16\jmp.exe" "C:/Script/Transfer_ALH_JMP01_into_newDB_220422.jsl" exit
```

Don't forget to close the Batch with "exit"

Advanced Note: Auto-Submit

If you want a **script to be executed via batch** instead of opened into the script window, then put the **command //! on the *first* line**



```
//!

plant = "ALH";
dt = Multiple File Import(
        <<Set Folder( "\\[redacted]\ATOS-LOGS\"||Eval(plant) ),
        <<Set Show Hidden( 0 ),
        <<Set Subfolders( 1 ),
        <<Set Name Filter( "*.log; " ),
```

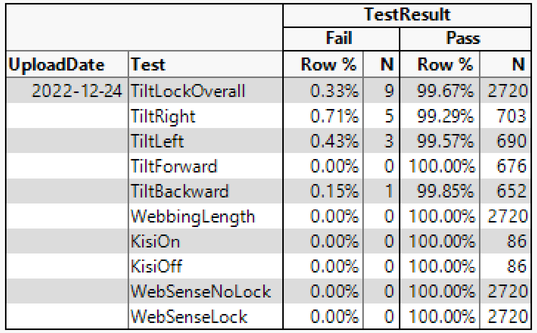
- To use **one algorithm for all plants**, we change the value of the global variable "plant" only.
- Thus, we can start the data transfer individually per plant

Examples based on Berlin-Time

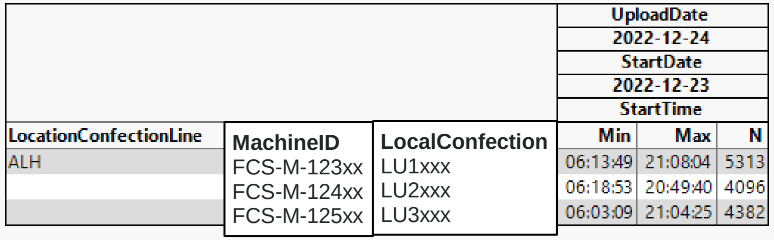
- **06:05 Autoliv Hungary (ALH)** after night shift (3 shifts)
- **07:15 Autoliv Rumania (ARO)** after night shift (3 shifts)

Autoliv

Creating Daily Reports overall Machines with JMP



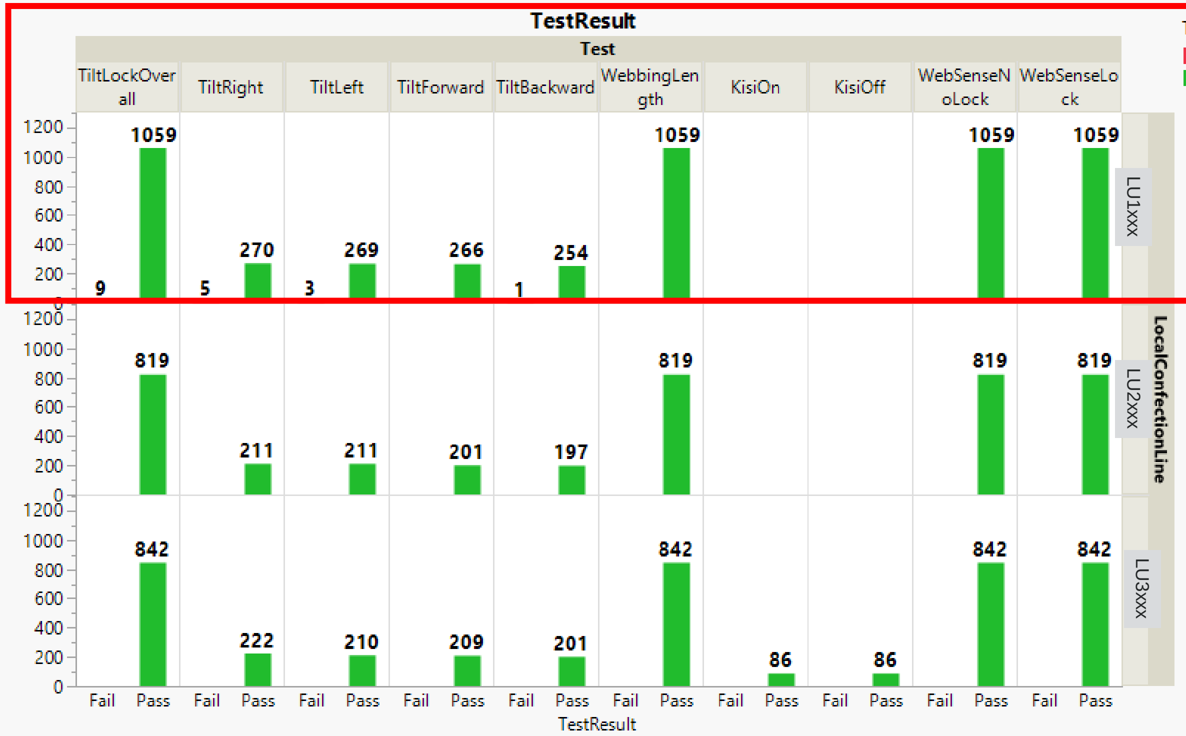
		TestResult			
		Fail		Pass	
UploadDate	Test	Row %	N	Row %	N
2022-12-24	TiltLockOverall	0.33%	9	99.67%	2720
	TiltRight	0.71%	5	99.29%	703
