Offshore Technology Conference 2026

OTC-37014-MS

Bridging the Gap of Smart Asset Management: Key Insights and Challenges from a 5-Year Journey Across 12 FPSOs

Leonardo Carvalho1, Manoela Boff1

1Shape Digital, Rio de Janeiro, RJ, Brazil

Abstract

Implementing effective digital asset management in complex offshore environments remains a significant challenge for the energy sector. (+++) This paper presents insights from a five-year deployment of digital predictive maintenance across 12 FPSOs off the Brazilian coast. Utilizing a Design Thinking for Innovation (DTI) framework, the study analyzes the program's maturation through three distinct phases: the initial deployment of generic anomaly models, the integration of FMEA-based user journeys, and recent augmentation via Large Language Models (LLMs). The system now monitors over 1,200 critical assets through 24,000 sensors, supporting 50 active weekly users. Results from the 2024 operating year indicate the program contributed to $14 million in potential production savings. Furthermore, the deployment of LLMs to synthesize engineering data has reduced root cause analysis time by approximately 40%. These findings provide a practical, non-theoretical framework for operators seeking to transcend basic technology implementation and achieve sustainable, AI-driven operational lifecycles.

Comentários:  
tirei os nºs das seções, não são usados nos artigos da otc. O alinhamento tbm nos paragrafos é bem à esquerda, conforme template la embaixo. Leo: Boa Manu!

Textos em amarelo: falta colocar informação específica

Introduction

Body text 1 paragraph.

* Relevance of Asset Management (ISO 55000) in heavy asset industries
* Condition Based Maintenance as opportunity of high impact with low investments
* Novelty subject with dynamic environment, new technologies emerging and no clear standardize way to achieve all CBM potential. (companies still learning about the subject)
* Buy tech versus Buy Value
* Design thinking methodology – people first

-----------------------

The offshore oil and gas industry faces increasing pressure to optimize operational efficiency, extend asset lifecycles, and reduce maintenance costs while adhering to strict safety standards. In response, digital transformation and predictive maintenance have become strategic imperatives for Floating Production Storage and Offloading (FPSO) operators. While the potential benefits of predictive maintenance and advanced analytics for offshore production systems are well documented, large-scale implementation in operational environments remains challenging. Studies such as Pandya et al. (2018) demonstrate significant gains in production efficiency through machine learning failure prediction; however, these gains are often achieved in well-scoped use cases and do not necessarily translate into sustained value when deployed across complex offshore assets. Industry reports suggest that a significant percentage of digital initiatives fail to deliver sustainable ROI, often not due to technology failures, but due to a lack of integration with operational workflows and user needs.

Current literature provides extensive research on algorithm development, sensor reliability, and prognostics techniques for condition-based maintenance. Comprehensive reviews by Jardine et al. (2006) and Peng et al. (2010) highlight the maturity of anomaly detection and failure prediction methods, particularly for rotating and critical equipment. However, there is a notable scarcity of longitudinal case studies that document the evolution of these systems in real-world environments. Specifically, there is limited guidance on how to bridge the gap between raw data generation and actual decision-making by frontline engineers. Without this "human-in-the-loop" focus, predictive systems often result in "alarm fatigue" rather than actionable insights.

This paper addresses this gap by presenting a five-year case study of a user-centered predictive asset management program deployed across 12 FPSOs on the Brazilian coast. Unlike purely technical implementations, this project utilized a Design Thinking for Innovation (DTI) framework to treat asset management as a user-experience challenge as much as an engineering one. We analyze the maturation of this program through three distinct evolutionary phases:

* the initial deployment of generic anomaly detection models;
* the pivot to FMEA-based user journeys to align alerts with failure modes;
* the current integration of Large Language Models (LLMs) to augment decision-making

By sharing the specific decisions, failures, and lessons learned from this journey—which culminated in over $14 million in production savings in 2024—this paper aims to provide practicing engineers and asset managers with a validated roadmap for navigating their own digital transformations.

Statement of Theory and Definitions

Body text 1 paragraph.

