



## Failure Codes

Failure Codes

Failure Class: NETWORK | Network Issues | Organization: EAGLENA | Attachments: View attachments

Problems (1 - 1 of 1)

Failure Code	Description
NETWACC	Can Not Access Network

Causes for NETWACC (1 - 3 of 3)

Failure Code	Description
SERVERDN	Server Down
CABLEEND	Faulty Cable End
CABLE	Faulty Network Cable

Remedies for SERVERDN (1 - 1 of 1)

Failure Code	Description
REBOOT	Reboot Server

# Identify Failure Coding to Help Control Defects

One tiny field on a work order.  
One giant leap to stop guessing and start knowing.

### The Model

Failure Class → Problem → Cause → Remedy

### The Method

Capture it in Maximo. Every time. 15 seconds or less.

### The Strategy

Turn coded work orders into reliability decisions

## LEARNING OBJECTIVES

### IDENTIFY FAILURES



**Identify standard**  
high level failure modes

### AWARENESS OF FAILURES



**Create an awareness of**  
what failures are and  
how to work toward  
eliminating them

### SITUATION AWARENESS



**Develop situational asset**  
**management** awareness of  
why something may fail

### ALIGN PROCESSES



Align your business  
processes to better facilitate  
elimination of defects found  
in your asset portfolio



## THE PROBLEM

# The Work Order That Lied

Picture this: Pump P-204. Six work orders in eight months. Six different techs. Six different descriptions — "noise," "vibrating," "leaking again," "adjusted," "checked okay," "repeat of last time." The asset history looks busy. But what does it tell you? **Nothing actionable.**

## What Was Written

- "Vibrating — adjusted"
- "Noise — lubed"
- "Leaking — tightened"
- "Repeat of last time"

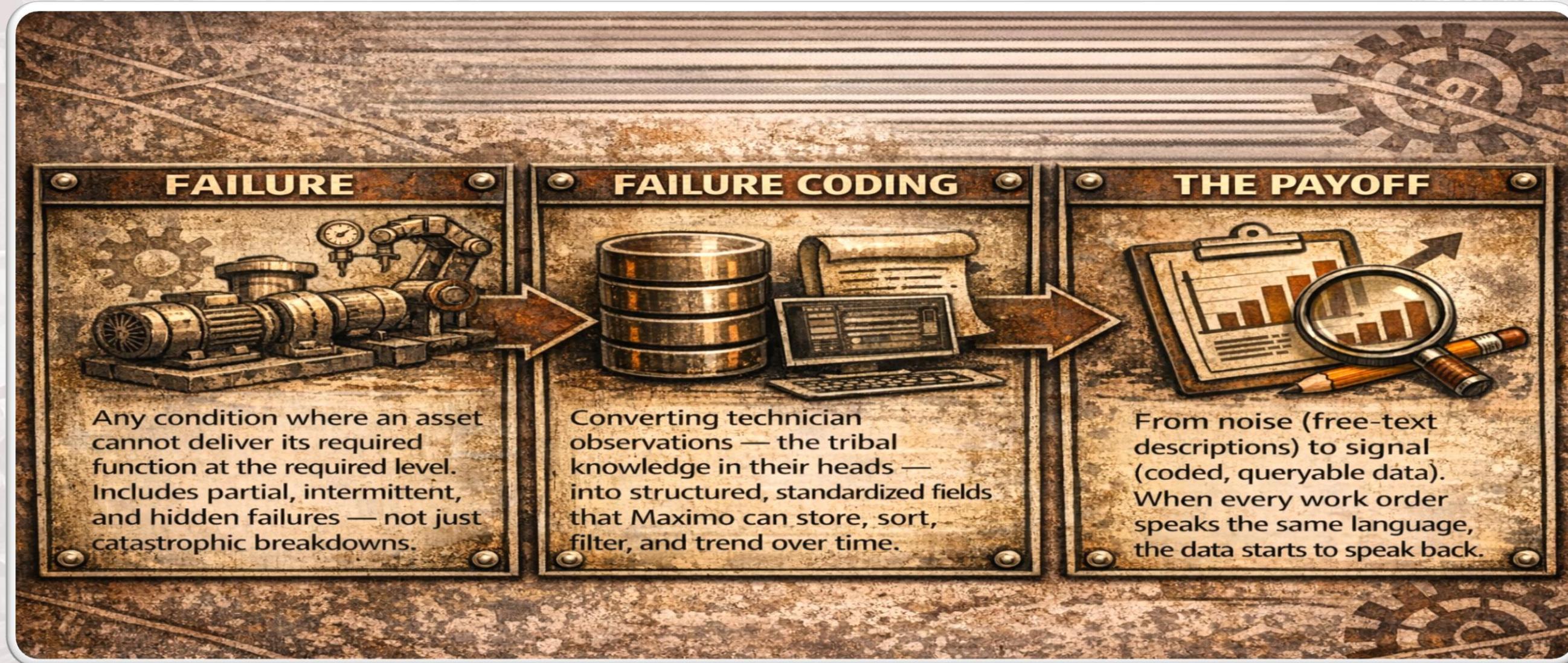
## What Was Lost

No failure class. No cause. No remedy code. No pattern. Just a long paper trail of tribal knowledge that walked out the door with the last tech who retired.

No codes = no memory. No memory = no strategy.

