# Achieving Three Nines for Product Data Monitoring Systems

Raghothama M. Rao[1], Travis Mann[2], and William E. Edinger[3]

[1,2,3]*Belcan Engineering, a Cognizant company, Windsor, CT, 06074, USA*

rrao@belcan.com
tmann@belcan.com
wedinger@belcan.com

## ABSTRACT

Effective data and health management are critical throughout the lifecycle of engineered systems. When implemented correctly, product health management tools and processes can help lower total product ownership costs, improve safety, maximize availability and utilization rates, thereby delivering value to system operators and maintainers. Central to achieving this capability is robust data management that includes data acquisition, secure transmission, efficient storage and timely processing. These steps ensure that health insights can be delivered to stakeholders in support of diagnostics, prognostics, and informed decision-making.

In sectors such as aerospace, defense and power utilities, the demand for high availability - targeting 99.9% uptime or "three nines" – places stringent requirements on Product Data Monitoring Systems (PDMS). Hardware infrastructure, software, engineering processes, and procedures are an integral part of achieving this target. This paper presents a focused exploration of the software, automation strategies, and best-in-class engineering processes that support high-reliability health data monitoring, with an emphasis on commercial aircraft engine applications. Drawing from Belcan Engineering's experience, we highlight key process improvements and practices that enable scalable and maintainable solutions. Additionally, we discuss how early integration of health management capabilities into the development cycle enhances product value and reduces lifecycle costs. The insights presented are based on real-world implementations and are intended to guide practitioners seeking to design and operate resilient, high-availability monitoring systems.

## 1. INTRODUCTION

In today's world, where data-driven decision-making is essential to ensuring safety, efficiency, and competitiveness of the aerospace and other regulated industries, Product Data Monitoring Systems (PDMS) have emerged as a critical capability. These systems are designed to continuously collect, process, and analyze data from complex engineered assets—such as aircraft engines, avionics systems, and mechanical subsystems—to assess health, predict failures, and enable proactive maintenance.

PDMS play a pivotal role in ensuring operational readiness, fleet availability, and safety compliance across the aerospace lifecycle—from development through sustainment. Their success directly affects mission assurance, maintenance planning, and ultimately, customer satisfaction. Given this criticality, the availability and reliability of these systems themselves become non-negotiable. A monitoring system that is offline or degraded undermines the very insights it is built to deliver. It is important to distinguish between high availability and fault tolerance. High Availability focuses on minimizing downtime, using strategies like rapid fault detection, automated restarts, and quick incident resolution. A key metric here is Mean Time to Recovery (MTTR). Fault Tolerance, in contrast, aims for zero downtime by enabling systems to operate through component failures without user-visible impact. Depending on the criticality of the service tier (e.g., real-time alerts vs. batch reporting), systems may adopt high availability or full fault tolerance designs accordingly.

PDMS availability is defined as the proportion of time a system is in a functioning condition and ready for use, typically expressed as a percentage of total operational time. Availability can be expressed as

$$Availability\ (\%) = \frac{Uptime}{Uptime+Downtime} \text{ x } 100 \qquad (1)$$

The term "Three Nines" refers to a system uptime of 99.9%, a widely recognized benchmark for high availability in critical infrastructure. While seemingly modest, this level of reliability translates to no more than 8.76 hours of downtime per year, or just over 10 minutes per week. In the context of Product Data Monitoring Systems (PDMS) for aerospace applications—where the consequences of system unavailability include missed fault detection, disrupted

maintenance planning, and increased operational risk—achieving this benchmark is both technically challenging and mission-critical.

This paper explores the journey toward achieving "Three Nines" for PDMS in the aerospace industry, grounded in lessons learned from previous implementation iterations. Through retrospective analysis, we examine the architectural, operational, and organizational changes that contributed to availability improvements across system generations. These insights are particularly timely as OEMs and operators increasingly look to scale predictive maintenance, comply with regulatory demands, and unlock value through fleet-wide data integration.

Furthermore, we position this work within a broader cross-industry conversation. Other safety-critical domains—such as automotive, healthcare devices, and industrial automation—face similar challenges in achieving resilient, always-on health monitoring. However, the unique operational constraints of aerospace systems, including power loss, intermittent connectivity, extreme environmental conditions, and rigorous certification requirements, amplify the complexity.

The aerospace ecosystem is increasingly reliant on real-time product data monitoring systems to support predictive maintenance, operational decision-making, and regulatory compliance. These systems require seamless integration across stakeholders—flight operators, OEMs, and suppliers. However, suppliers, who are responsible for providing component-level data and/or engineering services, often face structural and operational limitations that challenge their ability to meet the stringent availability requirements demanded by the industry. While flight operators and OEMs often have the infrastructure and resources to meet the benchmark of 99.9% availability, suppliers operate under a distinct set of constraints that challenge their ability to contribute effectively towards achieving "Three Nines".

Suppliers are frequently constrained by factors outside their control, including limited access to advanced infrastructure, legacy system dependencies and resources with legacy knowledge. The diversity of platforms and data formats used by operators and OEMs create significant barriers to integration and interoperability. A significant number of stakeholders continue to rely on legacy systems that lack modern APIs, telemetry capabilities, and scalability. These systems are costly to maintain, difficult to integrate, and prone to failure, making them ill-suited for high-availability environments.