* Definition of the development approach (DTI) – focus on people/user . "What is? What if? What wows? What works?"
* Definition of Asset Management / CBM and CBM program goals (anticipate failures that impact in vessel uptime and safety)
* Metrics used to guide the program. (WAU + Success Cases + Value Generated + Precision)
* Method to quantity value generated (based on past failure events – 3 scenarios)

This section outlines the theoretical foundations, definitions, and evaluation criteria that guided the design, implementation, and assessment of the predictive asset management program presented in this paper. The intent is to establish a common conceptual framework to support the analysis of the program’s evolution across its different maturity phases.

Design Thinking for Innovation

The User-Centered Predictive Asset Management Program presented in this paper is fundamentally grounded in a Design Thinking for Innovation (DTI) methodology, with a strong emphasis on understanding and supporting the needs of end users. In the context of offshore asset management, predictive analytics was treated not as standalone technical solution, but as a mean to improve how engineers, maintenance planners and operations teams make decisions under time and safety constraints.

The DTI approach was structured around four iterative stages. The What is? stage focused on developing a deep understanding of current operational practices, including how maintenance decisions are made, which information is trusted by users, and where friction, uncertainty, or delays occur in day-to-day workflows. This phase revealed that many predictive solutions failed not due to insufficient analytical capability, but because they did not align with how users diagnose failures or prioritize actions.

Building on these insights, the What if? stage explored alternative ways of delivering predictive information that could better support user decision-making. Rather than asking what additional data could be generated, this stage focused on how insights could be framed to answer the specific questions users face when responding to abnormal equipment behavior. Multiple analytical and workflow hypotheses were evaluated in collaboration with end users before implementation.

The What wows? stage emphasized rapid validation of solutions that demonstrably improved user experience and decision confidence. At this stage, solutions were considered successful only if they reduced cognitive load, increased trust in predictive insights, or accelerated diagnostic workflows. User feedback and early adoption indicators were used as primary success criteria, often taking precedence over incremental improvements in model accuracy.

Finally, the What works? stage addressed scalability, robustness, and long-term sustainability. Solutions validated in earlier stages were standardized, integrated into existing asset management processes, and assessed based on their ability to generate measurable operational and financial value. This stage ensured that user-centered innovations could be consistently replicated across assets and FPSOs, forming the basis for the framework’s progressive maturation over time.

Asset Management and Condition-Based Maintenance Definitions

In alignment with ISO 55000 principles, asset management in this program is defined as the coordinated activities required to realize value from physical assets while balancing performance, risk, and cost. Within this context, Condition-Based Maintenance (CBM) is treated as a strategic enabler for improving FPSO uptime, operational safety, and maintenance efficiency by anticipating failures before they result in unplanned downtime or safety-critical events.

The scope of the CBM program focused on critical production and utility assets whose failure modes have a direct impact on vessel availability, production deferment, or personnel safety. Rather than attempting to monitor all assets uniformly, the framework prioritizes assets based on failure consequences and operational criticality, ensuring that analytical efforts are concentrated where decision support yields the highest value.

Program Metrics and Adoption Indicators

Recognizing that analytical performance alone is insufficient to assess program success, a set of complementary technical, adoption, and value-based metrics was defined to guide decision-making throughout the program. These metrics were intentionally selected to reflect not only model performance, but also user engagement and decision impact across the different maturity phases of the framework.

User adoption was monitored using Weekly Active Users (WAU), which provided a consistent indicator of whether predictive insights were being actively consumed by engineers and maintenance teams. In parallel, documented success cases were tracked to capture qualitative and quantitative evidence of decisions influenced by predictive insights, serving as a direct link between analytics and operational action. To evaluate business impact, value generation was assessed based on the contribution of predictive insights to avoided production losses and optimized maintenance interventions.

To enable consistent comparison across the three maturity phases, alert precision was defined as a primary technical performance metric. In this context, precision is defined as the ratio between predictive alerts that led to a validated failure mode or justified maintenance action and the total number of alerts generated within a given period. Alerts were considered true positives only when they resulted in confirmed abnormal equipment behavior with operational relevance, as validated by engineering analysis, maintenance intervention, or observed failure progression. This definition was deliberately chosen to reflect the operational usefulness of predictive insights rather than purely statistical model accuracy. By focusing on decision-relevant precision, the metric captures the extent to which predictive outputs effectively support engineering judgment and maintenance prioritization. While this approach does not fully represent all aspects of model performance, it provides a pragmatic and user-centered basis for evaluating progress and trade-offs as the framework evolved from model-centric to decision-centric implementations.

