Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2019, 6(8):45-50

Research Article ISSN: 2394 - 658X

Integrated Field Service Operating System (IFOS): Deployable Solution for Machine Data Tracking

Naveen Koka

*Email id: *na.koka@outlook.com*

ABSTRACT

The Integrated Field Service Operating System (IFOS) incorporates a criterion-based scheduling daemon to monitor data continuously, alerting users of any discrepancies detected. These daemons, possibly leveraging machine learning algorithms, are trained within the IFOS environment. Upon qualification for preventive maintenance (PM), IFOS initiates predefined rules to generate requisite data and invoke configured HTTP requests. After receiving responses, the system stores the data for additional verification. This feature plays a pivotal role in system performance assessment, facilitating timely notifications to external systems to address service-related concerns and mitigate downtimes effectively. IFOS also includes a boot loader feature, mandating critical device details such as ID, location, installation date, and contact information, ensuring comprehensive system configuration and maintenance management. Through these functionalities, IFOS aims to streamline device maintenance operations, minimizing manual intervention while maximizing system efficiency and reliability.

IFOS offers a comprehensive suite of tools and features designed to optimize system performance and enhance security measures. In addition to its criterion-based scheduling daemon and boot loader functionalities, IFOS provides robust access management capabilities. Users can configure permissions and restrictions for both admin and non-admin users, ensuring secure access to critical system resources.

Furthermore, IFOS supports seamless integration with external systems through its flexible API framework. This allows for easy communication with third-party applications and services, enabling enhanced functionality and interoperability.

The system's logging and monitoring capabilities provide real-time insights into system activity, allowing administrators to quickly identify and address any potential issues or security threats. Detailed logs and reports offer comprehensive visibility into system performance, aiding in troubleshooting and optimization efforts.

Overall, IFOS represents a powerful and versatile solution for organizations seeking to enhance their network security and streamline system management processes. With its comprehensive feature set and robust security measures, IFOS offers peace of mind and reliability in an ever-evolving digital landscape.

Key words: Machine Learning, OS for Field Service, Machine self-manageable

---

INTRODUCTION

Machines are the backbone of our modern world, ranging from small devices to large industrial equipment. They serve myriad purposes, from everyday conveniences to critical functions such as medical care. The smooth operation of these machines is vital to our daily lives, and any malfunction can disrupt our routines significantly. As our reliance on machines grows, the need for reliable and efficient operating systems becomes increasingly evident.

PROBLEM STATEMENT

In today's world, the seamless operation of machines, ranging from small devices to critical medical equipment, is indispensable for daily functioning. However, the increasing reliance on these machines also amplifies the risks associated with downtime, which can disrupt essential services and daily routines. Existing operating systems lack the capability to proactively manage machines, leaving organizations vulnerable to unexpected failures and inefficiencies. There is a pressing need for an operating system solution that empowers machines to become self-manageable, capable of autonomously monitoring performance, identifying issues, and implementing corrective actions to prevent downtime and optimize service delivery. This problem statement underscores the urgency of developing an innovative operating system that addresses the evolving demands of our machine-dependent society, ensuring reliability, efficiency, and continuity in critical and everyday applications.

SOLUTION

By offering an operating system that enables machines to become self-manageable, we can mitigate downtime and enhance service delivery. Such a system would empower machines to autonomously monitor their own performance, identify issues, and implement corrective actions as needed. This proactive approach not only minimizes the risk of unexpected failures but also ensures continuous and optimized operation.

In the context of critical applications like healthcare, where machine downtime can have life-threatening consequences, the importance of self-manageable systems cannot be overstated. By leveraging advanced technologies such as artificial intelligence and predictive analytics, our operating system can anticipate potential issues before they escalate, thus safeguarding the uninterrupted function of critical equipment.

Furthermore, a self-manageable operating system offers benefits beyond reliability and uptime. It enables seamless integration with existing infrastructure, simplifies maintenance procedures, and enhances overall efficiency. By reducing the need for manual intervention and optimizing resource utilization, organizations can achieve cost savings and improve productivity.