	TiltLeft	0.43%	3	99.57%	690
	TiltForward	0.00%	0	100.00%	676
	TiltBackward	0.15%	1	99.85%	652
	WebbingLength	0.00%	0	100.00%	2720
	KisiOn	0.00%	0	100.00%	86
	KisiOff	0.00%	0	100.00%	86
	WebSenseNoLock	0.00%	0	100.00%	2720
	WebSenseLock	0.00%	0	100.00%	2720

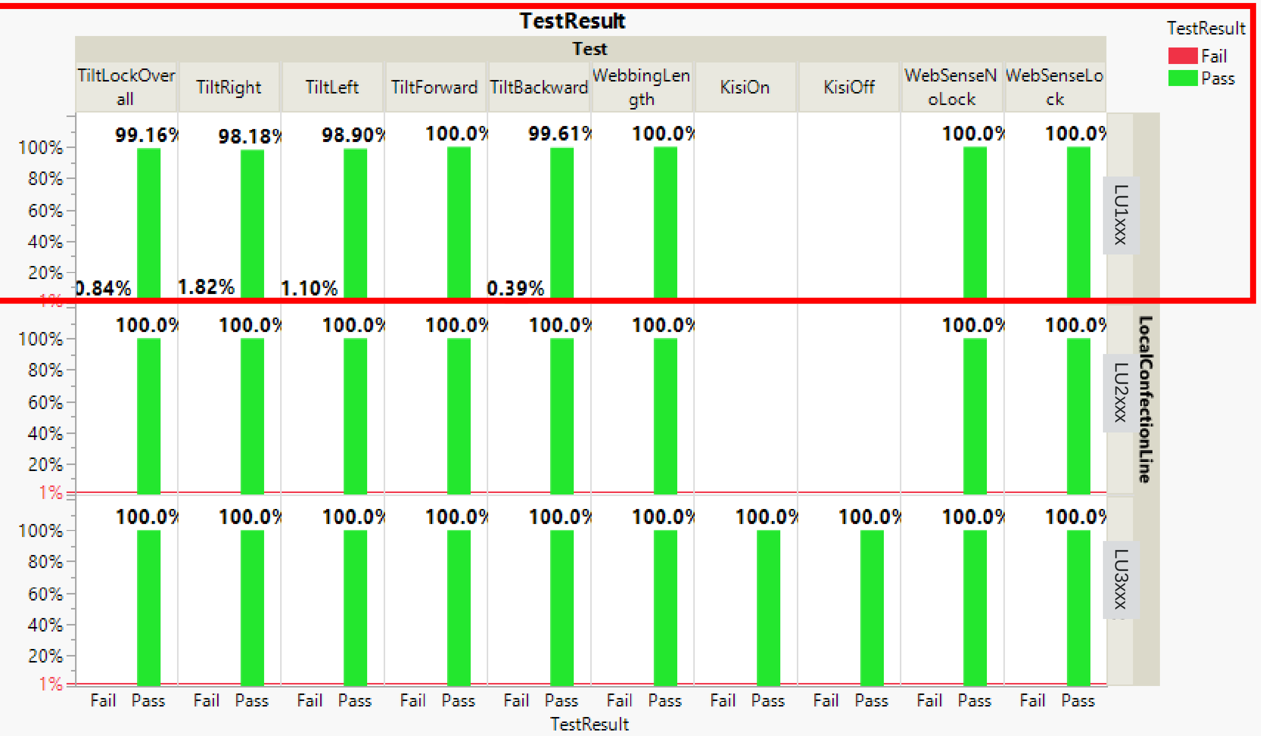


			UploadDate		
			2022-12-24		
			StartDate		
			2022-12-23		
			StartTime		
LocationConfectionLine	MachineID	LocalConfection	Min	Max	N
ALH	FCS-M-123xx	LU1xxx	06:13:49	21:08:04	5313
	FCS-M-124xx	LU2xxx	06:18:53	20:49:40	4096
	FCS-M-125xx	LU3xxx	06:03:09	21:04:25	4382

Report Structure:

(Table 1 Table 2
Graph 1 Graph 2)





Alarm Criteria are fulfilled => Alarm Report will be created for machine FCS-M-123xx

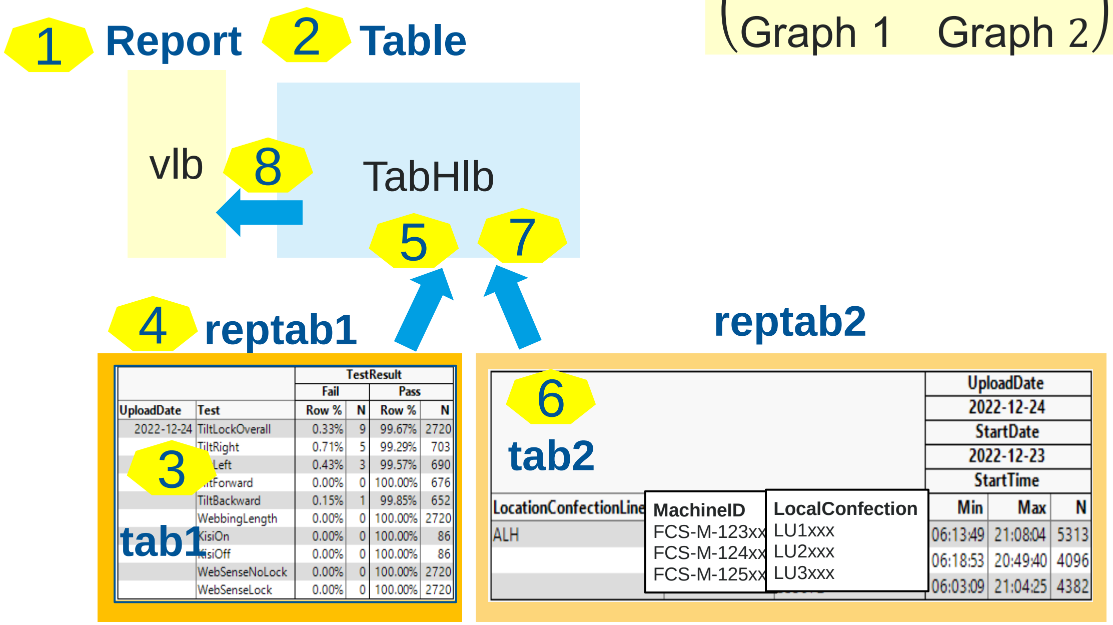
Creating DailyReports with JMP

„New Window“ and **vertical/horizontal Boxes** are used to ensure that **position and size of windows will stay the same**, even the **user is not logged in.** Bad Example: DashbordUserUnlogged.png

Report Structure:

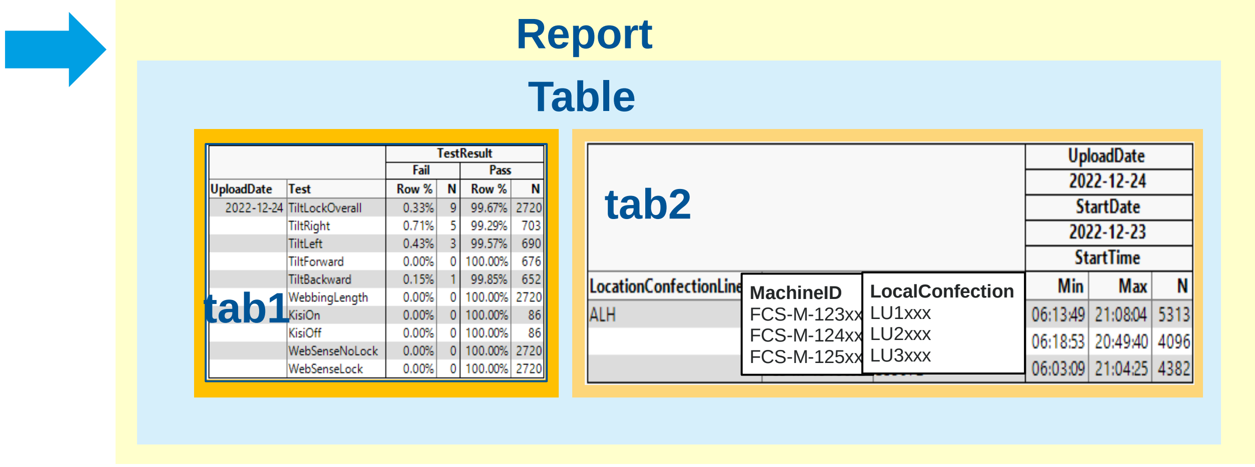
$$\begin{pmatrix} \text{Table 1} & \text{Table 2} \\ \text{Graph 1} & \text{Graph 2} \end{pmatrix}$$