Together, these metrics formed a feedback loop used to refine models, workflows, and user interfaces across successive iterations of the framework, ensuring that increases in analytical sophistication were consistently aligned with higher user trust, improved decision quality, and measurable business outcomes.

Method for Quantifying Value Generation

In order to quantify the value generated by the program over the years, a methodology was established to estimate the financial impact of the potential failure avoided cases, called internally as “success case”. This methodology considers the potential production losses, corrective maintenance costs and other penalties applied to the charter contract business model (client specific) to generate three financial impact scenarios, (1) Best, (2) Intermediate and (3) Worst Case failure scenarios.

Each of the scenarios are evaluated based on historical 4 years registry of production downtime and corrective maintenance costs that match the respective failure mode and equipment type of success case. For example, an incipient vibration increase due to a damaged coupling disk pack in a water injection centrifugal pump have its intermediate scenario valuated by a registered historical case where this failure mode [equipment trip due to high vibration] to this (or similar) pump application has happened and costs of unplanned coupling replacement already registered in Computerized Maintenance Management System (CMMS).

This approach gives a rational and close to deterministic approach to evaluating the impact of events of failure that were avoided by the predictive maintenance approach. It is known that evaluate impact of situations that did not actually happened is challenging and susceptible to errors and interpretations, but this methodology was sufficient to achieve consensus among stake holders interested in the program and has proven a practical approach to measure improvements over time. The financial impact quantification could be much higher if the methodology is adapted to consider oil production losses costs instead of just charter diaries.

Phase 1: The initial deployment of generic anomaly detection models

The smart asset management digital program started in 2019 as an internal digital initiative with the objective to drive impact in maintenance costs and reduction of penalties applied by unplanned operational downtime. The strategy was to leverage time-series data from a centralized Plant Information Management System (PIMS) applied to 8 FPSOs to anticipate potential failures events by alerting onshore engineering teams about incipient issues, allowing them to take actions proactively that would avoid unplanned trips and non-planned corrective maintenance. Examples of action are suggesting offshore operations team perform a changeover and further inspection, triggering the acquisition of a spare part or changing a process parameter that could impact the failure mechanism or extend the time to trip until a more adequate moment.

“what is?”: Prior the program initiation, the centralized PIMS has already allowed onshore engineers to look proactively at potential issues by manually evaluating the time-series sensors data on a daily basis and using each engineer own knowledge to identify potential issues. This approach, shown in Fig. 1., has proven to be beneficial since it has allowed few failures to be anticipated, which proves the potential of such digital program. But this approach presented several structural limitations:

* Manpower Intensive: Daily/weekly evaluation of sensors trends demands significant manpower from engineers and has a low impact to effort ratio, once most of the time no anomaly is found.
* Lack of standardization: The evaluation depends on the prior knowledge of the engineer performing the evaluation, which create discrepances between junior engineers with low expertise and senior engineers that can easily identify abnormal situations.
* Time delay: Evaluations performed periodically allowed anomalies to remain undetected for a couple of days until the next manual review cycle.

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Fig. 1— Baseline “What is?” operational workflow for condition monitoring prior to the implementation of automated anomaly detection.

“What if?:” The hypothesis that base the program was that it will be possible to use machine learning and data science techniques to identify automatically the deviations that the specialized engineers already were ablet o detect by manual evaluation, but reducing the limitations presented above. Such approach could also contribute to identifying more complex anomaly scenarios that even engineers could not detect with the existing visualization tools.

Considering this hypothesis, Fig. 2 introduces the “What if?” scenario, in which anomaly identification is automated through data-driven models. What if the engineers responsible for condition monitoring receive alerts about anomalies occurring in real time in the critical assets, allowing them to stop or reduce manually evaluating time series data and receive specialized suggestions of anomalies to evaluate. This approach has the potential to drastically reduce the manpower required in the “What is” state while increase the issues detected and processes by the engineers.