Operating under some of the above-mentioned constraints, reviewing previous iteration results and tracing the evolution of monitoring system performance, this paper aims to provide both a technical blueprint and a strategic framework for aerospace organizations pursuing Three Nines availability. Ultimately, this pursuit is not just about system uptime—it's about ensuring that critical health insights are always available when and where they are needed, supporting safe and sustainable aerospace operations.

## 2. LITERATURE REVIEW

The pursuit of high availability in monitoring systems has been extensively studied across domains such as cloud computing, telecommunications, and mission-critical embedded systems. However, literature specifically addressing 99.9% uptime targets for Product Data Monitoring Systems (PDMS) in the aerospace sector remains sparse and fragmented. This section reviews key contributions from adjacent disciplines, industry standards, and recent aerospace-focused publications to frame the context for this study.

### 2.1. High-Availability Systems in Industry Contexts

In the broader reliability engineering literature, availability is defined as the proportion of time a system is functional and accessible, typically expressed as a percentage. The concept of "nines of availability" (e.g., 99.9%, 99.99%) is well established in cloud and IT infrastructure domains (Weber et al., 2010; Beyer et al., 2016), where fault-tolerant design patterns, horizontal scaling, and automated recovery workflows are extensively applied.

In contrast, cyber-physical systems like aircraft PDMS operate under unique constraints - including hardware redundancy limitations, strict certification requirements, and limited network access - requiring customized high-availability strategies (Koopman, 2003). These differences render direct application of cloud-native practices insufficient without adaptation.

### 2.2. Aerospace-Specific Monitoring System Challenges

Research from government agencies and aerospace OEMs underscores the increasing reliance on onboard health monitoring systems for operational decision-making, including engine health management (EHM), structural health monitoring (SHM), and condition-based maintenance (CBM) (FAA AC 43-216, 2017; Seshadri and Krishnamurthy, 2017; Roy et al., 2017). Several studies highlight challenges related to data fidelity, system integration, and real-time telemetry, which directly affect monitoring system availability (Li et al., 2020).

Moreover, the reliability of the monitoring system itself has emerged as a risk factor. Research by Saxena et al., (2008) emphasizes that unavailability of PHM capabilities can lead to missed failure precursors, maintenance delays, and reduced fleet readiness. However, few published works quantify system-level availability or tie reliability metrics to architectural evolution.

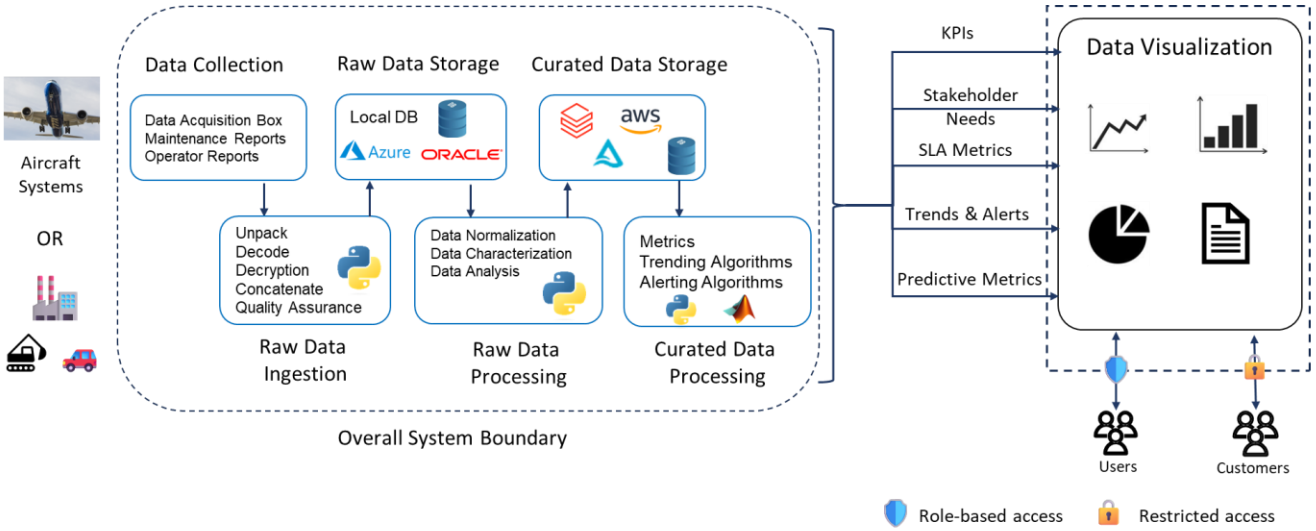


Figure 1. Product Data Monitoring Systems Data Flow

### 2.3. Availability Metrics and Modeling Techniques

Literature from the dependability and safety engineering community provides a robust foundation for modeling system availability. Key techniques include Markov models, fault tree analysis, and reliability block diagrams, which are commonly used in Failure Modes, Effects and Criticality Analysis (FMECA) and reliability-centered maintenance (RCM) frameworks (IEC 61025, MIL-STD-1629A). While these methods are applied to aerospace subsystems, they are not always extended to the monitoring systems themselves.

Recent trends in Prognostics and Health Management (PHM) research, particularly within the PHM Society, explore integration of machine learning for anomaly detection and fault prediction (Vachtsevanos et al., 2006; Saha & Goebel, 2009). However, these works often emphasize prediction accuracy over infrastructure reliability—highlighting a gap in addressing end-to-end system uptime.