# What "Failure Coding" Actually Means

## Let's clear the air.

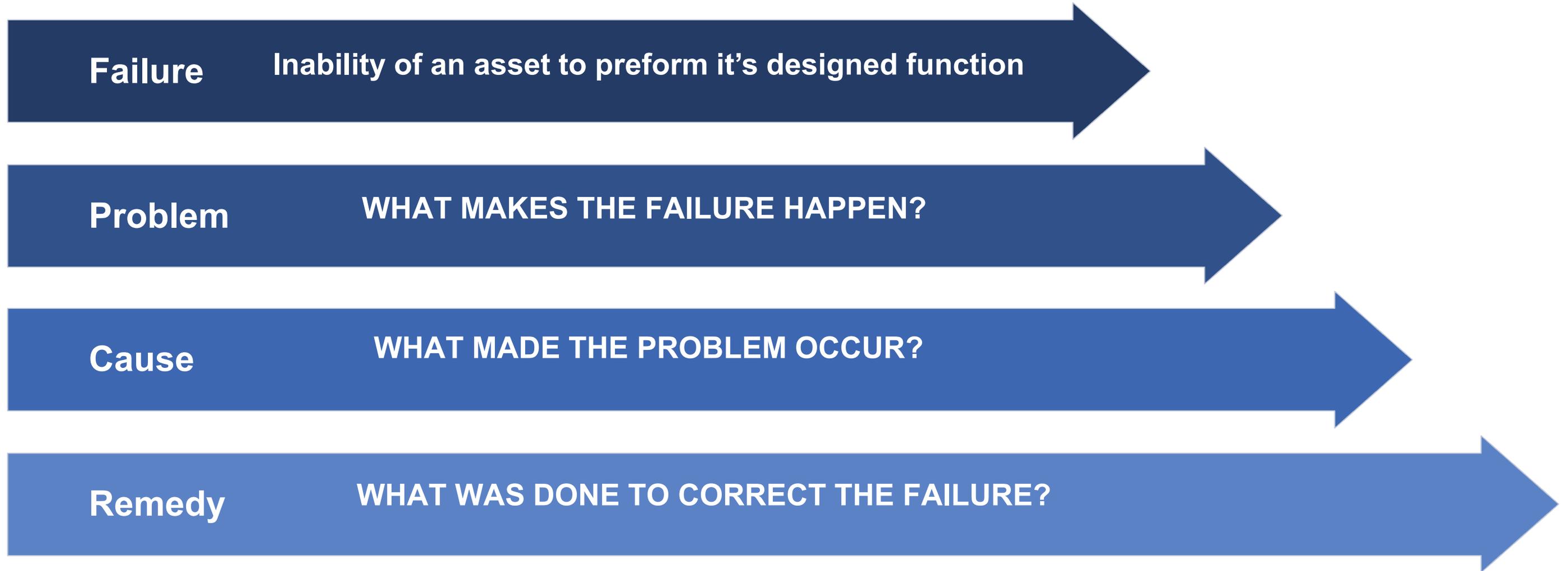


FAILED

# Failure → Problem → Cause → Remedy



This four-field hierarchy is the backbone of failure reporting in IBM Maximo Application Suite. It's reliability thinking made operational. Each field narrows the description, building a complete picture of what failed, why, and how it was resolved.



Every coded work order is a data point. A hundred coded work orders is a reliability dataset. That's when the patterns emerge — and the debates end.

# The CODES



## FAILURE

**ELECTRICAL**

**VENTILATION**

**WATER**

**FUEL**

**LOCK**

**UNDERGROUND**

**FIRE**

ERROR

## CAUSE

WHAT MADE THE PROBLEM OCCUR?

 TRIPPED BREAKER	 EQUIPMENT MALFUNCTION	 ASSEMBLY/SETUP	 IMPROPER STORAGE
 IMPROPER MAINTENANCE	 INADEQUATE MAINTENANCE	 INADEQUATE DESIGN	 END OF LIFE

ERROR

## PROBLEM

WHAT MAKES THE FAILURE HAPPEN?