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Fig. 2— Baseline “What if?” operational workflow introducing data-driven anomaly detection.

To achieve this new scenario, the project planned to identify through interviews with end users the most promising applications of the automatically detection approach, identifying and prioritizing scenarios that have high impact and relatively high frequency of occurrence, which will be used to validate the hypothesis.

With prioritized detection scenarios, the project team were able to map the respective sensors data for each scenario, replicate it to all prioritized equipment and start experimenting the data-based solutions. This lead to a scenario of:

* TODO: Add picture of data flow/ user journey considering the topology (SENSORS -> MODELS -> ALARM). The user journey stop here at this phase. Only the models knows about the sensors. Models are complete entities that generate alarms, equipment is not an entity itself.
* TODO: List the selects “detection scenarios” deployed at the time. Couling, scalling,

“What wows?”: Based on back tests applied on historical data, the results seem to be promising. Once validated with the end user, the models deploy started and the strategy was replicated to 8 vessels, 131 equipment and 171 models and a simple alert mechanism was stablished to issue an outlook e-mail for the user when a new anomaly situation happened.

* TODO: print of a e-mail from that time alerting to a failure mode.

After couple of months of operations, few success cases started to be identified, and the action loop could be tests from the model automatically detected through an action to be taken by offshore team. At this moment the hypothesis was validated and the “wow moment” occurred.

TODO: add slide about the case

“what works?” After 1 year of running the solution with this approach and applying the DTI principles to collect feedback from the end users and addressing his concerns, several opportunities for improvements were perceived. The most relevant ones are presented below.

TODO: add metrics of the end of this phase (aseets monitored, success cases impact value, precision, wau)

TODO: add metrics recall?? (get from RERT report)

1. **False Alarms:** After replicating the solution to thousands of detection scenarios, the volume of false alarms, measured through Precision, became a major concern among end users. While success cases confirmed the value of this approach, frequent false alarms undermined the user confidence and consumed valuable engineering time. For each alarm evaluation the engineer needs to dedicate a few hours, and the time spent on this evaluation were sometimes increasing the manpower required for condition monitoring, instead of reducing it as the initial hypothesis.
2. **High Diagnosis Effort:** A mapping of the end user journey highlighted that the detection of the anomaly is the trigger for a second and much more time-consuming step that is perform the diagnostics of the anomaly to substantiate an action decision. Depending on the format where the alarm anomaly is presented, it could demand much more time from the user to put together all information required for the diagnosis, turning him to individual trends plots files and several data collection tasks.

This high effort to perform diagnosis alongside the high quantity of false alarms were starting to become major challenges into scaling such kind of solution for a large quantity of assets and should be addressed in order to achieve long-term sustainable results. It was established that would be unviable to expand the quantity of detection scenarios without addressing those major issues.

During Phase 1, the solution architecture positioned the anomaly detection model as the central element of the condition monitoring process. Each scenario was implemented as an independent analytical model, responsible for ingesting sensor data and generating alarms when deviations from the expected behavior were identified. In this model-centric approach, assets and failure modes were implicitly represented through model logic, rather than being explicitly structured as entities within the system.

As the program evolved, this architecture led to a rapid increase in the number of deployed models. A limit set of use cases resulted in the deployment of 171 models, generating approximately 984 alarms, with an overall alert precision of 3%. While this level of precision was sufficient to demonstrate technical feasibility and produce isolated success cases, it proved insufficient to support large-scale operational adoption.

From the user perspective, the information delivered did not align with how engineers reason about equipment health and failure risk. Users were less interested in knowing that a sensor was statically anomalous and more concerned with understanding wich equipment was at risk, which failure mode might be developing and what actions should be considered. The absence of this contextual information required significant manual effort for alert interpretation and diagnosis. These insights brought to light that scaling the program would not be viable without a fundamental shift away from a model-centric approach toward a user-centered, failure-oriented framework, motivating the transition to Phase 2.