### 2.4. Standards and Guidelines

Regulatory guidance such as RTCA DO-178C, SAE ARP4754A, and RTCA DO-326A inform the development of safety-critical software and cybersecurity for onboard systems but provide limited specificity regarding real-time system availability or PDMS-specific uptime metrics. Emerging standards in PHM—such as FAA AC 43-218, 2022; IEEE 1856; and ISO 13374—begin to address data management and diagnostics, yet do not enforce availability thresholds.

Additionally, SRE (Site Reliability Engineering) practices from the software industry offer process-centric tools such as Service Level Indicators (SLIs) and Objectives (SLOs), which can be adapted to aerospace telemetry and ground station reliability (Beyer et al., 2016). Still, there is limited published work on tailoring these frameworks to regulated aerospace environments.

### 3.5 Gaps Identified

The current body of literature reveals the following gaps:

- A lack of quantitative benchmarks for availability in deployed aerospace monitoring systems.
- Insufficient cross-domain synthesis of software reliability, systems engineering, and PDMS infrastructure.
- Few case studies tracing iteration-to-iteration improvements in real-world monitoring system uptime.
- Limited understanding of how availability targets affect downstream activities such as maintenance planning, compliance reporting, or operational decision-making.

This paper seeks to address a subset of the above gaps by presenting a structured analysis of iteration results from aerospace PDMS deployments, identifying the operational and organizational levers that contributed to achieving or approaching “Three Nines” availability.

## 3. PDMS SYSTEMS DATA FLOW

Figure 1 illustrates an end-to-end architecture of a Product Data Monitoring System tailored for aerospace applications. It captures the flow of data from collection at the source—whether aircraft, ground systems, or maintenance facilities—through various stages of storage and processing, culminating in curated visualization layers accessible to both internal users and external customers. This structured pipeline ensures traceability, security, quality, and actionable insight delivery aligned with operational needs.

The PDM system is designed to collect, preprocess, store, process, and visualize data from aircraft systems and other external sources. The design is mandated through a set of functional and non-functional requirements. The goal is to generate Key Performance Indicators (KPIs) that help stakeholders monitor and predict the health and performance of the product.

The overall system of interest, for which achieving "Three Nines" is discussed in this paper, is shown with a boundary in Figure 1.

**Data Collection and Ingestion**

PDMS begins at the edge with data collection from various sources such as onboard data acquisition units, maintenance logs, and operator reports. This raw data is ingested through a preprocessing pipeline that includes unpacking, decoding, decryption, concatenation, and quality assurance. The goal of this phase is to ensure that the ingested data maintains structural integrity and is ready for further analysis. Furthermore, it is important for the reader to recognize that the above preprocessing steps could be an integral function or could be modular with a sequential set of operations. Moreover, to achieve optimal operational efficiencies, one or more of the modular units could spill over into the downstream processing section.

**Raw Data Storage and Processing**

Following ingestion, the data is stored in local or cloud-based data storage such as Azure, Oracle, or on-premises solutions. The raw processing stage applies data normalization, statistical characterization, and analytical pre-processing to enhance the fidelity of the data and prepare it for deeper insights. This stage is often executed using scripting environments (e.g., Python) and ensures the datasets are both structured and semantically enriched.

**Curated Data Storage and Processing**

Processed data is curated in a secure, governed storage layer - typically implemented via cloud-native data warehouses or federated databases (e.g., Amazon Redshift, Azure Synapse). In this phase, algorithmic processing plays a central role. Trend detection, anomaly detection, and predictive algorithms (including machine learning and statistical models) are applied to transform data into meaningful diagnostic and prognostic metrics.

**Visualization and Role-Based Access**

The final stage involves visualization of Key Performance Indicators (KPIs), Service Level Agreement (SLA) metrics, alerts, and predictive insights tailored to user roles. Data is rendered into dashboards, charts, and reports accessible through secure, role-based access control (RBAC). Users (e.g., engineering teams) and customers (e.g., airlines) are granted access based on authorization policies to ensure data confidentiality and relevance.

To summarize, in aerospace PDMS, the design of the data flow architecture is a decisive factor in achieving both high availability and operational effectiveness. While the stages of data ingestion, storage, processing, and access control are common across industries, their implementation in the aerospace context demands heightened rigor due to strict performance, safety, and regulatory requirements.

A key architectural shift enabling greater availability and resilience is the decoupling of the storage and processing layers. By separating these concerns, PDMS can ensure that data ingestion, availability, persistence / durability, replication, disaster recovery and scalability are independent from downstream computational workloads. This architectural pattern supports data durability, allowing raw and pre-processed health data to be securely stored and later reprocessed without loss or degradation - critical in scenarios involving delayed connectivity or re-analysis for compliance.

The storage layer must prioritize redundancy, replication, and consistency to preserve data integrity under all operating conditions. In contrast, the processing layer must deliver scalable, real-time analytics and anomaly detection while tolerating component failures. This separation enhances fault isolation and allows independent scaling, maintenance, or recovery of each layer without jeopardizing overall system uptime.

Furthermore, features such as role-based access control, auditability, and secure transmission protocols are now standard expectations, aligning aerospace systems with best practices from adjacent industries like finance and healthcare. However, aerospace applications must also contend with intermittent connectivity, certifiability, and embedded system constraints, which demand customized adaptations of these practices.