```
1  nwt  = New Window("Report",   vlb = VlistBox()); //Vertical Box

2  nwth = New Window("Table", TabHlb = HlistBox()); //Horizontal Box

3  tab1 = dt << Tabulate(Show Control Panel( 0 ),
               Add Table(Column Table( Grouping Columns(  :TestResult ),
                                       Statistics( Row %, N ) ),
               Row Table( Grouping Columns( :UploadDate, :Test ) )),
               Local Data Filter(Close Outline( 1 ), Add Filter( columns(
               :LastMeasurement ), Where( :LastMeasurement == "Last" ) )));

4   reptab1 = Report( tab1 )[Tabulate Box( 1 )];
    tab1 << Move Window(0,0);

5   TabHlb << Append(reptab1);

6   tab2 = dt << Tabulate( … );
    reptab2 = Report( tab2 )[Tabulate Box( 1 )];
    tab2    << Move Window(900,0);

7    TabHlb << Append(reptab2);

8    nwt    << Append(TabHlb);

    //tab1 << Close Window; will be used later
    tab2   << Close Window;
    TabHlb << Close Window;
```



		TestResult			
		Fail		Pass	
UploadDate	Test	Row %	N	Row %	N
2022-12-24	TiltLockOverall	0.33%	9	99.67%	2720
	TiltRight	0.71%	5	99.29%	703
	TiltLeft	0.43%	3	99.57%	690
	TiltForward	0.00%	0	100.00%	676
	TiltBackward	0.15%	1	99.85%	652
	WebbingLength	0.00%	0	100.00%	2720
	KisiOn	0.00%	0	100.00%	86
	KisiOff	0.00%	0	100.00%	86
	WebSenseNoLock	0.00%	0	100.00%	2720
	WebSenseLock	0.00%	0	100.00%	2720

			UploadDate		
			2022-12-24		
			StartDate		
			2022-12-23		
			StartTime		
LocationConfectionLine	MachineID	LocalConfection	Min	Max	N
ALH	FCS-M-123xx	LU1xxx	06:13:49	21:08:04	5313
	FCS-M-124xx	LU2xxx	06:18:53	20:49:40	4096
	FCS-M-125xx	LU3xxx	06:03:09	21:04:25	4382



TestResult Fail Pass UploadDate Test Row % N Row % N 2022-12-24 TiltLockOverall 0.33% 9 99.67% 2720 TiltRight 0.71% 5 99.29% 703 TiltLeft 0.43% 3 99.57% 690 TiltForward 0.00% 0 100.00% 676 TiltBackward 0.15% 1 99.85% 652 WebbingLength 0.00% 0 100.00% 2720 KisiOn 0.00% 0 100.00% 86 KisiOff 0.00% 0 100.00% 86 WebSenseNoLock 0.00% 0 100.00% 2720 WebSenseLock 0.00% 0 100.00% 2720
UploadDate 2022-12-24 StartDate 2022-12-23 StartTime LocationConfectionLine MachineID LocalConfection Min Max N ALH FCS-M-123xx LU1xxx 06:13:49 21:08:04 5313 FCS-M-124xx LU2xxx 06:18:53 20:49:40 4096 FCS-M-125xx LU3xxx 06:03:09 21:04:25 4382

Saving DailyReports as *.png

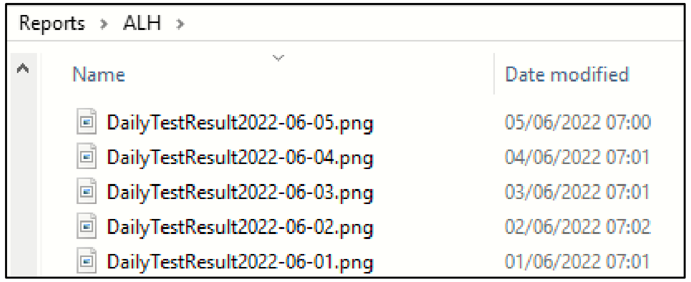
We save reports as ***.png**, since …

- there **doesn’t exist page breaks**
  - **high flexibility** to construct reports via horizontal and vertical boxes
- **no additional software** is needed for opening the file.

If the same report “ntw” would be saved as ***.pdf**, then it results into a file with **4 pages**, namely, Table 1, Table 2, Graph 1, Graph 2.

Folder structure on the server:

- Reports
  - Plant (ALH, ARO, …)
    - → Daily Report-file



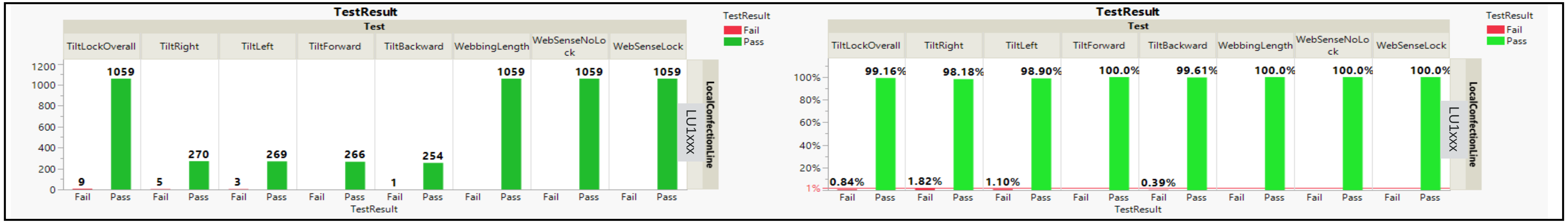
Reports › ALH ›

Name	Date modified
DailyTestResult2022-06-05.png	05/06/2022 07:00
DailyTestResult2022-06-04.png	04/06/2022 07:01
DailyTestResult2022-06-03.png	03/06/2022 07:01
DailyTestResult2022-06-02.png	02/06/2022 07:02
DailyTestResult2022-06-01.png	01/06/2022 07:01

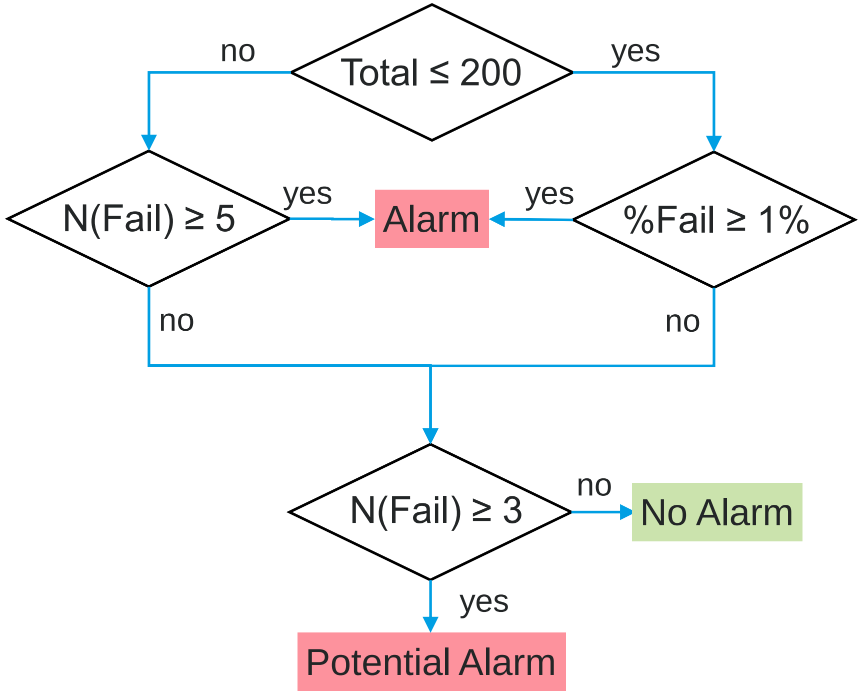
$$\text{ntw} = \begin{pmatrix} \text{Table 1} & \text{Table 2} \\ \text{Graph 1} & \text{Graph 2} \end{pmatrix}$$

```
FileDailyTestResult = "\\servername\Reports$\"||Eval(plant)||"\DailyTestResult"||Char( Format(Today(), "YYYY-MM-DD"))||".png";