Model as main actor

* 5% precision (get presentation with Jardel)
* 1 use case in a year – XX models --- XX alarms – XX precision
* What information user wants
* CHALLENGES / METRICS/ INSIGHTS

Phase 2: The pivot to FMEA-based user journeys to align alerts with failure modes

Following the validation of the Proof of Concept in Phase 1, specific challenges needed to be addressed to achieve scalability. The objective was to expand the scope from XXX assets and XX sensors to YYY assets and XXX sensors.

Initial attempts to expand scope revealed the necessity of a data contextualization layer. Because multiple models often needed to access the same sensor on the same equipment, the development team was forced to manually map sensors for each instance. This resulted in excessive hard-coded information, making it difficult to audit whether the correct sensors were being utilized.

Consequently, the team applied the DTI methodology, conducting interviews and workshops with condition monitoring stakeholders to establish new 'what-if' scenarios.

“what-if”: Contextualization of assets data. / Failure Mode Analysis to drive models’ development.

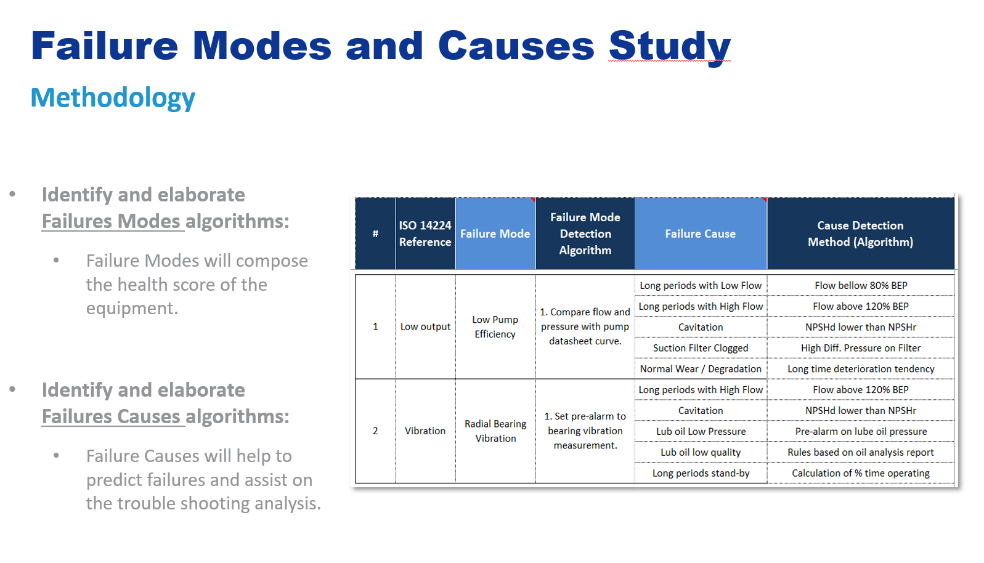
1. What if we apply a Failure Mode and Causes Analysis (FMECA) methodology with subject matter experts to better understand possible detection scenarios already known by engineers?
2. What if all data consumed by the models comes from a data organization layer that contextualizes data and serves it to models and for users?
3. What if the assets contextualized data and the FMECA analysis is used to reduce the effort of the user in the diagnosis phase?

Hypothesis 1: In order to address the major problem identified in phase one related to “low precision” (quantity of false alarms), it was observed by the user interviews that the proper definition of what should be a useful alarm was missing. This lack of clear definition creates a gap between what was being developed by the project and what will be useful for end-user.

A typical example is, an slightly increase in centrifugal pump vibration that the data driven model has not seen in training data was raised as an alarm. But after the subject matter specialist evaluation, it was reported that the increased vibration could be justified by a significant pump flow temporarily reduction induced by process requirement changes. In such case, besides the vibration has increased, it is not directly related to the identification of Vibration failure mode and should not be alarming.

To avoid this kind of situation, several workshops were conducted with subject matter experts to capture their knowledge and expertise on what kind of issues should be alarming and what information the alarm should contain to expedite their diagnosis phase.

FMECA methodology and workshops were strongly inspired in ISO 13379 series.

  
A diagram of a tree with roots

AI-generated content may be incorrect.