- NO POWER
- LIGHTS OUT
- LEAK/BREACH
- FLOOD
- ALARM
- ALARM
- HAZARD

ERROR

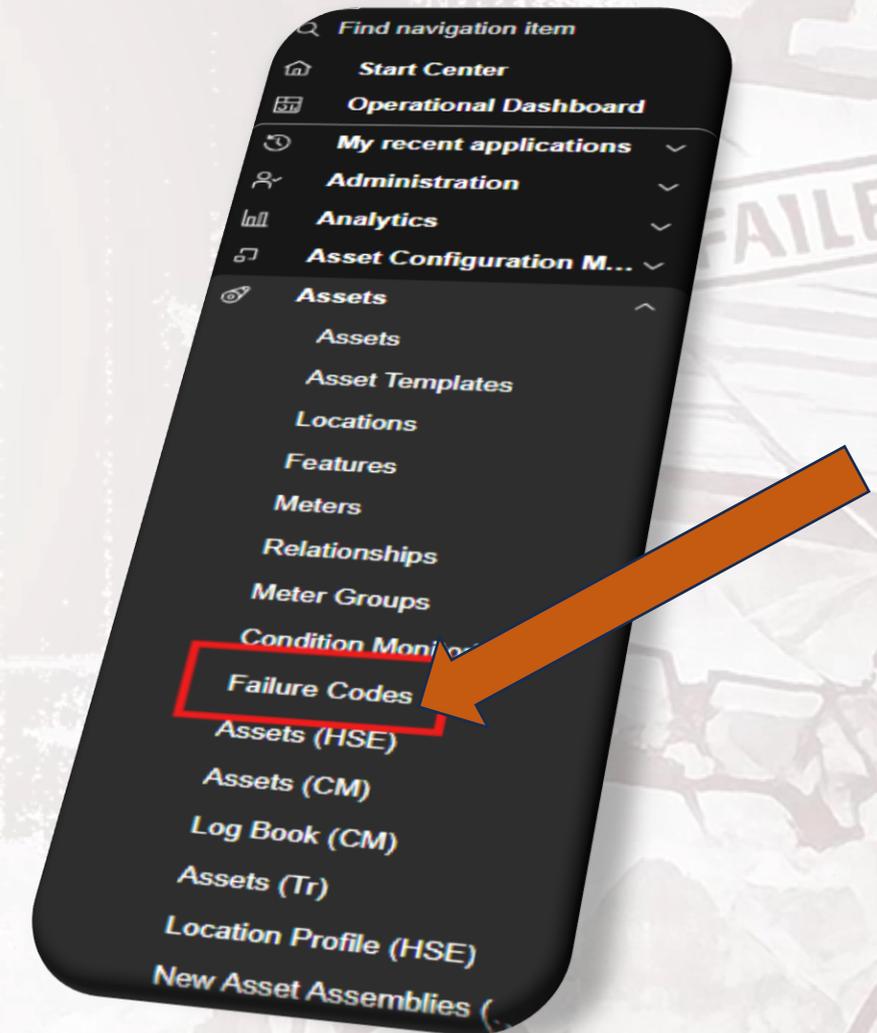
## REMEDY

WHAT WAS DONE TO CORRECT THE FAILURE?

FINALLY, THE FIX!!