nwt << Save Picture( FileDailyTestResult);
```

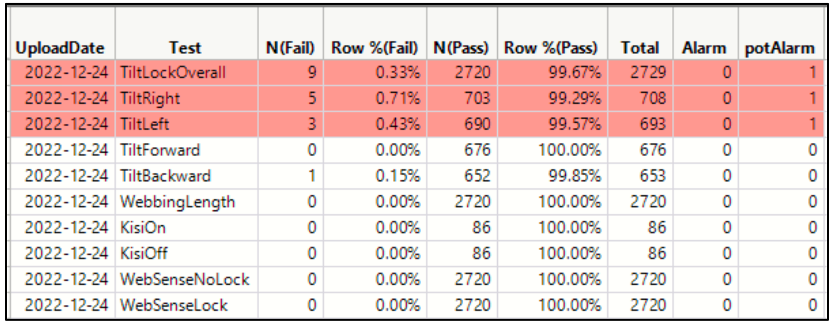
Creating Alarm Reports for Root Cause Analysis



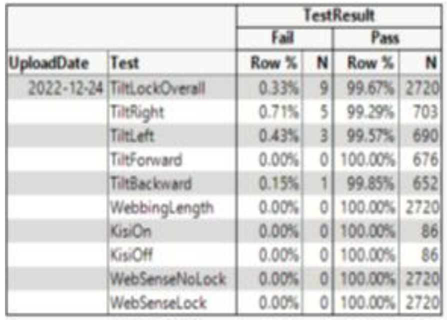
Rules for an Alarm for every Level:



1. Level: Potential Alarm Criterium is fulfilled **overall machines** in ALH

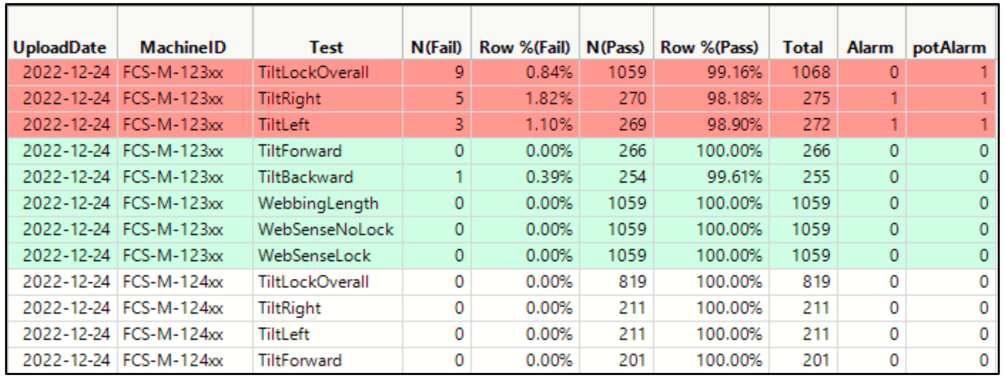


UploadDate	Test	N(Fail)	Row %(Fail)	N(Pass)	Row %(Pass)	Total	Alarm	potAlarm
2022-12-24	TiltLockOverall	9	0.33%	2720	99.67%	2729	0	1
2022-12-24	TiltRight	5	0.71%	703	99.29%	708	0	1
2022-12-24	TiltLeft	3	0.43%	690	99.57%	693	0	1
2022-12-24	TiltForward	0	0.00%	676	100.00%	676	0	0
2022-12-24	TiltBackward	1	0.15%	652	99.85%	653	0	0
2022-12-24	WebbingLength	0	0.00%	2720	100.00%	2720	0	0
2022-12-24	KisiOn	0	0.00%	86	100.00%	86	0	0
2022-12-24	KisiOff	0	0.00%	86	100.00%	86	0	0
2022-12-24	WebSenseNoLock	0	0.00%	2720	100.00%	2720	0	0
2022-12-24	WebSenseLock	0	0.00%	2720	100.00%	2720	0	0



		TestResult			
		Fail		Pass	
UploadDate	Test	Row %	N	Row %	N
2022-12-24	TiltLockOverall	0.33%	9	99.67%	2720
	TiltRight	0.71%	5	99.29%	703
	TiltLeft	0.43%	3	99.57%	690
	TiltForward	0.00%	0	100.00%	676
	TiltBackward	0.15%	1	99.85%	652
	WebbingLength	0.00%	0	100.00%	2720
	KisiOn	0.00%	0	100.00%	86
	KisiOff	0.00%	0	100.00%	86
	WebSenseNoLock	0.00%	0	100.00%	2720
	WebSenseLock	0.00%	0	100.00%	2720

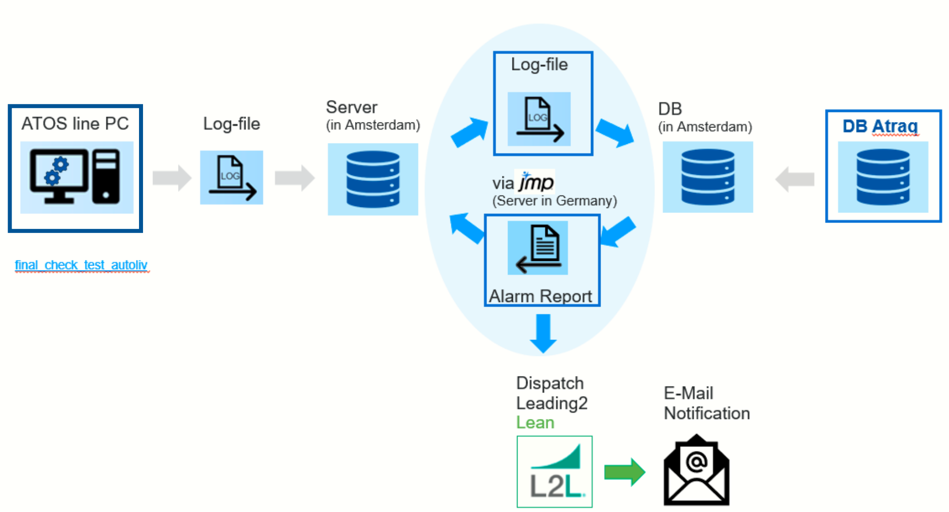
2. Level: Alarm Criterium is fulfilled for **machine FCS-M-123xx**



UploadDate	MachineID	Test	N(Fail)	Row %(Fail)	N(Pass)	Row %(Pass)	Total	Alarm	potAlarm
2022-12-24	FCS-M-123xx	TiltLockOverall	9	0.84%	1059	99.16%	1068	0	1
2022-12-24	FCS-M-123xx	TiltRight	5	1.82%	270	98.18%	275	1	1
2022-12-24	FCS-M-123xx	TiltLeft	3	1.10%	269	98.90%	272	1	1
2022-12-24	FCS-M-123xx	TiltForward	0	0.00%	266	100.00%	266	0	0
2022-12-24	FCS-M-123xx	TiltBackward	1	0.39%	254	99.61%	255	0	0
2022-12-24	FCS-M-123xx	WebbingLength	0	0.00%	1059	100.00%	1059	0	0
2022-12-24	FCS-M-123xx	WebSenseNoLock	0	0.00%	1059	100.00%	1059	0	0
2022-12-24	FCS-M-123xx	WebSenseLock	0	0.00%	1059	100.00%	1059	0	0
2022-12-24	FCS-M-124xx	TiltLockOverall	0	0.00%	819	100.00%	819	0	0
2022-12-24	FCS-M-124xx	TiltRight	0	0.00%	211	100.00%	211	0	0
2022-12-24	FCS-M-124xx	TiltLeft	0	0.00%	211	100.00%	211	0	0
2022-12-24	FCS-M-124xx	TiltForward	0	0.00%	201	100.00%	201	0	0

Autoliv

Join Test Data from ATOS-line with ATRAQ Component Data

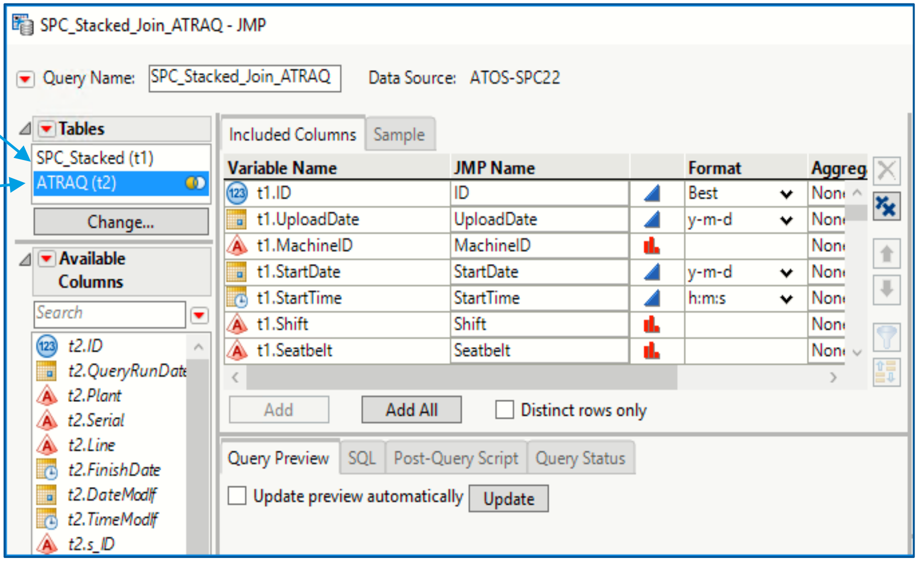


Atraq is used at Autoliv …

- as a **traceability system** for assembly processes
- for **product identification** and
- for **product components** data storage