Hypothesis 2:

While Hypothesis 1 addressed the *quality* of the alerts, Hypothesis 2 addressed the *scalability* of the deployment. As the program prepared to expand from a pilot scope to fleet-wide monitoring, structural bottlenecks in the data architecture became apparent.

In Phase 1, the system utilized a "flat" architecture characterized by Direct Tag Binding. As illustrated in **Fig. 3 (A)**, analytical models were connected directly to raw sensor tags (e.g., 40\_TI\_1434.PV) within the historian database.

This approach presented significant friction during expansion:

* Developing a model for "Compressor A" required a data scientist to manually locate specific tags and hard-code them into the algorithm.
* Deploying the same model to "Compressor B" was not a simple replication. Because tag nomenclature often varies between assets (and even more so between different FPSOs), engineers had to manually map new tags (e.g., 40\_TI\_1433.PV) for every new instance.
* If a sensor was replaced and the tag name changed in the PIMS, the hard-coded link would break, causing model failure.

This linear relationship between engineering effort and asset count meant that scaling to thousands of assets would require an unsustainable amount of manual configuration.

To resolve this, the team tested the following "What if" scenario:

*“What if all data consumed by the models comes from a data organization layer that contextualizes data, decoupling the analytical logic from the raw database?”*

This hypothesis drove the transition to an **Asset-Centric Data Model**, introducing an abstraction layer between the raw data and the models, as shown in **Fig. 3 (B)**. This semantic layer functions through a two-step process:

1. Raw, cryptic tags are mapped once to a standardized "Asset Template." For example, tag 40\_TI\_1434.PV is mapped to the standard attribute "Radial Bearing Temp. DE" (Drive End) within the Compressor class.
2. The analytical models are re-written to ingest the standard attribute ("Radial Bearing Temp") rather than a specific tag number.

The implementation of the semantic layer fundamentally changed the deployment velocity. By decoupling the model logic from the data source, the program achieved:

* Models became "write once, deploy many." Once a model is defined for a class of equipment (e.g., Water Injection Pumps), it can be instantly instantiated across all assets that share that template, as the data mapping is handled upstream.
* Changes to raw tags are managed in the semantic layer, leaving the model logic untouched and robust.

This architecture shifted the developer focus. Prior to this layer, developers spent an estimated 80% of their time searching for and mapping tags. With the semantic layer, this time is reallocated to value-added tasks such as failure mode analysis and model tuning.

This architectural pivot was a prerequisite for Phase2, providing the structured data foundation necessary for the eventual integration of Large Language Models.

A diagram of a diagram

AI-generated content may be incorrect.A diagram of a model

AI-generated content may be incorrect.

A diagram of a model

AI-generated content may be incorrect.

Hypothesis 3: Web application to allow user to interate with data and use context information for diagnosis (contrast with phase 1 that the alarm was sent by email)

Introduce the lighthouse web application…

TODO: Add the “what is and what if” diagram highlighting the new core areas scope of the project. (Engineer diagnosis is now part of the scope).

“what-wow”: workshops, more detailed diagnosis from model, boom of knowledge based models (simpler data approach) – refer to ISO 13379 definintions

The implementation of the actions to validate the presented hypothesis rapidly proven efficiently by drastically impacting in the program results. In XX time it was possible to expand the approach from XX equipment’s to YY equipment’s with increase in the precision of the alarms.

During FMECA workshops, many opportunities mapped where able to be addressed by the usage of “Knowledge-Based” models (ISO 13379 reference) instead of just “Data-driven” models as phase 1 premises. “Knowledge-Based” are modeling strategies that uses engineering-based equations and pre stablished limits based on asset project design to raise the alerts. This kind of model proves to be a powerful strategy aligned with the data context layer to leverage success cases detection results.

At this time, both models approach started to call user attention to specifics failure modes and failure causes instead of generically alarming, this approach reduces the time of end-user engineer to evaluate the raise alarm and perform diagnosis because the objective of the detection is clear and based on a prior engineer study (FMECA).