- REPLACE
- REPAIR
- RESET
- ADJUSTED
- NO ACTION TAKEN

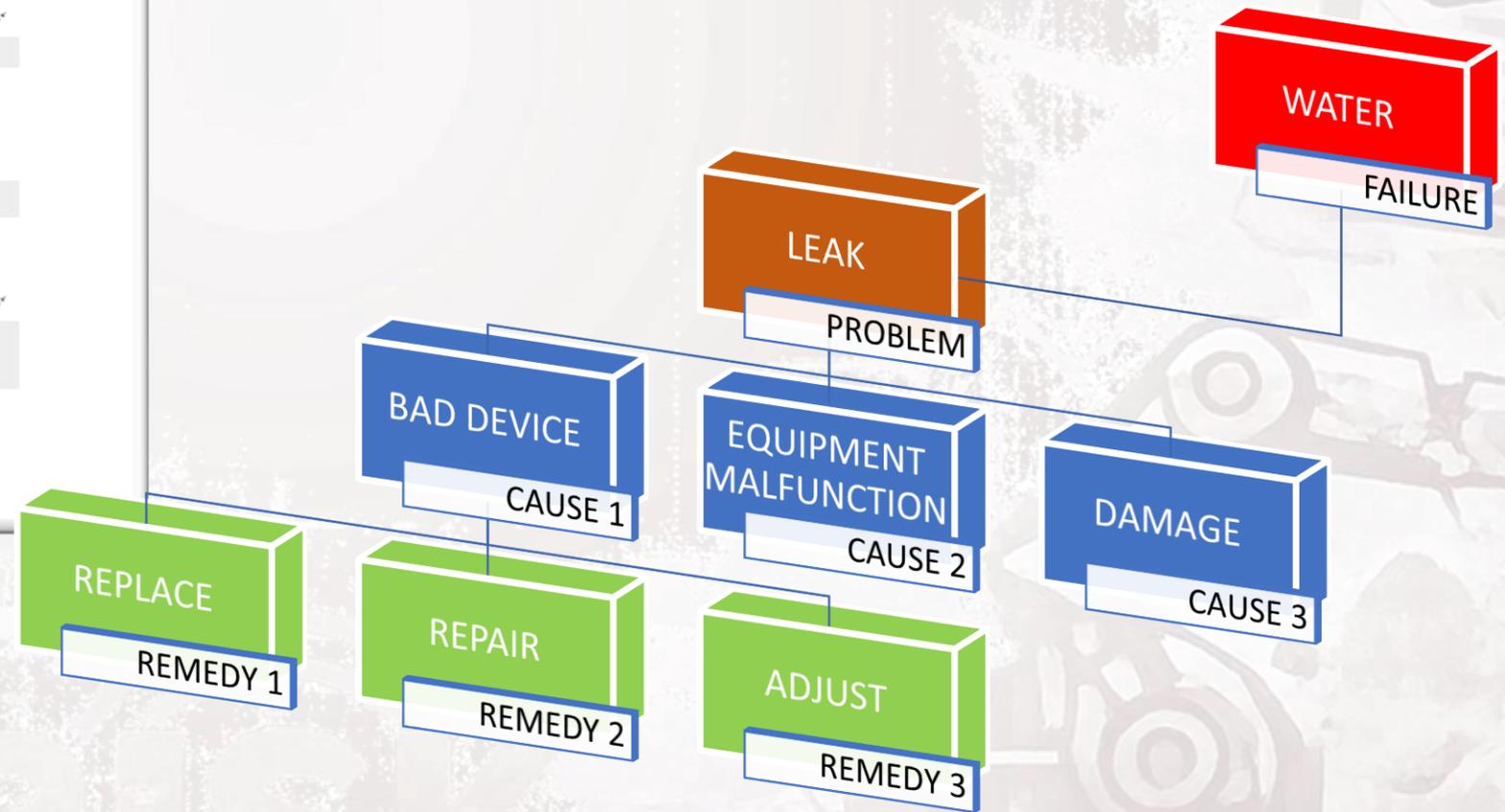
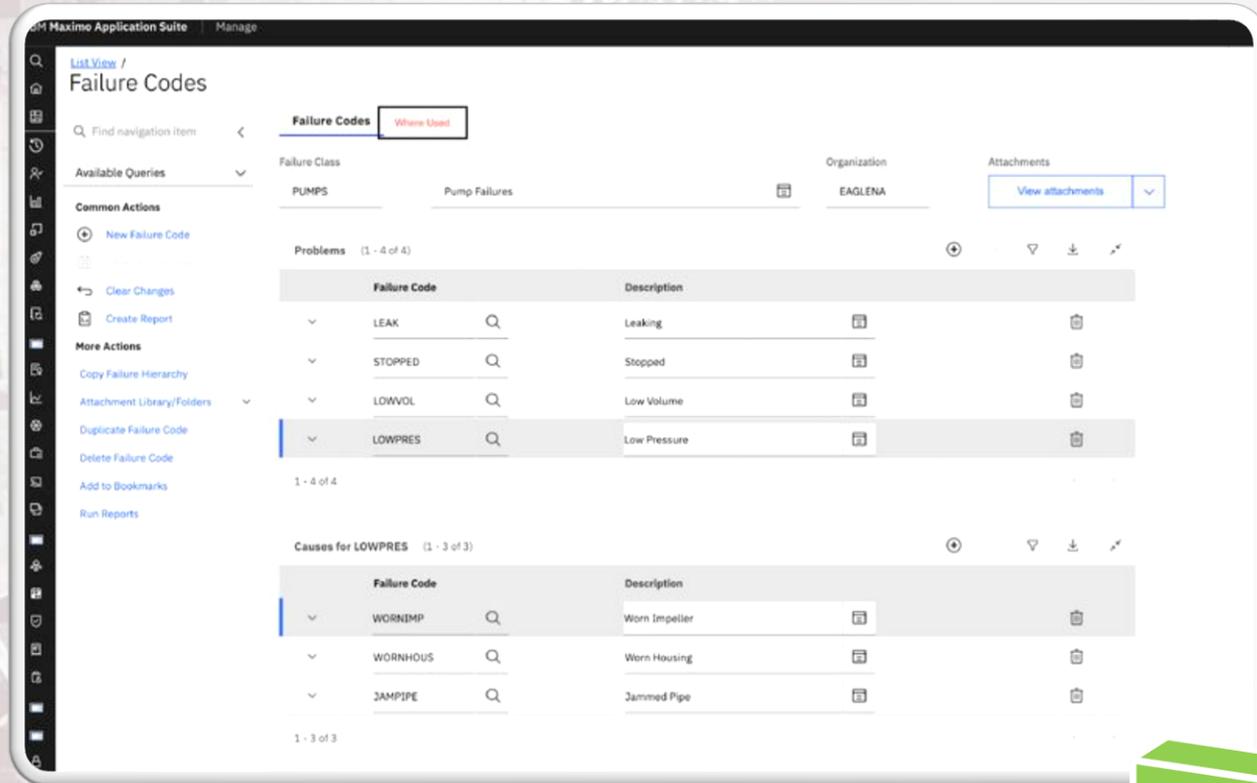
ERROR



# CREATE THE HIERARCHY



Once your classifications are identified lay out into a hierarchy then applied to a work order, asset, location or inventory





# Where It Happens in Maximo

Failure codes live in the **Work Order Tracking** application within IBM Maximo Application Suite. When a technician completes or closes a work order, the Failure Reporting tab is where the magic — or the missed opportunity — happens.

## The Capture Point

- Work Order → Failure Reporting tab
- Triggered at completion or close
- Links to asset's failure class hierarchy
- Fields: Problem, Cause, Remedy

## The Non-Negotiable

"If it's not captured here, it doesn't exist."

Your CMMS is only as smart as what your team puts into it. A perfect taxonomy means nothing if the field stays blank. Configuration, training, and process design all lead back to this one moment of capture.

Work Order Tracking

> **Work Order** Plans Assignments Related Records Actuals Safety Plan Log Data Sheet **Failure Reporting** Specifications Service Address Map

Work Order: 2001 Ventilation Fan - Check-out Noise

Location: SHIPPING Shipping and Receiving Department

Asset:

Configuration Item:

WO Total Work Units:

Parent WO:

Classification:

Class Description:

Launch Entry Name:

Site: BEDFORD

Class: WORKORDER

Work Type: CM

GL Account: 6500-350-??

Failure Class:

Problem Code:

Storeroom Material Status:

Direct Issue Material Status:

Work Package Material Status:

Attachments: View attachments (1)

Status: INPRG

Status Date: 6/30/25 8:40 PM

Inherit Status Changes?