- as **process parameter** storage per product ...

1. Select Data tables in DB

- ATOS filtered by alarm data
  - Machine FCS-M-123xx
  - TiltLock
- ATRAQ



SPC_Stacked_Join_ATRAQ - JMP

Query Name: SPC_Stacked_Join_ATRAQ   Data Source: ATOS-SPC22

Tables
- SPC_Stacked (t1)
- ATRAQ (t2)

Change...

Available Columns

Search

- t2.ID
- t2.QueryRunDate
- t2.Plant
- t2.Serial
- t2.Line
- t2.FinishDate
- t2.DateModf
- t2.TimeModf
- t2.s_ID

Included Columns | Sample

Variable Name	JMP Name		Format	Aggreg
t1.ID	ID		Best	Non
t1.UploadDate	UploadDate		y-m-d	Non
t1.MachineID	MachineID			Non
t1.StartDate	StartDate		y-m-d	Non
t1.StartTime	StartTime		h:m:s	Non
t1.Shift	Shift			Non
t1.Seatbelt	Seatbelt			Non

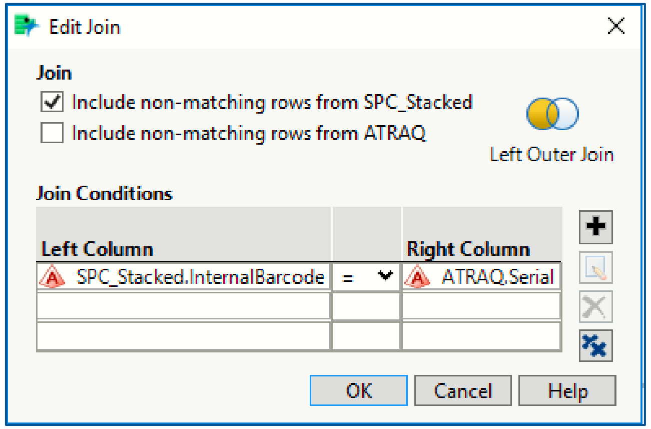
Add   Add All   □ Distinct rows only

Query Preview | SQL | Post-Query Script | Query Status

□ Update preview automatically   Update

2. Make a **left outer join**

"InternalBarcode" & "Serial" are **unique codes**



Edit Join

Join

☑ Include non-matching rows from SPC_Stacked

□ Include non-matching rows from ATRAQ

Left Outer Join

Join Conditions

Left Column		Right Column
SPC_Stacked.InternalBarcode	=	ATRAQ.Serial

OK   Cancel   Help

3. Run a **new query**

Alarm Report Structure per Machine/Local Confection Line

Every alarm report has the same structure:

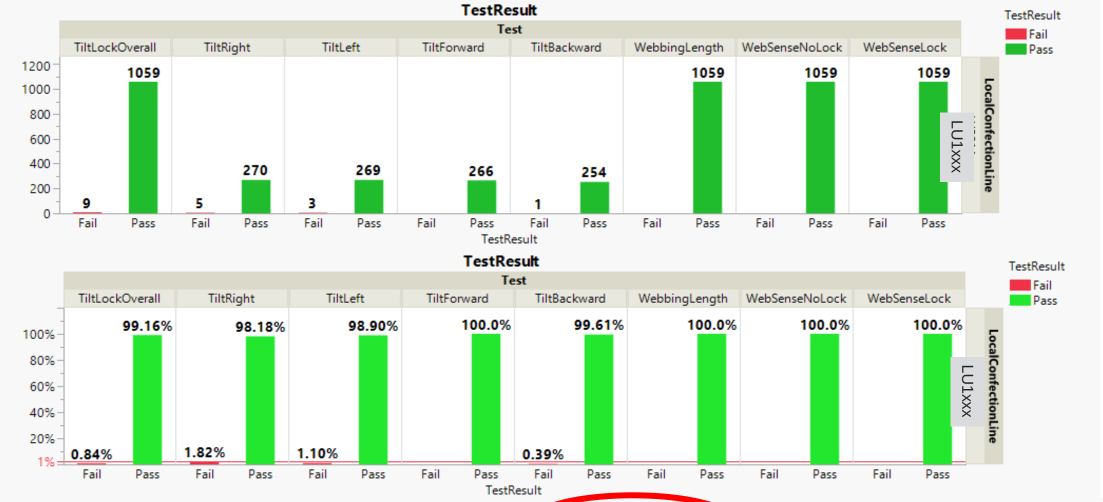
1. **Summary Tables** of machine FCS-M-xyz
2. **Summary Bar Charts** of machine FCS-M-xyz
3. **ATOS-Tables of all input parameter** restricted to the **effected seatbelt number**
4. **Explanation** of “SummaryMachinePara” based on the corresponding test
5. **ATAQ-Tables of all input parameter** restricted to the **effected seatbelt number**

1.

					TestResult			
					Fail		Pass	
UploadDate	LocationConfectionLine	MachineID	LocalConfectionLine	Test	Row %	N	Row %	N
2022-12-24	ALH	FCS-M-123xx	LU1xxx	TiltLockOverall	0.84%	9	99.16%	1059
				TiltRight	1.82%	5	98.18%	270
				TiltLeft	1.10%	3	98.90%	269
				TiltForward	0.00%	0	100.00%	266
				TiltBackward	0.39%	1	99.61%	254
				WebbingLength	0.00%	0	100.00%	1059
				WebSenseNoLock	0.00%	0	100.00%	1059
				WebSenseLock	0.00%	0	100.00%	1059

	UploadDate		
	2022-12-24		
	StartDate		
	2022-12-23		
	StartTime		
LocalConfectionLine	Min	Max	N
LU1xxx	06:13:49	21:08:04	5313

2.



Information of affected Seatbelts from ATOS and ATRAQ for TiltLockOverall

3.

													TestResult			
													Fail		Pass	
LocalConfectionLine	StartDate	Shift	Test	Seatbelt	Retractor	ProductionDay	GlobalLineID	ProductFamily	Customer	CarPositionWX	CarPositionWY	SummaryMachinePara	Row %	N	Row %	N