In summary, designing PDMS data flows for aerospace requires not only adherence to cross-industry architectural principles but also intentional enhancements that account for mission-critical reliability and system-level availability goals. A well-architected, decoupled data flow with durable storage and resilient processing is essential for supporting the high bar set by the "Three Nines" availability target.

## 4. REAL-TIME MONITORING & ALERTING SYSTEMS

Real-time monitoring and alerting systems are foundational to achieving high availability and reliability targets, such as the "three nines" (99.9% uptime), in any software architecture. These auxiliary systems enable continuous oversight of critical metrics, detect anomalies as they occur, and trigger immediate responses to potential failures.

By minimizing detection and response latency, real-time monitoring not only reduces the mean time to recovery (MTTR) but also ensures that service-level agreements (SLAs) are met consistently.

### 4.1. Logging

The bedrock for effective monitoring and alerting solutions is system wide logging of key metrics and data flow through a system. The simplest logs often take the form of timestamped printouts in a text file. This can serve well for root cause analysis and post-incident reviews but is not purpose-built for real time monitoring and alerting.

Purpose-built solutions include:

1. Document-Oriented NoSQL Databases such as OpenSearch and MongoDB. Documents enable scalable monitoring and querying on predefined fields.
2. Timeseries databases, such as Prometheus, enable tracking trends over time. This is crucial for detecting and resolving anomalies prior to system outages.
3. Managed solutions such as AWS CloudWatch, Azure Monitor, and Google Cloud Logging for cloud-based infrastructures. These solutions support real-time log streaming, metric extraction, alerting, and role-based access control.

### 4.2. Alerting Systems

Once key metrics are captured through structured logging, alerting solutions must be considered to mitigate issues in real time. Comprehensive logging solutions such as AWS CloudWatch will often have alerting capabilities built right in. Other tools such as OpenSearch provide easy-to-use APIs to support custom solutions.

For data ingestion and processing systems such as PDMS, there are three key alert types:

1. Threshold exceedance: Triggered when a predefined metric (i.e., CPU utilization, memory consumption, error rate, latency, queue size) crosses a critical limit. These alerts provide a straightforward way to detect system failure and are especially useful for catching immediate, high-impact issues.
2. Ingestion / processing / delivery gaps: These alerts ensure consistent throughput between each component of a larger PDM system. For enterprise scale systems, empty queues often translate to upstream system failures and should be monitored to detect component failure.
3. Time-series trend changes: Used to detect anomalies prior to full scale system outages. These types of alerts can involve more complex statistical models or simple heuristics such as a queue which does not decrease over a given period.

### 4.3. Visualization

Visualization plays a vital role, transforming raw metrics, logs, and events into intuitive, real-time insights. These visualizations help teams to quickly identify patterns, anomalies, or performance regressions across complex systems. Time-series graphs, heatmaps, and correlation views, along with well-established baselines, allow analysts to contextualize alerts, trace failures, and identify degraded system states. Common visualization tools include Grafana for time-series data with built in alerting tools, OpenSearch for tables, dashboards, and heat-map visualizations on document records, and AWS QuickSight for easy insights into cloud hosted data stores.

### 4.4. Operational Support

Logging, visualization, and alerting strategies quickly fall apart without equally reliable operational support to action identified issues. The so called “Three Nines” (99.9%) system availability necessitates a 24 / 7 on call support team to ensure a minimized mean time to recovery (MTTR). Given that alerts can (and will) go off at all hours of the day, it is critical to have rock solid work instructions for the support team. Standard operating procedures with a well-reviewed decision tree and clear escalation points are key to success.

## 5. OPERATIONAL STRATEGIES TO SUPPORT THREE NINES

In aerospace Prognostics and Health Management (PHM) systems, ensuring high availability is critical for mission assurance, safety, and operational efficiency. Achieving "Three Nines" availability, or 99.9% uptime, requires a combination of robust architectural design, disciplined operational practices, and proactive fault mitigation strategies. This section outlines the core operational strategies used to support high availability in PHM environments, emphasizing data delivery assurance, infrastructure redundancy, automated recovery, software reliability, and scalable cloud migration approaches. By integrating these elements, organizations can build resilient systems capable of maintaining continuous service, even in the face of component failures or evolving system demands.

### 5.1. Operational Excellence and Data Delivery Assurance

A cornerstone of operational excellence is the systematic tracking of data delivery performance. Monthly reporting of ingested versus delivered data provides visibility into pipeline health, allowing stakeholders to monitor discrepancies between raw binary data inputs and their transformed outputs (e.g., CSV, JSON, Parquet formats). Metadata tracking enables accountability at each processing step and intermediate monitoring for system degradation / outages.