Accepts Charges?

Is Task?

Under Flow Control?

Suspend Flow Control?

Flow Action:



## Work Order Tracking

> Work Order Plans Assignments Related Records Actuals Safety Plan Log Data Sheet **Failure Reporting** Specificatio

Work Order  
2001

Ventilation Fan - Check-out Noise

Site  
BEDFORD

Status  
INPRG

### Failure Details

Failure Class  
FAN > Fan overheating and not circulating

Remarks

Failed Date  
2/19/26 9:39 AM

Remark Date

Failure Codes (1 - 3 of 3)

Select Failure Codes

Type	Failure Code	Description		
PROBLEM	FAN	Fan overheating and not circulating		
CAUSE	DAMAGE	Damaged		
REMEDY	FUSE	Repair/Replace Fuse		

1 - 3 of 3

## The Capture Point

- Work Order → Failure Reporting tab
- Triggered at completion or close
- Links to asset's failure class hierarchy
- Fields: Problem, Cause, Remedy

## The Non-Negotiable

"If it's not captured here, it doesn't exist."

Your CMMS is only as smart as what your team puts into it. A perfect taxonomy means nothing if the field stays blank. Configuration, training, and process design all lead back to this one moment of capture.



# The 15-Second Rule

"If completing a failure code takes more than 15 seconds, it won't happen consistently — no matter how many times you ask."

Adoption is an engineering problem, not a discipline problem. The solution is to design your taxonomy and Maximo configuration so that correct coding is the path of least resistance. Three tactics that work:

**1**

## Limit Choices

Cap each level at 5–8 options per parent node. Maximo's cascading hierarchy is your friend — use it to filter ruthlessly so techs only see what's relevant to their asset.

**2**

## Smart Defaults

Configure Maximo work order templates and job plans to pre-populate the most probable failure class for recurring work types. Less cognitive load = better compliance.

**3**

## Shrink "Other"

Every "Other" selection is a lost data point. If your techs are choosing it more than 10% of the time, your taxonomy isn't specific enough — expand it, don't chase compliance.



# Guardrails Without Bureaucracy

Good governance isn't about locking everything down — it's about setting up the game, so the right people make the right calls at the right time. Failure code governance in Maximo only needs three things to function: clear ownership, a lightweight review cadence, and one hard threshold that tells you when the taxonomy is failing you.



## Who Codes

Technician selects at close. Planner or supervisor spot-checks weekly. Reliability engineer owns taxonomy design and dictionary changes.



## When to Review

Monthly 30-minute review of top failure codes. Quarterly taxonomy audit. Annual full hierarchy refresh tied to equipment strategy review.



## The Hard Threshold

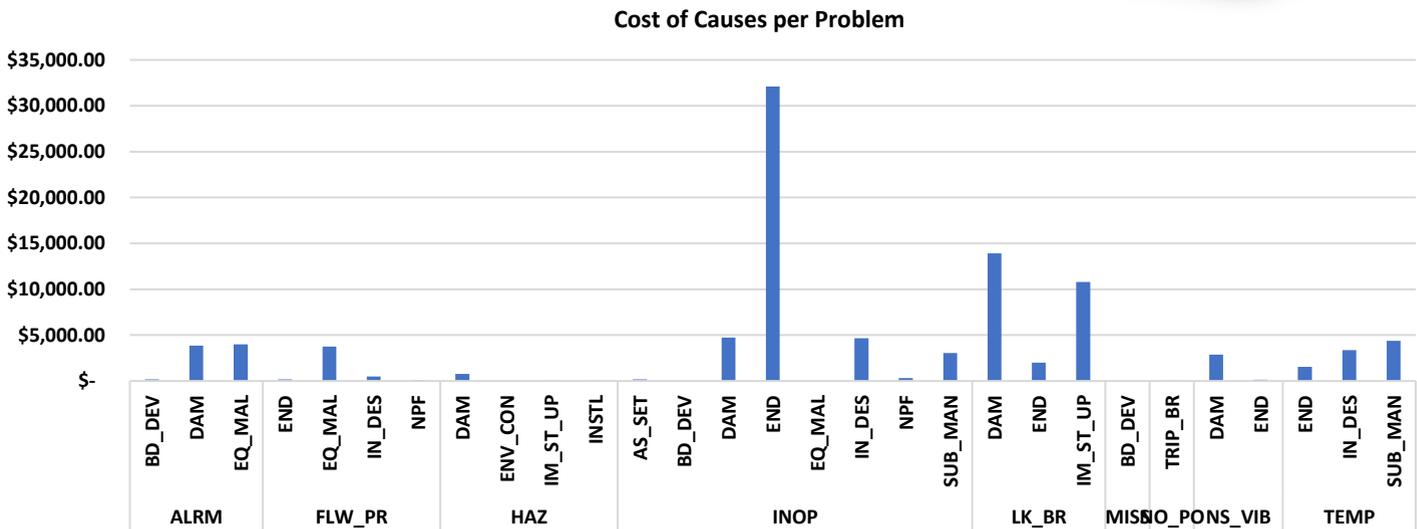
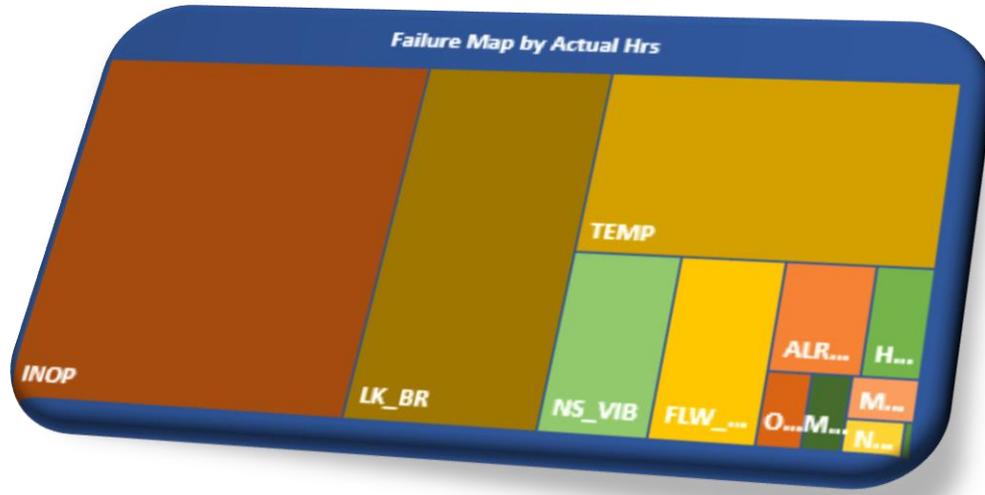
**"Other" exceeding 10–15% of selections** is a taxonomy failure signal — not a discipline problem. Expand the hierarchy; don't punish the techs.