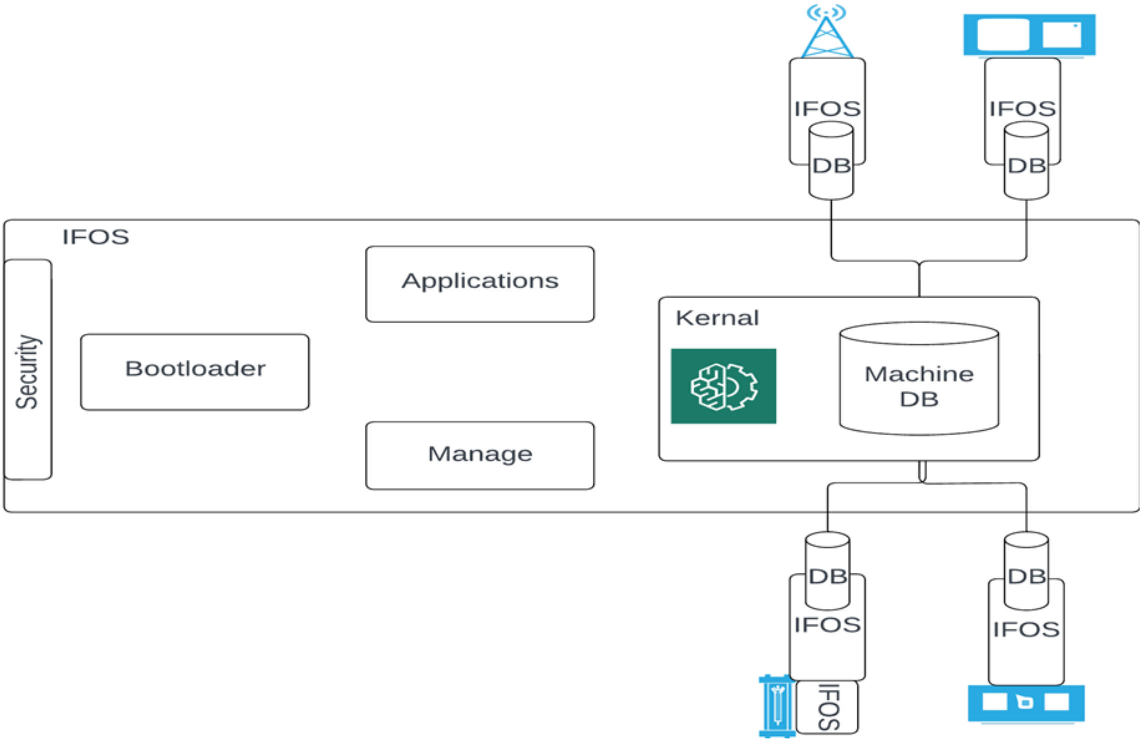

Figure 1: IFOS with multiple machines deployed

IFOS

The platform serves as the fundamental infrastructure for delivering applications, representing the critical junction where practical implementation takes place. It bears the responsibility of managing diverse organizations, users, and applications. The following components require careful attention and maintenance. As the platform operates on a multi-tenant model, development becomes crucial, and careful attention must be given to each of the following components:

Boot Loader

This startup component is essential for initializing other services. Operating on our Linux-based platform, once Linux is initiated, our boot loader is triggered to commence loading kernel services, along with any subscribed applications set to start on boot, and any partially completed or missed daemons due to previous downtime.

This will be constructed using Python, enabling access to the internal operating system. The following components need to be developed and internally deployed:

Tkinter UI application for configuring and managing the registration of other services.

Database for storing the configurations.

Kernel

This constitutes the essence of machine data tracking. During deployment, the kernel can be classified as either a receiver or a sender. If it functions as a sender, the data is stored as part of the sensor data. Upon receiving sensor data, we must begin processing the information and then save it to the database.

Manage

The Tkinter UI application is designed to oversee and configure any necessary message processing services before data is saved to the database. The UI should also offer various ports for listening to all sensor data, along with the capability to establish rules for accepting sensor data, thereby enabling the elimination of unnecessary data. These sensor ports will be exposed as services for application access.

Persist

Once the data is prepared for storage, the persistence module is tasked with saving it into the database. The database is utilized for storing machine data.

Tail

The tail module is responsible for sending data based on the database contents and the predefined rules established within the management system.

Shell

The shell application facilitates the swift execution of commands. Since the senders are compact sensor devices accessible via SSH, operations are conducted through the terminal. In this scenario, commands are utilized to configure functionalities typically managed within the boot loader and core management system.

The shell offers various useful commands such as:

- Initiate and terminate a service
- Customize an application within the boot loader
- Adjust data services within the management system

Applications

Here, customers can create applications tailored to their specific requirements. Within a sender device, developers can craft applications capable of accessing and processing database data. Additionally, they can subscribe to services offered by the kernel to execute various operations.

It's important to note that these applications are limited to read-only access to the database. Furthermore, they can solely listen to the services facilitated by the kernel.

System Library

As part of the operating system, we offer reusable libraries to facilitate certain functionalities. For instance, if your application needs to interact with the database, we provide the JDBC connector.

Daemons

This is the platform where users can schedule lengthy tasks. A user interface is available for scheduling jobs and linking associated services. Admins are presented with various options to create and manage these tasks.

Additionally, a logging user interface is provided to display daemon statuses, execution logs, and detailed statements.

Security

Given that this is a Linux-based operating system, users can configure antivirus software and firewalls.
The Integrated Field Service Operating System (IFOS) includes an internal access management system. Users must log in using their credentials. Initially, a default user is provided, and the administrator must modify and configure additional users. Two user types exist: admin and non-admin. Admins can limit non-admin users to specific tasks such as querying data, configuring daemons, and monitoring logs.

FIELD SERVICE

Until we've established the necessary OS features to manage all essential components for servicing the machine, we'll outline steps to streamline device maintenance with minimal manual intervention. This segment is referred to as Field Service.
The field service system may be considered optional, with the concept being to offer mechanisms for supplying all necessary information to external systems.

Asset Management

Once all the machines are deployed, we need to uniquely identify these assets and maintain the data in the database. This gives us the ability to interact based on the context.
As an integral component of the boot loader, it is essential to furnish the device ID, location specifics, address, installation date, and contact details.