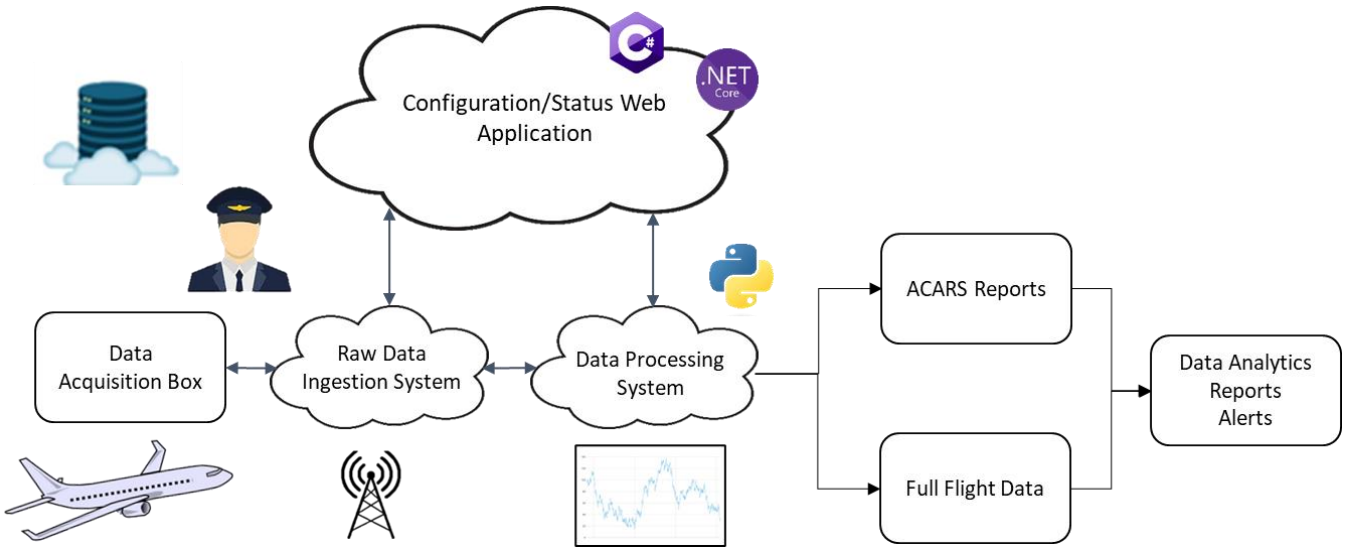


Figure 2. Real-Time & Near Real-Time Data Acquisition

To support high availability, automated reprocessing mechanisms are triggered upon failure, reducing human intervention and recovery time. System uptime metrics are correlated with data delivery thresholds to ensure timely and complete transmission of processed data. Supporting this process are instance health checks, which can be either custom-developed or based on native platform tools (e.g., AWS EC2 health checks). Faulty virtual machines (VMs) or containers are automatically terminated and replaced through orchestration services (i.e., Kubernetes, Podman, Docker Swarm), further ensuring service continuity. Dedicated monitoring solutions such as Grafana, AWS CloudWatch, or internally developed dashboards provide real-time insight into system health. With well-defined alerting criteria and a 24/7 support rotation, teams are positioned to respond to critical alerts within 15 minutes or less, meeting stringent recovery expectations.

### 5.2. Software Testing and Release Assurance

Availability goals are also supported through rigorous software testing practices. Standards, such as RTCA DO-178C, Level 1A provide detailed guidance for achieving reliable software deployments. This includes unit, integration, and system testing to validate individual components and their interactions. Regression testing to ensure new updates don't reintroduce old bugs. Functional and non-functional testing—particularly performance, reliability, and failover scenarios—are essential to validate system behavior under real-world conditions. Organizations must weigh the benefits of solid testing against the risks related to the loss of service to determine the appropriate verification approach.

Additionally, User Acceptance Testing (UAT) plays a critical role by involving actual end users in validating that the system meets operational requirements before deployment, catching issues that automated testing may overlook. Testing can thus be categorized across multiple layers, including Unit, Integration, System, Regression, Performance, Security, and UAT, each addressing different risk domains.

By embedding these comprehensive testing methodologies into Continuous Integration/Continuous Deployment (CI/CD) pipelines, PDMS can confidently deliver new features and updates without compromising uptime. This proactive defect detection and resolution process significantly decreases production issues, minimizing outages and system downtime. Consequently, rigorous and layered testing directly supports achieving and sustaining high availability targets, such as the Three Nines (99.9%) uptime, critical for mission-critical aerospace operations.

## 6. CASE STUDIES AND LESSONS

Two case studies illustrate the transformative impact of advanced PDMS in data acquisition system monitoring. These case studies highlight how automation, data analytics, and cloud integration can significantly enhance system availability, reduce operational costs, and improve decision-making.

**Case Study 1:** Modernizing Data Acquisition Monitoring through Automation and System Integration.

An established manufacturer partnered with Belcan to modernize its legacy data acquisition and processing