# Repeat Failures: Catching Bad Actors

Repeat failures are the most expensive form of noise in any maintenance program. IBM Maximo Application Suite lets you query for assets with the same failure code appearing within a configurable window — say, 30 or 60 days. That query is your bad actor list. These assets are costing you more than parts and labor; they're costing you credibility.

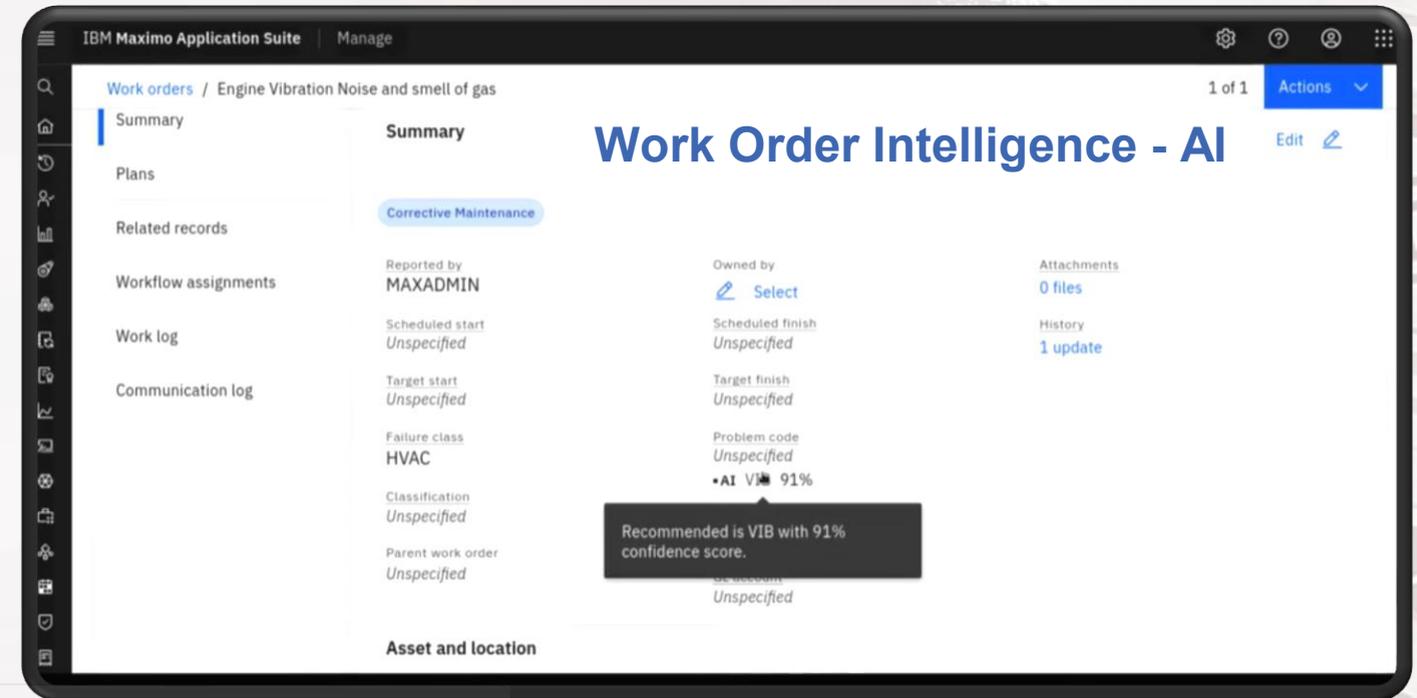
- 1 Day 0**  
Work Order closed. Seal leak coded. Remedy: replaced seal.
- 2 Day 18**  
Same asset. Same failure code. Second seal replacement in 18 days. Bad Actor flag triggered.
- 3 Day 19**  
RCFA initiated. Cause identified: installation error, not a parts quality issue.
- 4 Day 30**  
Installation standard updated. Job plan revised in Maximo. Warranty claim filed. Problem eliminated.