Preventive Maintenance

This is essential for evaluating system performance and triggering notifications to external systems to generate related service requests. Configuring this feature is paramount to prevent any downtimes.
The IFOS offers a criterion-based scheduled daemon that continuously monitors data and sends notifications upon detecting any discrepancies. These daemons can consist of machine learning algorithms, which are continuously trained within the IFOS environment.
Once the IFOS meets the criteria for preventive maintenance, the defined rules will generate the necessary data and trigger configured HTTP requests. Upon receiving the response, we will store it for further verification purposes.

USES

IFOS represents a powerful and versatile solution for organizations seeking to enhance their network security and streamline system management processes. With its comprehensive feature set and robust security measures, IFOS offers peace of mind and reliability in an ever-evolving digital landscape.

Medical Industry

Machine downtime can have life-threatening consequences, the importance of self-manageable systems cannot be overstated. By leveraging advanced technologies such as artificial intelligence and predictive analytics, our operating system can anticipate potential issues before they escalate, thus safeguarding the uninterrupted function of critical equipment.
To better provide an example:
The proposed solution involves configuring a rule within the system to track and analyze the brightness levels of X-ray reports in real-time. This rule would specify a maximum allowable difference in brightness compared to the expected baseline. If the observed brightness exceeds this threshold or deviates significantly from the norm over a defined period or number of instances, the system triggers a notification to alert relevant personnel of a potential issue with the light source. Additionally, the system can transmit machine-related data and other quantifiable parameters for further analysis and verification, facilitating prompt diagnosis and resolution of the problem.

Overall, the implementation of this rule-based system enhances the monitoring and maintenance of X-ray machines, ensuring optimal performance and minimizing the risk of inaccurate diagnostic results due to malfunctioning components such as the light source.

Auto Industry

The automotive industry offers specialized warranty packages for customers seeking additional coverage to prevent future issues. Within this framework, specific applications can be installed exclusively for these customers. By proactively analyzing data well in advance and promptly addressing potential issues, customer satisfaction can be greatly enhanced.

Digital Automation & Energy Management Industry

There is a need for a comprehensive, one-stop solution capable of seamlessly collecting, aggregating, and analyzing data from a wide range of machines across diverse sectors. This solution should provide a unified platform for data management, offering functionalities for real-time monitoring, historical analysis, predictive maintenance, and security monitoring. By integrating with machines from various manufacturers and industries, the solution enables organizations to gain holistic visibility into their operations, identify trends, and make informed decisions to improve efficiency, productivity, and security.

Cost Savings

Internally managed, the platform facilitates straightforward application development and user reviews. Continuous feedback fosters agility, yielding prompt results. With only one platform to learn, manage, and maintain, the maintenance cost is minimized.

SCOPE

The scope of IFOS is confined to delivering the operating system and essential tools for managing and learning the devices installed by providers. Field Service, on the other hand, facilitates external communication with other systems, enabling the maintenance of a system of record.

CONCLUSION

IFOS focuses on delivering a robust operating system and essential tools tailored for managing and learning installed devices, catering to provider needs efficiently. Complementing IFOS, the Field Service component extends outward communication capabilities to interface with external systems, ensuring seamless maintenance and management of a comprehensive system of record. This integrated approach underscores a commitment to optimizing device performance and enhancing interoperability within a diverse technological landscape.

REFERENCES

[1]. Bertil Folliot, Ian Piumarta, and Fabio Riccard, A Dynamically Configurable, Multi-Language Execution Platform, 1998, DOI:10.1145/319195.319222.

[2]. Yann LeCun, Yoshua Bengio, Geoffrey Hinton. 2015. Deep learning. https://www.nature.com/articles/nature14539

[3]. Hyman, William. 2003. The Theory and Practice of Preventive Maintenance. https://journals.lww.com/jcejournal/abstract/2003/01000/the_theory_and_practice_of_preventive_maintenance.37.aspx

[4]. Shannon Wells. 2016. Way Engineering Replaces Paper-Based Field Service Processes with A Completely Digital, Web-Based Mobile Field Service Solution By Fieldconnect, Inc. https://www.prweb.com/releases/way_engineering_replaces_paper_based_field_service_processes_with_a_completely_digital_web_based_mobile_field_service_solution_by_fieldconnect_inc_/prweb13832135.htm#.

[5]. Wei Vivien Shi. 2016. Sensors Applied in Healthcare Environments. DOI: 10.4236/jcc.2016.45015.

[6]. Andreas Schumacher, Wilfried Sihn, Selim Erol. Automation, digitization and digitalization and their implications for manufacturing processes.

[7]. https://www.researchgate.net/publication/318877006_Automation_digitization_and_digitalization_and_their_implications_for_manufacturing_processes

[8]. Marcel Eckert, Dominik Meyer, Jan Haase, Bernd Klauer. Operating System Concepts for Reconfigurable Computing: Review and Survey. 2016. DOI:10.1155/2016/2478907

[9]. Jim Torresen. A Review of Future and Ethical Perspectives of Robotics and AI. 2018. https://doi.org/10.3389/frobt.2017.00075