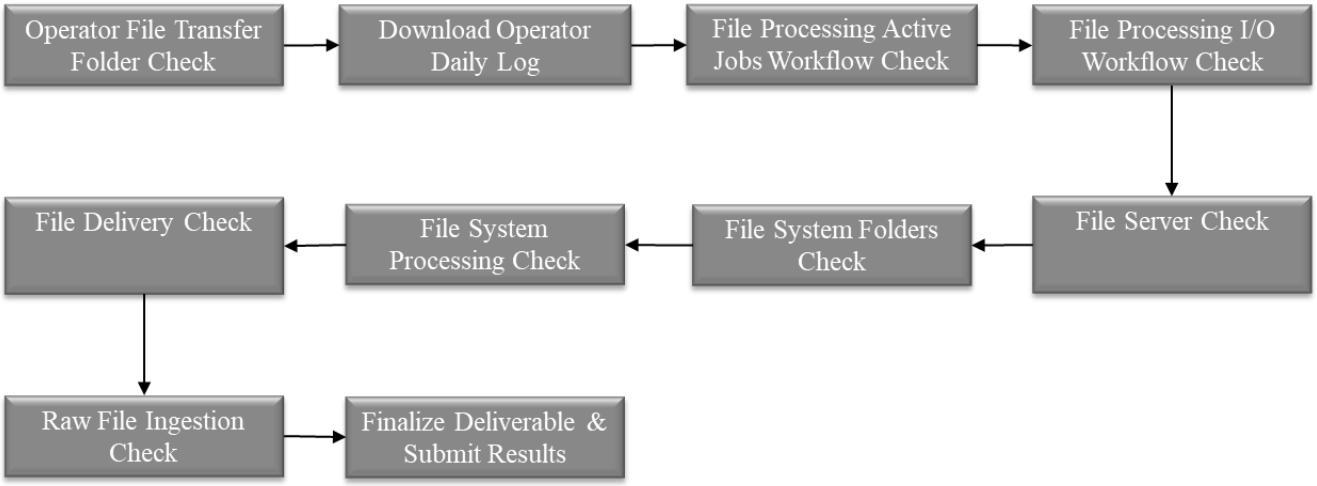


Figure 3. System Integrity Check Workflow

monitoring systems. The real-time and near real-time data acquisition system is represented in Figure 2. While the initial scope centered on verification and validation (V&V) tasks, the engagement quickly evolved into a comprehensive effort encompassing software development, data analysis, and operational support. This transition was driven by growing challenges associated with cross-domain integration and increasing demands for system availability and responsiveness.

Belcan team faced a variety of complex challenges, including the need to integrate embedded software with hardware systems, web applications, and cloud-based services. Frequent system outages and recurring software issues caused unacceptable levels of downtime, threatening contractual service-level agreements. Addressing these challenges required a multi-disciplinary team capable of bridging gaps across system design, analytics, operations, and training.

The system integrity check workflow, as illustrated in Figure 3, involves ten sequential steps—from checking operator file transfer folders to finalizing deliverables and submitting results. Each step, while essential, was manually executed and consumes valuable time measured in minutes per task. This manual approach not only was limited in frequency and flexibility, but introduced inefficiencies, risks of human error, and delays in downstream processes.

To resolve these issues, Belcan developed and deployed 24/7 automated system monitoring tools that provided real-time alerts and operational visibility. In parallel, the team implemented automated regression testing to catch defects earlier in the software development lifecycle. These solutions significantly reduced manual intervention and system downtime while improving system stability. As a result, the client was able to significantly reduce response time (and therefore MTTR) and redirect valuable engineering resources to higher-impact tasks.

The financial benefits of these improvements were substantial. Broader system design and validation improvements delivered millions of dollars in cumulative cost avoidance and savings, while automation of system integrity check workflow alone reduced engineering hours by more than 90%. In total, the program achieved approximately 49% overall savings, highlighting the long-term value of integrated monitoring and automation in modern product data monitoring systems.

**Case Study 2:** Achieving and Sustaining 99.9% System Availability.

Building on the success achieved in case study 1, the team shifted its focus to a new and ambitious performance target: achieving 99.9% system availability on a rolling three-month average. This goal translates to a stringent constraint of no more than 44 minutes of system downtime per month, a major leap from the prior benchmark of 99.5% availability (approximately 219 minutes of downtime).

This effort centered on enhancing the reliability of data ingestion, processing, and delivery - particularly the operational reports generated by sourcing systems. These data streams, which include both snapshot telemetry and event data, must be ingested, processed, and distributed without disruption.

To meet these demands, the team developed a suite of Python-based monitoring tools that leveraged Selenium WebDriver for web interface testing and SQL for database validation. These tools provided deep visibility into system health and enabled rapid identification of system degradations and outages. In tandem, the team established a 24/7 operations center, integrating human oversight with automated monitoring to provide operational redundancy.