LU1xxx	2022-12-23	afternoon	TiltLockOverall	SB12345	R123456x	22356	L00123	R123	xyzabc	-12	0	15;27;20;0;Continuous;.;500;5000;8;10;37;50	2.50%	3	97.50%	117
							L00456	R123	xyzabc	-12	0	15;27;20;0;Continuous;.;500;5000;8;10;37;50	2.53%	6	97.47%	231
						22357	L00456	R123	xyzabc	-12	0	15;27;20;0;Continuous;.;500;5000;8;10;37;50	0.00%	0	100.00%	72
		morning	TiltLockOverall	SB12345	R123456x	22356	L00123	R123	xyzabc	-12	0	15;27;20;0;Continuous;.;500;5000;8;10;37;50	0.00%	0	100.00%	494
							L00456	R123	xyzabc	-12	0	15;27;20;0;Continuous;.;500;5000;8;10;37;50	0.00%	0	100.00%	60
						22357	L00456	R123	xyzabc	-12	0	15;27;20;0;Continuous;.;500;5000;8;10;37;50	0.00%	0	100.00%	72

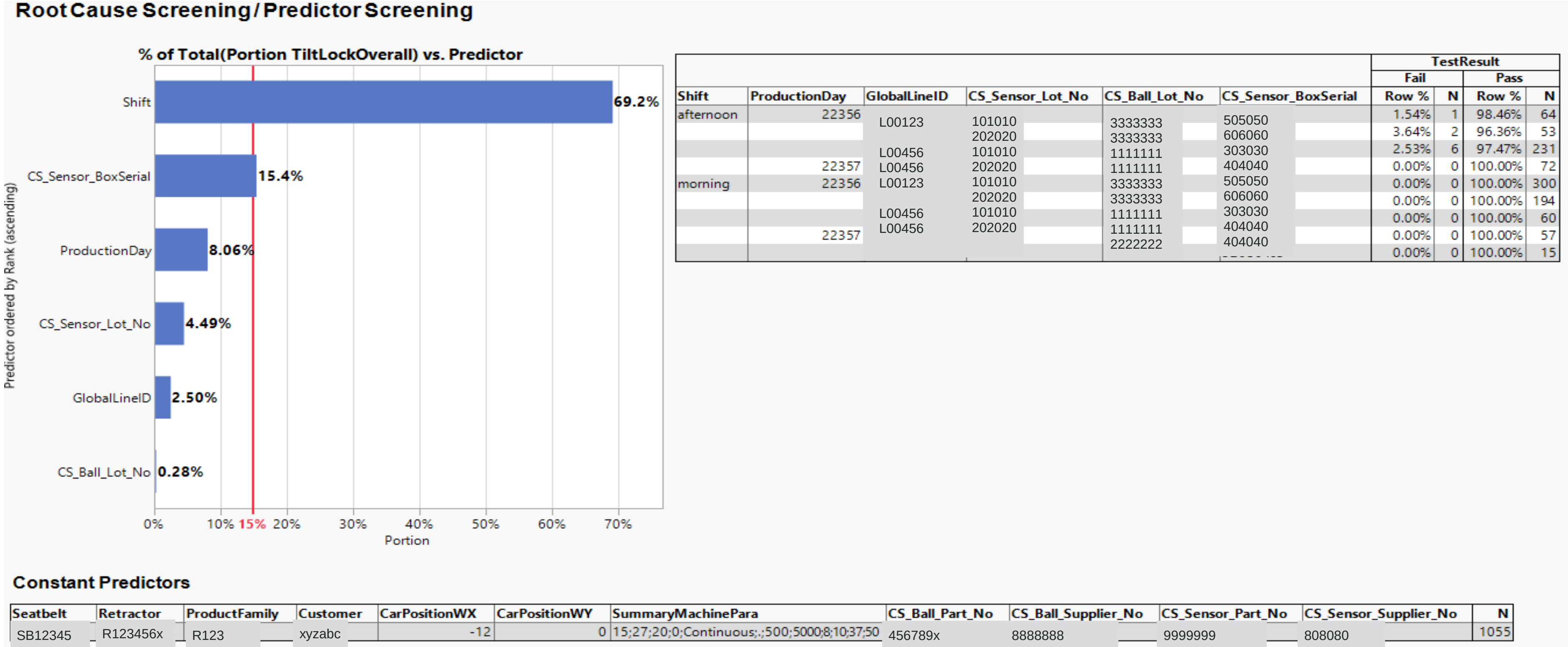
4.

SummaryMachinePara	List of MachinePara	N
TiltLockSummary	LSL; USL; LockThreshold; Delay; Method; SenseDist; SenseVel; SenseAcc; WebStartAngle; TiltVel; TiltAcc; WebbingAtStart	1

5.

	CS-Ball				CS-Sensor				WS-Sensor				TestResult			
													Fail		Pass	
Seatbelt	CS_Ball_Part_No	CS_Ball_Lot_No	CS_Ball_BoxSerial	CS_Ball_Supplier_No	CS_Sensor_Part_No	CS_Sensor_Lot_No	CS_Sensor_BoxSerial	CS_Sensor_Supplier_No	WS_Sensor_Part_No	WS_Sensor_Lot_No	WS_Sensor_BoxSerial	WS_Sensor_Supplier_No	N	Row %	N	Row %
SB12345	456789x	1111111	4444444	8888888	9999999	101010	303030	808080	Missing	Missing	Missing	Missing	6	2.02%	291	97.98%
						202020	404040	808080	Missing	Missing	Missing	Missing	0	0.00%	129	100.00%
		2222222	5555555	8888888	9999999	202020	404040	808080	Missing	Missing	Missing	Missing	0	0.00%	15	100.00%
		3333333	6666666	8888888	9999999	101010	505050	808080	Missing	Missing	Missing	Missing	1	0.47%	212	99.53%
						202020	606060	808080	Missing	Missing	Missing	Missing	2	0.80%	247	99.20%
			7777777	8888888	9999999	101010	707070	808080	Missing	Missing	Missing	Missing	0	0.00%	152	100.00%

Alarm Report Structure per Machine/Local Confection Line



Root Cause Screening / Predictor Screening

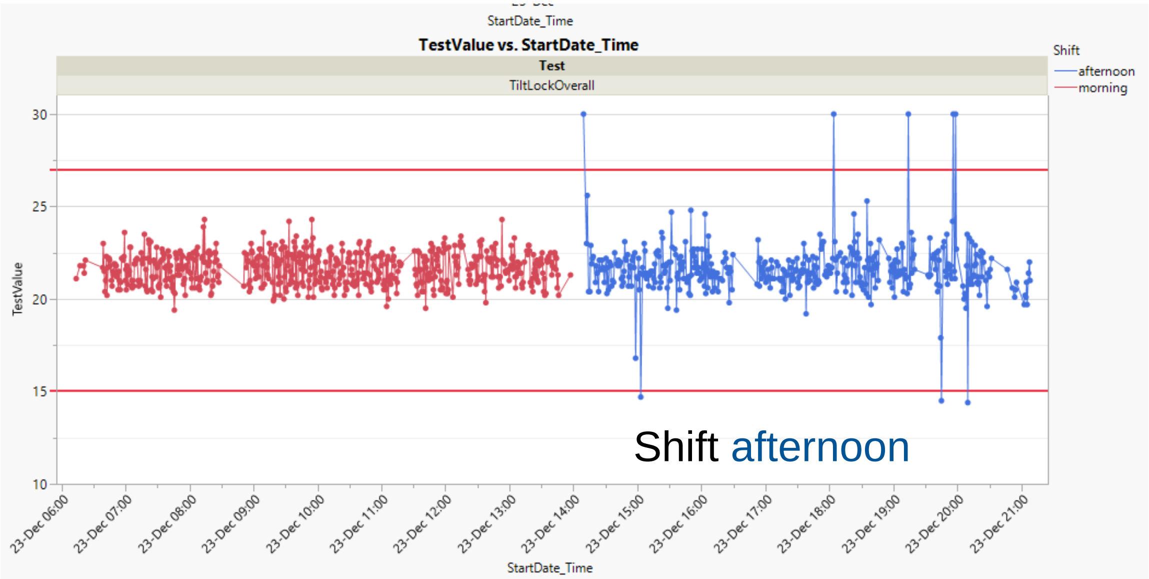
						TestResult			
						Fail		Pass	
Shift	ProductionDay	GlobalLineID	CS_Sensor_Lot_No	CS_Ball_Lot_No	CS_Sensor_BoxSerial	Row %	N	Row %	N
afternoon	22356	L00123	101010	3333333	505050	1.54%	1	98.46%	64
			202020	3333333	606060	3.64%	2	96.36%	53
		L00456	101010	1111111	303030	2.53%	6	97.47%	231
	22357	L00456	202020	1111111	404040	0.00%	0	100.00%	72
morning	22356	L00123	101010	3333333	505050	0.00%	0	100.00%	300
			202020	3333333	606060	0.00%	0	100.00%	194
		L00456	101010	1111111	303030	0.00%	0	100.00%	60