# FMEA Meets Maximo: Living Data

Reliability engineers invest serious time in Failure Mode and Effects Analysis. But most FMEAs live in a spreadsheet, reviewed once and shelved forever. IBM Maximo Application Suite changes that — when your failure code taxonomy mirrors your FMEA structure, every closed work order updates the living FMEA automatically.

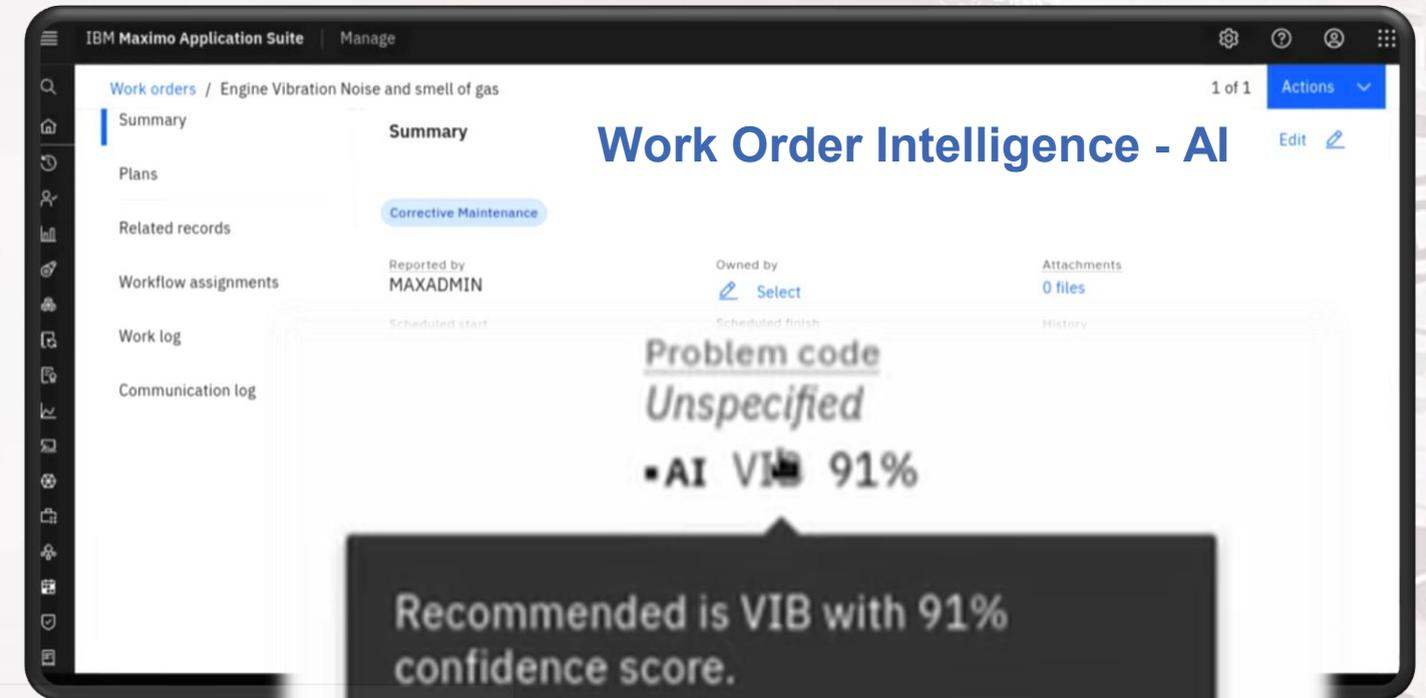


FMEA Element	Maximo Field	Location in MAS	Action Driver
Function / Asset	Asset / Location	Work Order Header	Identifies the system at risk
Failure Mode	Problem Code	Failure Reporting Tab	Triggers PM or RCFA review
Cause	Cause Code	Failure Reporting Tab	Drives job plan revision
Control / Mitigation	Remedy + PM / Job Plan	Job Plans, PM Records	Validates or updates strategy



