Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 10 (2017) 1020 – 1030

Procedia MANUFACTURING

45th SME North American Manufacturing Research Conference, NAMRC 45, LA, USA

Streaming Machine Generated Data to Enable a Third-Party Ecosystem of Digital Manufacturing Apps

Shaurabh Singh[a], Atin Angrish[a], James Barkley[b]*, Binil Starly[a]*, Yuan-Shin Lee[a], Paul Cohen[a]

[a]*Edward P. Fitts Department of Industrial and Systems Engineering*
North Carolina State University, Raleigh, NC 27695
[b]*UI Labs, Digital Manufacturing Design Institute, 1415 N Cherry Ave, Chicago, IL 60642*

Abstract

The digital factory of the future will be driven by the integration of physical smart machine tools and cyber-enabled software, working seamlessly to increase manufacturing intelligence, flexibility, agility and production efficiency. The objective of this study is develop and demonstrate a middleware software architecture to interface physical machines on a shop floor with client manufacturing applications. We have connected both legacy and modern 'smart' machines to a highly scalable database capable of storing streaming time-series data generated by on-board sensors and machine controllers. Three client applications were developed to demonstrate the mechanism through which third-party apps can be written without direct physical communications with machines on the shop-floor. The first, is an application that resides within the Digital Manufacturing Commons (DMC) which demonstrates the ability to query data from any physical machine on the floor; the 2nd application demonstrates a python app which compares digital product data with machine generated data; and the 3rd application demonstrates building a LabView app built to interface with the middleware service. This proposed architecture enables an ecosystem of smart manufacturing applications to be built and deployed on the shop-floor through open-sourced software and hardware devices thereby reducing cost of manufacturing software development.

Peer-review under responsibility of the organizing committee of the 45th SME North American Manufacturing Research Conference

Keywords: Digital Manufacturing; MTCONNECT; PostgreSQL; Digital Manufacturing Commons (DMC)

* Corresponding author. Tel.: +1 (972) 839-2255
E-mail address: bstarly@ncsu.edu; james.barkley@uilabs.org

Peer-review under responsibility of the organizing committee of the 45th SME North American Manufacturing Research Conference
doi:10.1016/j.promfg.2017.07.093

1. Introduction

The integration of digital manufacturing technology across the product lifecycle is making its way down to the physical machines on the production floor due to advancements made in hardware and software solutions in manufacturing plants [1,2]. Its understanding critically hinges on the ability to securely and easily capture, transfer, and analyze real-time streaming data from production machine tools to central IT systems. While many modern machine tools possess sensing and control systems, the data communications and digital interfaces are frequently complex and/or proprietary. The lack of plug-and-play type digital integration is an obstacle to achieving seamless digital operation of these machines within the manufacturing enterprise. While new CNC machines are MT-CONNECT enabled, many legacy machines will require hardware devices along with specific code to translate data to the MT-CONNECT standard. Once machines are connected on a shop-floor and integrated to a factory level IT system, insights can be generated to conduct shop-floor and enterprise level analytics. Such analytics in turn can help to inform both engineering and business decisions [3,4].

Manufacturers are beginning to respond to opportunities presented by the digital revolution in design, fabrication, production and service [2]. Many large manufacturers are finding value in using data analytics to optimize factory operations, boosting equipment utilization and product quality while reducing energy consumption. Such analytics can only be performed when machines connected on the shop-floor are streaming data in quantities and rates far beyond current data collection methods. In work that led to the MTCONNECT standard [5], Vijayraghavan and Dornfield et. al, demonstrated use cases for sharing data seamlessly in a common format, including virtual verification and tool position analysis. During initial development, basic relational database systems was primarily the data structure holding data streams from the machine. However, traditional relational database systems are not compatible for storing streaming machine data on the order of kilohertz frequencies. In subsequent work by NIST and others in the community, a volatile data stream (VDS) was developed based on a non-structured data schema implemented through MongoDB [6]. The advantage of using a non-structured data schema is that it allows data retrieval from several disparate sources of information such as CAD file formats, quality inspection data and machine data streams and then integrated together for a system wide view. Traditionally such high volume data had been stored in proprietary formats in shop-floor level plant historians maintained by software vendors. Such closed source nature of file formats increases the barrier for third-party software apps to be developed. Open-source high-throughput databases along with open machine-to-machine communication standards can enable innovation that open to any competent software developer. This specific paper looks into the building a middleware architecture enabled by a structured data schema but suited for storing high frequency streaming machine-generated data.

Current advances in industrial communication protocols, such as in TCP/IP, Profinet, MTQQ, have made it possible for shop-floor software solutions to capture high fidelity streaming data at rates beyond 1Hz [7]. In this scenario, three motivating factors arise: 1) If multiple types and size of machines are live streaming data from sensors and its control systems, traditional relational database systems, such as MySQL, may not be sufficient to write such high velocity data without expected performance degradation. 2) How can we enable an ecosystem of 3rd party apps be written to interface with a server achieve as much as possible hardware abstraction? and, 3) Would a middleware software stack enable faster plug-n-play type integration of apps which interface with the machines on the shop-floor? To address these questions, this paper describes a middleware software stack solution which allows 3rd party software app integration to manufacturing machines on the shop-floor while using a structured data schema.

Enabling third party software integration to building software for manufacturing machines drastically reduces the cost of building manufacturing software for the machines. Machine controllers such as those provided by Siemens and FANUC have necessary application programming interfaces (APIs) which allow communication into and out of the machine. This ability provides manufacturers the ability to stream data out of the machine and have it stored in a database either located within the enterprise or in the cloud. Wherever the data resides, a middleware architecture can enable third party software providers to build software apps that can scale to any manufacturer who may need them. This analogy is akin to how software apps are been built on the Android and iOS platform. Such middleware can radically change how software is built for manufacturing. In fact, commercial solutions such as GE Predix™,