	22357	L00456	202020	1111111	404040	0.00%	0	100.00%	57
				2222222	404040	0.00%	0	100.00%	15

Constant Predictors

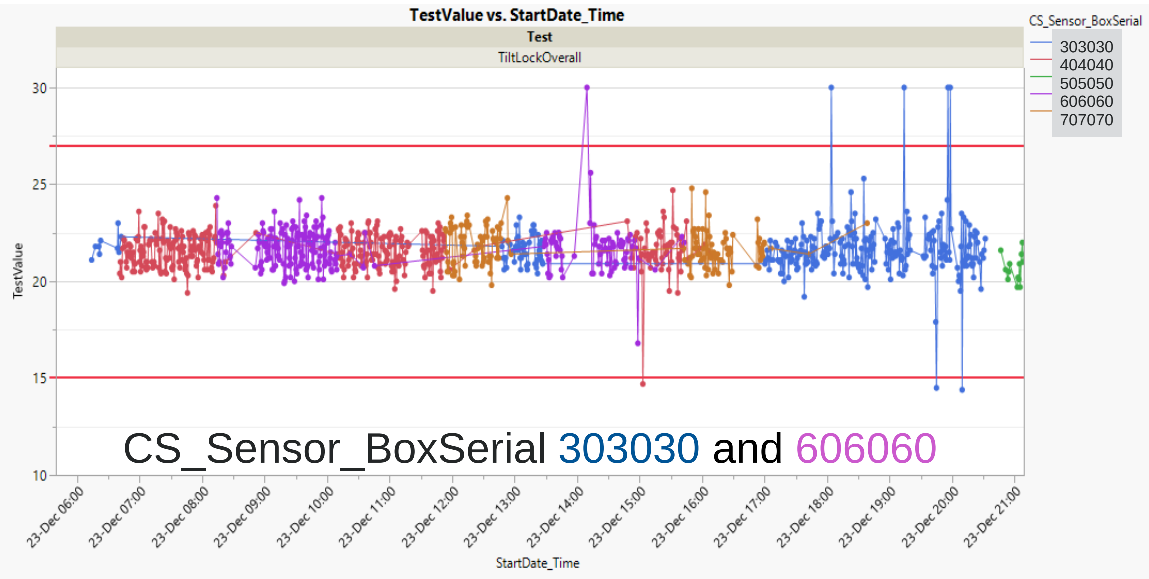
Seatbelt	Retractor	ProductFamily	Customer	CarPositionWX	CarPositionWY	SummaryMachinePara	CS_Ball_Part_No	CS_Ball_Supplier_No	CS_Sensor_Part_No	CS_Sensor_Supplier_No	N
SB12345	R123456x	R123	xyzabc	-12	0	15;27;20;0;Continuous;.;500;5000;8;10;37;50	456789x	8888888	9999999	808080	1055

6. **Predictor Screening** for the test results **(pass/fail)**
   - Predictor screening uses **bootstrap forest partitioning** to evaluate the **contribution of predictors on the response**.
   - Predictor screening can **identify predictors** that might be **weak alone but strong** when used **in combination with other predictors**
7. **Table of Predictors** used for Predictor Screening
8. **Table of constant Predictors**

Alarm Report Structure per Machine/Local Confection Line

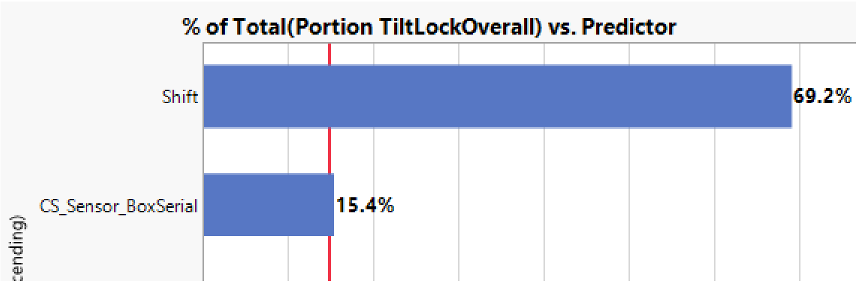


Shift afternoon



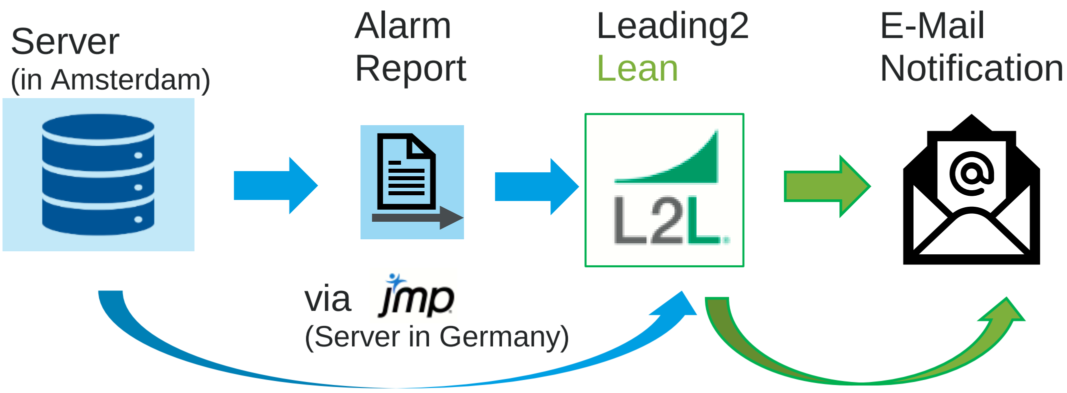
CS_Sensor_BoxSerial 303030 and 606060

9. **Graphs "TestValue vs. StartDateTime"** …
   - according to the **predictors**, whose **screening proportion ≥ 0,15**
   - with **colored test value** according to the **levels of the predictor** variable
   - with **specification limits as reference lines** according to the machine parameter list
   - with a calculated **adjusted range**



Leading2Lean Dispatch System

- Leading2Lean (L2L) is a **continuous improvement tool** at Autoliv, for **real-time production** and **maintenance management.**
- L2L is configured to **automatically send notifications** to the **correct person** (machine owner, problem solver) via mail.
- L2L includes a **Dispatch Change History**
  - **Who** created the dispatch?
  - **When** was the dispatch created ?
  - **Who** made each edit and addition?
  - **Who** completed and closed the dispatch?



Folder structure on the server:

- Reports
  - Machine Number (FCS-M-xxxxxxx)

    → Alarm Report-file

    Alarm_FCS-M-123xx_2022-12-24.png

An alarm notification can be ignored, but a **notification via L2L includes a dispatch process**, which **must be closed**

Sending Web-based Links of Alarm Reports to L2L via JMP

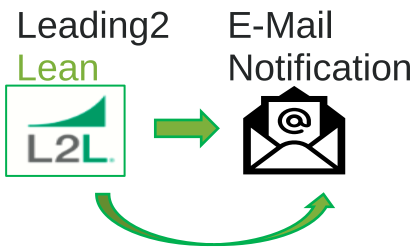
If there is an alarm, then

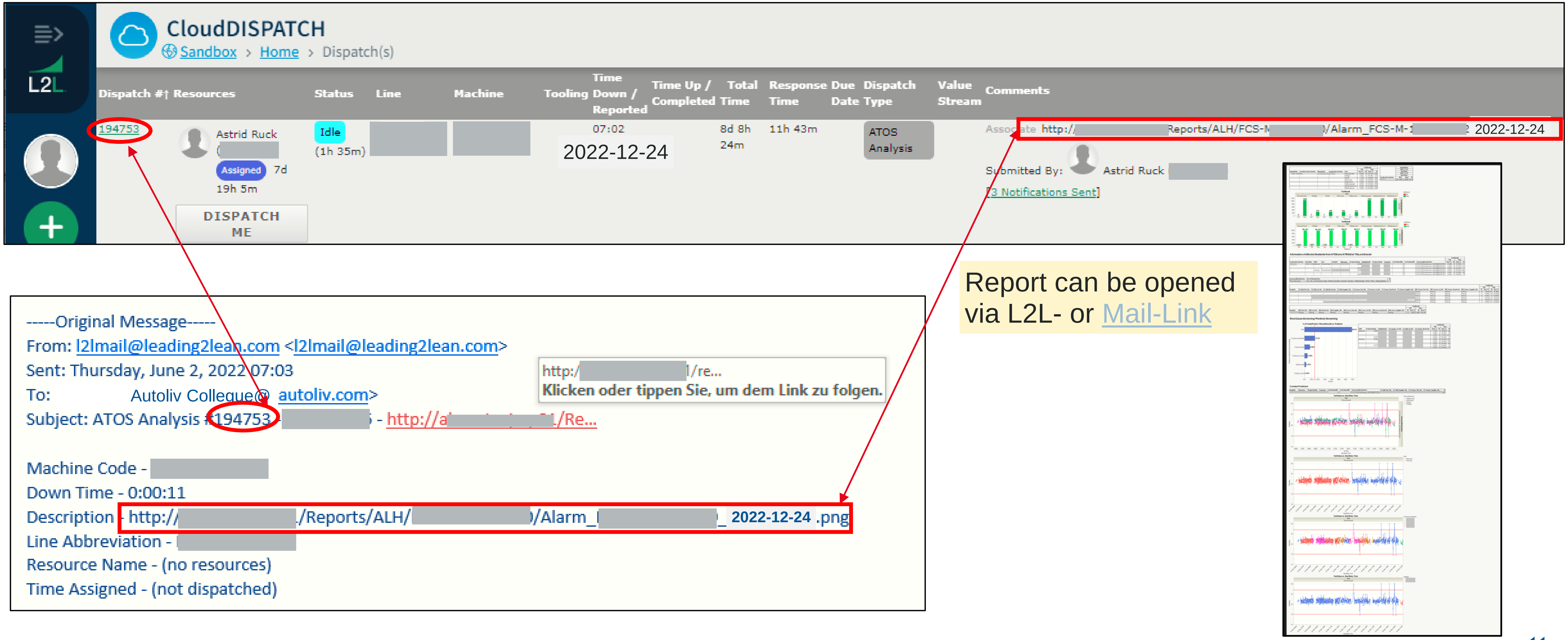
- save the alarm report
- assign the **corresponding http-path to** a **variable “alarm”**
- Create an **associative array** (dictionary/ hash map) based on **keys** and **values**