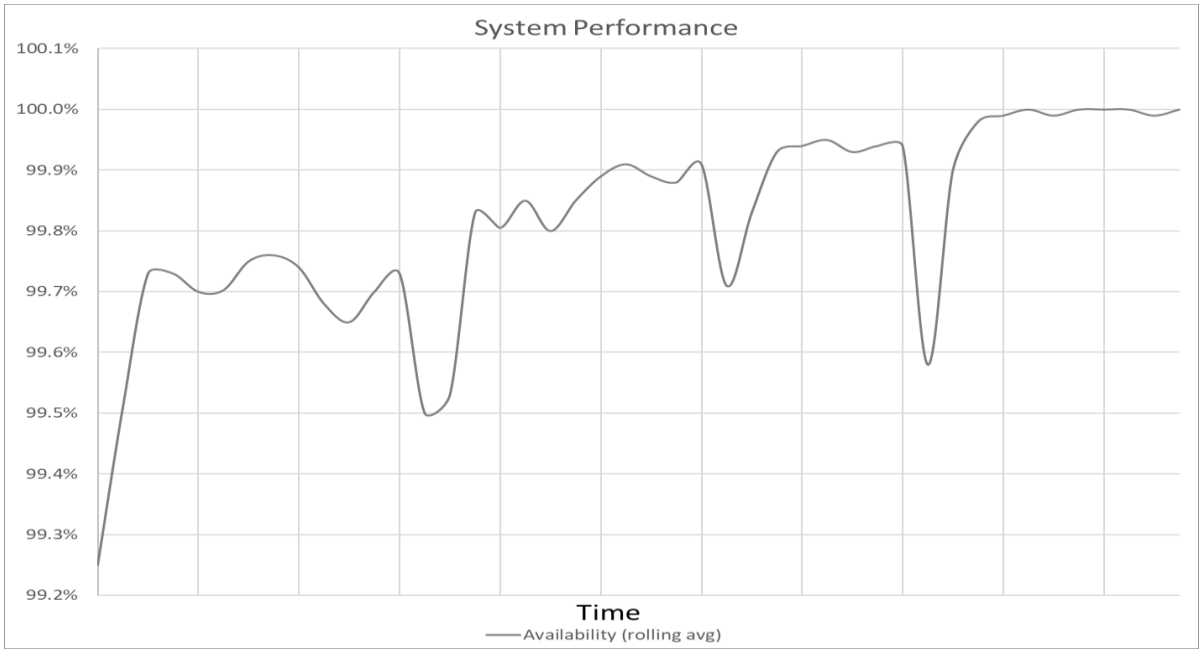


Figure 4. Overall System Availability Performance

The results were immediate and measurable. Figure 4 shows the 3-month rolling average of the overall system availability.

The availability of the system shows a clear trend of improvement over time, reaching levels at or near 100% towards the end. This was primarily due to the automated tools as well as optimizing and maturing them over time. Early in the period, system outages, software releases, etc., would cause extended downtime impacting the availability metrics. In the later period, incorporating the lessons learned, the team was able to fully customize the tools to adjust frequency, variables and system components to minimize MTTR and maintain availability levels at or near 100% despite sharp declines.

These case studies underscore the critical role of automation, cloud integration, and cross-functional collaboration in achieving high system availability in PDMS environments. The transition from manual to automated monitoring not only improved reliability but also delivered substantial cost savings and operational efficiency. These outcomes serve as a benchmark for future PDMS implementations across the aerospace industry.

## 7. CONCLUSION

Achieving 99.9% availability in Product Data Monitoring Systems (PDMS) represents a significant engineering, operational, and organizational milestone for the aerospace industry. As monitoring systems continue to evolve from passive data loggers to critical infrastructure that reinforces safety, availability, and predictive maintenance, the demand for reliable, always-on performance has never been higher.

This paper has explored the “Three Nines” availability goal through the lens of previous system iterations, tracing how process refinements, and operational maturity have incrementally advanced system uptime. These iterations reveal that availability is not achieved through one-time design choices, but through a sustained, system-level strategy that includes redundancy, automation, observability, and intelligent fault management.

Importantly, this study reinforces that availability must be defined in functional terms, not just as uptime percentages. A system can only be considered "available" when it can reliably capture, transmit, process, and deliver health insights that inform safety-critical decisions in real time. This layered understanding of availability is essential as PDMS become deeply integrated with digital twins, automated maintenance, and data-driven airworthiness frameworks.

Looking ahead, continued progress toward and beyond Three Nines will require:

- Rigorous iteration cycles informed by real-world usage and failure data.
- Cross-disciplinary collaboration across software, systems, reliability, and safety engineering teams.
- Investments in AI/ML, edge computing, and autonomous health assessment to further reduce dependency on centralized systems.
- Standards evolution to reflect functional availability expectations within safety-critical certification regimes.

In closing, the pursuit of Three Nines is not simply about minimizing downtime—it is about maximizing trust in the systems that monitor and maintain the health of complex aerospace assets. By embedding availability as a core design and operational requirement, aerospace organizations can ensure that their PDMS are ready to meet the growing demands of a connected, data-driven aviation future.

## ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to Belcan Engineering, a Cognizant company, for their generous support and sponsorship of this industry research. Their commitment to advancing innovation in aerospace health monitoring systems has been instrumental in enabling this study. We also acknowledge the valuable insights and collaboration provided by the company's technical and operational teams, which greatly enriched the quality and relevance of this work.

## NOMENCLATURE

| | |
|---|---|
| *AI/ML* | Artificial Intelligence / Machine Learning |
| *CBM* | Condition-Based Maintenance |
| *CI/CD* | Continuous Integration and Continuous Delivery |
| *EHM* | Engine Health Management |
| *FMECA* | Failure Modes, Effects and Criticality Analysis |
| *KPI* | Key Performance Indicator |
| *MTTR* | Mean Time to Recovery |
| *OEM* | Original Equipment Manufacturer |
| *PDMS* | Product Data Monitoring Systems |
| *PHM* | Prognostics and Health Management |
| *RCM* | Reliability-Centered Maintenance |
| *SHM* | Structural Health Management |
| *SLA* | Service Level Agreement |
| *SLI* | Service Level Indicator |
| *SLO* | Service Level Objective |
| *SRE* | Site Reliability Engineering |
| *RBAC* | Role-Based Access Control |
| *UAT* | User Acceptance Testing |
| *V&V* | Verification and Validation |

## REFERENCES

Beyer, B., Jones, C., Petoff, J., & Murphy, N. R. (Eds.) (2016). *Site Reliability Engineering: How Google Runs Production Systems.* O'Reilly Media. ISBN 978-1491929124.

Federal Aviation Administration (FAA) Advisory Circular 43-216 (2017). *Software Management During Aircraft Maintenance.*

Federal Aviation Administration (FAA) Advisory Circular 43-218 (2022). *Operational Authorization of Integrated Aircraft Health Management (IAHM) Systems.*

International Electrotechnical Commission. (2006). *IEC 61025: Fault tree analysis (FTA)* (2nd ed.). Geneva, Switzerland: IEC.