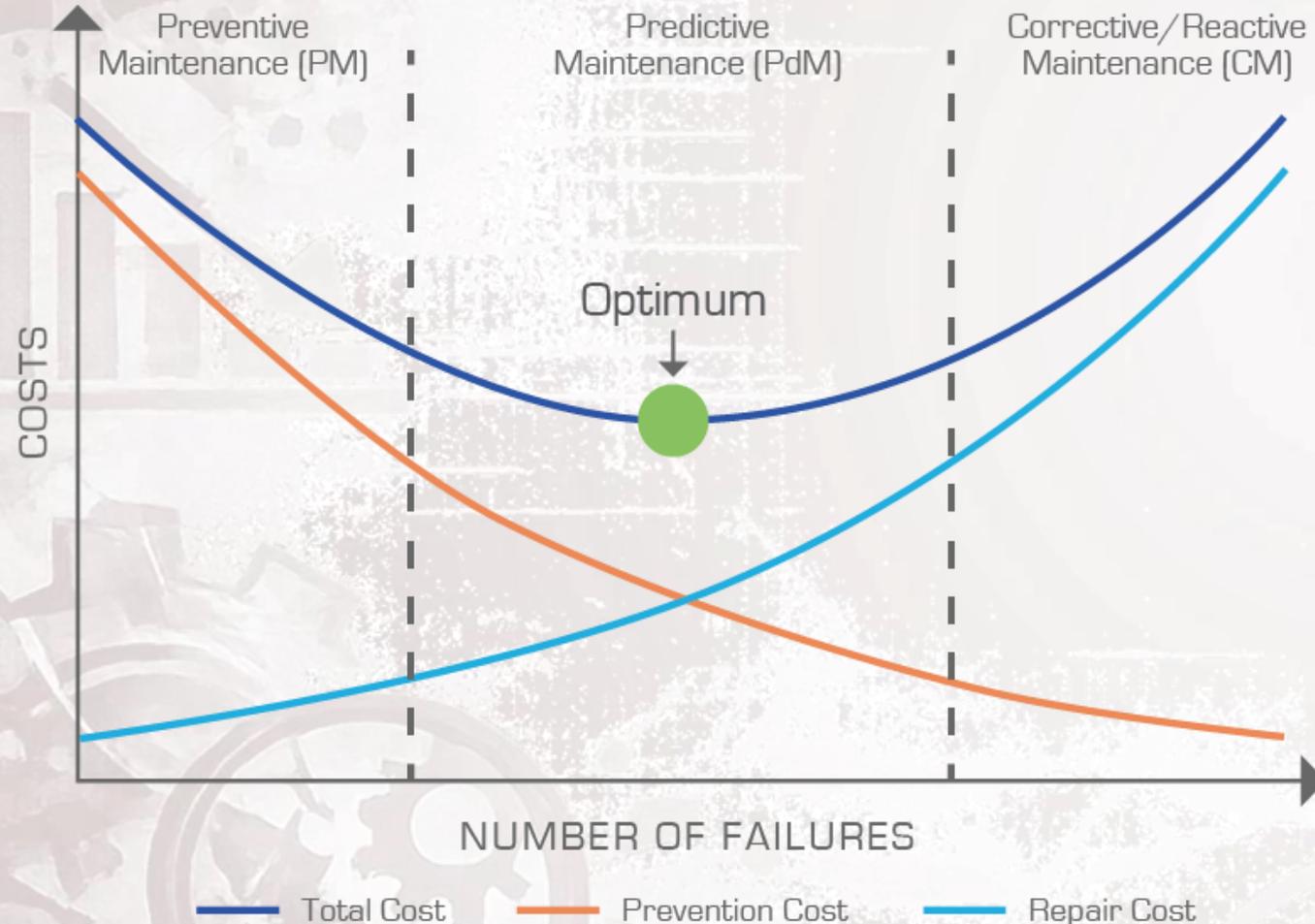
# FMEA Meets Maximo: Living Data

Reliability engineers invest serious time in Failure Mode and Effects Analysis. But most FMEAs live in a spreadsheet, reviewed once and shelved forever. IBM Maximo Application Suite changes that — when your failure code taxonomy mirrors your FMEA structure, every closed work order updates the living FMEA automatically.



FMEA Element	Maximo Field	Location in MAS	Action Driver
Function / Asset	Asset / Location	Work Order Header	Identifies the system at risk
Failure Mode	Problem Code	Failure Reporting Tab	Triggers PM or RCFA review
Cause	Cause Code	Failure Reporting Tab	Drives job plan revision
Control / Mitigation	Remedy + PM / Job Plan	Job Plans, PM Records	Validates or updates strategy

# ELIMINATING THE DEFECTS



Once the failures are identified, you'll need to distinguish what type of work aids you best in eliminating your defect. This can be done in a few different ways based on the results to the left.

The objective is to meet your organization mission and AIM to determine the best course of action.

$$\text{Return on Assets (ROA) Formula} = \frac{\text{Net Income}}{\text{Average Total Assets}}$$



# Your 30-Day Starter Plan

You don't need a 12-month transformation project. You need one team, one system, one failure class, and the discipline to code every work order for 30 days. That's enough to generate your first Pareto, find your first bad actor, and make your first data-backed reliability decision. Then you scale.

## Week 1

Pick 1 critical system. Build or refine its failure class in the Maximo Failure Codes application. Cap it at 10 codes maximum. Configure the work order template to default the failure class.



## Week 2–3

Brief your techs — 15 minutes, not a full training day. Show them the 15-second path to a correct code. Spot-check completions daily. Coach, don't police.



## Week 4

Run 3 reports in Maximo: Failure Pareto, Repeat Failures (same code, 30 days), and PM Effectiveness. Share results with your team. Let the data do the talking.

"Codes turn history into strategy. Every work order is a vote — make sure your data is voting for the right outcomes."

## LEARNING OBJECTIVES

### IDENTIFY FAILURES



**Identify standard**  
high level failure modes

### AWARENESS OF FAILURES



**Create an awareness of**  
what failures are and  
how to work toward  
eliminating them

### SITUATION AWARENESS



**Develop situational asset**  
**management** awareness of  
why something may fail

### ALIGN PROCESSES



Align your business  
processes to better facilitate  
elimination of defects found  
in your asset portfolio



# THANK YOU!



**Michael Guns, Jr.**  
**Director, Asset Strategy & Innovation**

[michael.guns@mavenasset.com](mailto:michael.guns@mavenasset.com)

[www.mavenasset.com](http://www.mavenasset.com)