```
alarm = "http://server/Reports/ALH/FCS-M-123xx/Alarm_FCS-M-123xx_2022-12-24.png"



array = [=>];
array["auth"] = “12345678abcde";
array["site"] = 123;
array["user"] = “my-user-id";
array["tradecode"] = "Associate";
array["machinecode"] = matchL2L;
array["description"] = alarm;
array["dispatchtypecode"] = "ATOS Analysis";


httppost = New HTTP Request(
               URL("https://autoliv-eu.leading2lean.com/api/1.0/dispatches /open/"),
               Method( "POST"),
               Form(Fields(array))  );
data = httppost << Send
```

- **Run** a **New HTTP Request,** which sends the **link** of the **alarm report into L2L**

L2L – Email – Alarm Report





Dispatch #↑	Resources	Status	Line	Machine	Time Tooling Down / Reported	Time Up / Completed	Total Time	Response Time	Due Date	Dispatch Type	Value Stream	Comments
194753	Astrid Ruck ([redacted] Assigned 7d 19h 5m DISPATCH ME	Idle (1h 35m)	[redacted]	[redacted]	07:02 2022-12-24		8d 8h 24m	11h 43m		ATOS Analysis		Associate http://[redacted]Reports/ALH/FCS-M[redacted]/Alarm_FCS-M-1[redacted] 2022-12-24 Submitted By: Astrid Ruck [redacted] [3 Notifications Sent]

-----Original Message-----
From: l2lmail@leading2lean.com <l2lmail@leading2lean.com>
Sent: Thursday, June 2, 2022 07:03
To: Autoliv Colleque@ autoliv.com>
Subject: ATOS Analysis #194753 [redacted]5 - http://a[redacted]1/Re...

http:/[redacted]/re...
Klicken oder tippen Sie, um dem Link zu folgen.

Machine Code - [redacted]
Down Time - 0:00:11
Description - http://[redacted]/Reports/ALH/[redacted]/Alarm_[redacted]_ 2022-12-24 .png
Line Abbreviation - [redacted]
Resource Name - (no resources)
Time Assigned - (not dispatched)

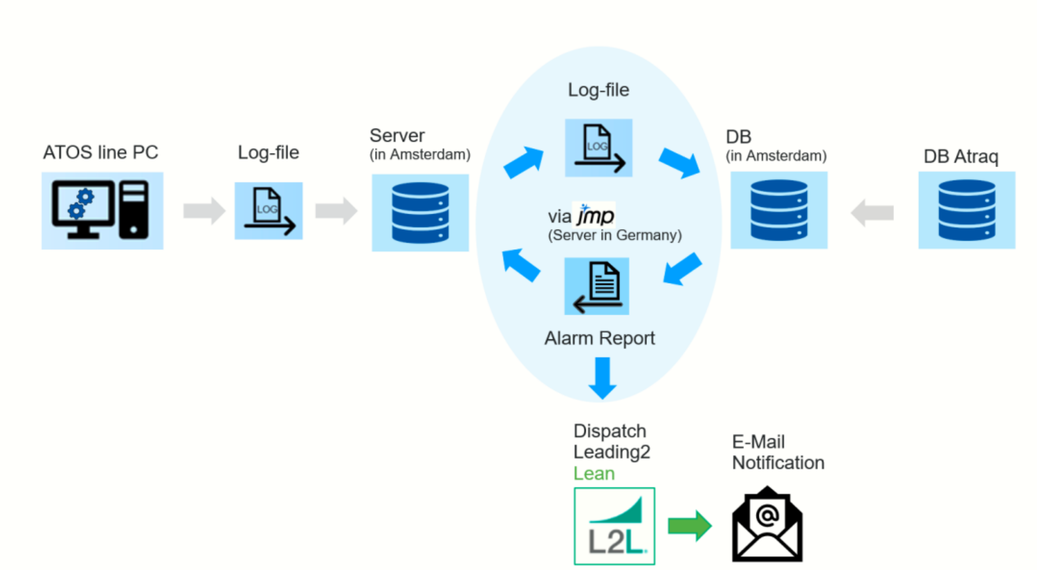
Report can be opened via L2L- or Mail-Link

Conclusion/Outlook

The total process of

- Importing **Multiple Files**
- Transferring data into a database via **SQL-Queries**
- **Joining tables** in a database via **SQL-Queries**
- **Creating Reports**
- **Predictor Screening**
- Running new **HTTP Requests** for sending **alarm report links** into L2L

can be realized via JMP.



In the same way components of **sub-assemblies could be used for root cause analysis by multiple joins of tables in a database.**

“JMP writes 90% of your code – the skeleton, …”

Jordan Hiller, JMP Senior Systems Engineer

=> the other 10% is learning by doing