The new application, when interface build to lavage the context data layer also played a important role in expedite the diagnosis step, because now all digital information related to the asset are compiled in a single place. Once it is much simpler to check assets sensors that are not particularly being user by the alarming model, as well as compare other alarms from other failure modes for the same asset in the same time.

It was also possible to start tracking user behavior on the application and turn it to metrics that support the DTI process workflow, understanding which information is mostly used by users in the diagnosis phase.

TODO: Include success metrics over time. WAU, ASSETS MONITORED, MODELS, PRECISION, VALUE IN SUCCESS CASES, QTY OF SUCCESS CASES

A screenshot of a computer

AI-generated content may be incorrect.

“what-works”:

What information user wants / how it will generate value to him

* FMEA based hypothesis
* CHALLENGES / METRICS/ INSIGHTS

Phase 3: The current integration of Large Language Models (LLMs) to augment decision-making

What is:

* Maintaining KB models are some times costly.. problems persist for long time and if constantly alarming it stops generatinge value as expected.
* The increase quantity of identified cases started to highlight bottlenecks futher in the process, related to corrective maintenance prioiritization, contanstly risk evaluation and optimization in planning. Those are the next step of the process that are included in phase 3 in order to maximize results. (identified failure modes not being fixed at a necessary time prior equipment trip or greater damage).
* \*\*The predictive alarms monitoring still needs human support as first step to “filter” false alarms and perform initial diagnosis. Is this step could by made by AI, the operational costs to run such workflow could dramatically reduce.

What if:

* Context layer starting to include more information as CMMS, cases, documents etc..
* Application interface and models not just for detection, but for support prioritization and planning.
* Lavarage **LLM** for alarms and context data.
* CHALLENGES / METRICS/ INSIGHTS

**Conclusions**

Body text 1 paragraph.

Description and Application of Equipment and Processes.

Body text 1 paragraph.

**Presentation of Data and Results**

Body text 1 paragraph.

**Conclusions.**

Body text 1 paragraph.

**Acknowledgments**

Body text 1 paragraph.

**References**

Jardine, A. K. S., Lin, D., & Banjevic, D. 2006. A Review on Machinery Diagnostics and Prognostics Implementing Condition-Based Maintenance. Mechanical Systems and Signal Processing, Vol. 20, No. 7, pp. 1483–1510.

Pandya, D., Goebel, K., Eklund, N., & Roychoudhury, I. 2018. Increasing Production Efficiency via Compressor Failure Predictive Analytics Using Machine Learning. Offshore Technology Conference (OTC), Paper OTC-28990-MS, Houston, Texas.

Peng, Y., Dong, M., & Zuo, M. J. 2010.Current Status of Machine Prognostics in Condition-Based Maintenance. Mechanical Systems and Signal Processing, Vol. 24, No. 2, pp. 282–293.

First-Level Heading

Body text 1 paragraph.

Body text 2 paragraphs.

Second-Level Heading.

Third-Level Heading.

Fourth-Level Heading.

**ATTENTION! DELETE AFTER READING.**

**The final layout of the manuscript will differ from your original submission.**

All manuscripts will be tagged into XML format for final publication. Standardized styles and fonts will be used in the final layout of the manuscript. Links will be added for references, figures, tables, and equations. All figures and tables with associated captions will be placed after the paragraph of first mention in the final layout of conference manuscripts and as near as possible to first mention in journal manuscripts.

Reminder: Styles have been built into the template. If you type directly over an element, the built-in formatting will be applied automatically. When adding additional text, highlight the new text and then select the appropriate style from the Home > Styles tab. Instructions for using the OTC Manuscript Template are available under the Templates heading on the SPE Author Resources webpage at <https://www.spe.org/en/authors/resources/>.

Fig. 1—Figure captions should begin with an overall descriptive statement of the figure followed by additional supporting text. Captions should be placed immediately after each figure. Figure parts are indicated with lower-case letters: (a) Part 1; (b) Part 2; (c) Part 3.

Table 1—Table captions should begin with a short description of the table. Format tables using the Microsoft Word table commands and structures. Do not create tables using spaces or tab characters. Large tables presenting rich data should be presented as separate Excel or .csv files, not as part of the main text.