IEEE Instrumentation and Measurement Society. (2016). *IEEE 1856–2016: Guide for Prognostics Health Management (PHM) standards—Data formats, architectures, and processes for PHM systems*. Piscataway, NJ: IEEE.

International Organization for Standardization. (2003). *ISO 13374-1: Condition monitoring and diagnostics of machines—Data processing, communication and presentation—Part 1: General guidelines*. Geneva, Switzerland: ISO.

Koopman, P. (2003). *Elements of the Self-Healing System Problem Space. In Workshop on Architecting Dependable Systems/WADS03*.

Li, R., Verhagen, W. J. C., Curran, R. (2020). *Toward a methodology of requirements definition for prognostics and health management system to support aircraft predictive maintenance*. *Aerospace Science and Technology, 102*, 105877. https://doi.org/10.1016/j.ast.2020.105877

Roy Choudhury, I., Reveley, M. S., Phojanamongkolkij, N., & Leone, K. M. (2017). *Assessment of the state-of-the-art of system-wide safety and assurance technologies* (NASA/TM–2017–219515). NASA Glenn Research Center. https://ntrs.nasa.gov/citations/20170011193

RTCA, Inc. (2012). *DO-178C: Software considerations in airborne systems and equipment certification* (ED-12C). Washington, D.C.: RTCA, Incorporated.

RTCA, Inc. (2014). *DO-326A/ED-202A: Airworthiness security process specification*. Washington, D.C.: RTCA, Incorporated.

SAE International. (2010, December). *ARP 4754A: Guidelines for development of civil aircraft and systems*. Warrendale, PA: SAE International.

Saha, B., & Goebel, K. (2009, September). *Modeling Li-ion battery capacity depletion in a particle-filtering framework*. In *Proceedings of the Annual Conference of the Prognostics and Health Management Society* (Vol. 1, No. 1). PHM Society.

Saxena, A., Goebel, K., Simon, D., Eklund, N., & Saha, S. (2008). *Damage propagation modeling for aircraft engine degradation prognostics.* In *Proceedings of the International Conference on Prognostics and Health Management (PHM)* (Vol. 10, pp. 123–132). IEEE.

Seshadri, B. R., & Krishnamurthy, T. (2017). Structural health management of damaged aircraft structures using digital twin concept. In *25th AIAA/AHS Adaptive Structures Conference* (p. 1675).

U.S. Department of Defense. (1980). *MIL-STD-1629A: Procedures for performing a failure mode, effects and criticality analysis (FMECA)*. Washington, D.C.: U.S. Department of Defense.

Vachtsevanos, G., Lewis, F. L., Roemer, M., Hess, A., & Wu, B. (2006). *Intelligent fault diagnosis and prognosis for engineering systems* (1st ed.). John Wiley & Sons.

Weber, R. H., & Weber, R. (2010). *Internet of Things – Legal Perspectives*. Heidelberg, Berlin: Springer

## BIOGRAPHIES

**Raghothama M. Rao** is a seasoned engineering leader with over two decades of experience in product design and systems integration across the aerospace, fuel cell, and power generation industries. He currently serves as a Group Leader in Systems and Software Engineering at Belcan, a Cognizant company Throughout his career, Dr. Rao has built a strong reputation for driving innovation and delivering high-impact solutions in safety-critical and high-reliability environments. His expertise spans model-based systems engineering (MBSE), process improvement, cost reduction, and complex verification and validation (V&V) initiatives. He is known for his ability to lead cross-functional teams and optimize systems for performance and efficiency. In recognition of his professional excellence, Dr. Rao was named Distinguished Engineer of the Year by ASME Hartford in 2020. He also served as President of the New England Chapter of the International Council on Systems Engineering (INCOSE) in 2022. Dr. Rao earned his Bachelor of Science degree in Chemical Engineering from the Indian Institute of Technology, Mumbai, in 1996, and his Ph.D. in Chemical Engineering at Clarkson University, Potsdam, New York, in 2006.

**Travis G. Mann** is an aerospace systems and software engineer with nearly half a decade of experience in data acquisition systems, software testing, and system availability. He currently serves as the engineering technical lead and project manager on two teams for a commercial flight data acquisition program at Belcan, a Cognizant company. Travis graduated summa cum laude from the University of Connecticut in 2021 with his Bachelor of Science in Mechanical Engineering and earned his Master of Computer Science from Worcester Polytechnic Institute (WPI) in 2025. Known as a strong cross-functional leader across operations, engineering, and development teams in the aerospace and defense fields, with a strong focus on automation and process improvement. In recognition of his technical excellence, he was selected as the Belcan Engineer of the Year for back-to-back years in 2024 and 2025.

**William Edinger** is an experienced software engineer with 40 years of experience developing real time embedded software solutions for military aircraft, submarine imaging systems and life support systems for space applications. Mr. Edinger is a recognized leader in establishing a culture of continuous improvement and process innovation related to software development, health management verification solutions and data transfer monitoring. He currently serves as a Section Leader in Systems Software Engineering at Belcan, a Cognizant company. Mr. Edinger earned his Bachelor of Science degree in Physics at the University of Massachusetts, Dartmouth, North Dartmouth, MA in 1984 and his Master of Science degree in Electrical Engineering at Western New England University in Springfield, MA in 1988.