



Data Capture and Visualisation on a Shoestring: Demonstrating the Digital Manufacturing on a Shoestring Project

Michael J. McNally^(✉), Jack C. Chaplin, Giovanna Martínez-Arellano,
and Svetan Ratchev

Institute for Advanced Manufacturing, Advanced Manufacturing Building,
University of Nottingham, Jubilee Campus, Nottingham NG7 2RD, UK
mike.mcnally@nottingham.ac.uk

Abstract. The adoption of digital manufacturing in small to medium enterprises (SMEs) in the manufacturing sector in the UK is low, yet these technologies offer significant promise to boost productivity. Two major causes of this lack of uptake is the high upfront cost of digital technologies, and the skill gap preventing understanding and implementation. This paper describes a common approach to data capture and visualisation that is cheap and simple. Cheap through the utilisation of low cost and readily available consumer technologies, and simple through the pre-defined flexible approaches that require a minimum of configuration. This approach was implemented on three demonstrators to showcase the flexibility of the approach. These were a tool condition monitoring system, a job and machine status monitor, and a robotic process monitor. The development process resulted in a software architecture where processes were separated and communicated by message queues. We conclude that a service oriented architecture would be the best system for carrying forward the development process. This research was conducted as part of the wider EPSRC Digital Manufacturing on a Shoestring project.

1 Introduction

SMEs in the EU manufacturing sector account for 99% of companies, 58% of employees, but only 41% of value added [1]. Narrowing the productivity gap between small and large enterprises is a strong motivator for improving access to digital technologies. In addition, according to the Made Smarter Report, effective incorporation of digital technologies will bring not only benefits to the UK industry itself (estimated £454BN over the next decade), but it is estimated that it will increase job satisfaction by 19%, significant CO₂ emissions reduction (estimated to 5.4 million tCO₂e), 70% less machinery breakdowns and 72,600 workplace injuries avoided [2].

However, as well as the obvious quantitative factors, there are qualitative reasons for SMEs to improve digital skills. For example, product passports have been proposed to enable greater product stewardship and end of life management [3];

SMEs risk losing access to supply chains without adequate digital technologies to participate. Conversely, with the right systems in place they have significant potential for transition to sustainable eco-production [4].

One barrier to adoption faced by SMEs is that of justifying the capital expenditure and a requirement for specialised personnel to implement and maintain the digital technologies [5]. A critical challenge when supporting SMEs on their digital transformation is to introduce new methods and digital solutions that are simple and low-cost, avoiding unnecessary investments of time and simplifying the business case [6]. In this context, the Digital Manufacturing on a Shoestring project aims to enable a wider uptake of digital technologies in SMEs by challenging traditional constructs of dedicated, pre-integrated technology to consider the viability of simple, light-weight and lower cost approaches and providing guidelines to implement these [7].

Smaller firms commonly rely on manual skills for assembly tasks in particular, where capital costs of specialised assembly tools are prohibitive for the scale of production: for example, PCB pick-and-place and reflow soldering equipment. In recognition of this, efforts have been made to introduce smart robotics for SME assembly and manufacturing needs [8], especially in complement to the traditional reliance on hand work [9,10]. An example of a traditional manual task is painting fastener heads, where a low cost approach has been demonstrated using a specialised robot tool head and machine vision [11]. Additional research has focused on providing knowledge resources to workers, such as augmented reality assisted systems [12] and the use of RFID to track progress of a part through a sequence of assembly processes [13]. Digitally enabled smart tools, like a low cost gap measurement microscope [14] offer new ways for workers to measure quality outcomes in assembly operations. Finally, widespread data capture is essential to take advantage of the emerging ‘big-data’ methods in precision assembly [15].

Underlying many of these technologies and systems is data capture and visualisation. In this paper we describe a flexible cloud-based approach to data capture and visualisation that is applicable to the Shoestring project and works with discrete event, low frequency, and high frequency data. To validate the applicability of the approach, we deployed the data capture solution to three existing platforms.

Demonstration platforms were selected to represent a common range of data collection problems in manufacturing with a variety of data types and frequencies. The base manufacturing platforms were chosen as examples of different environments: an assembly line, a cooperative robot-human work cell, and a Computer Numerical Control (CNC) milling machine. These were then used to validate the application of one or more of the digital manufacturing technologies identified as a priority by a series of focused workshops with manufacturing SMEs. The results of these workshops can be found in [16]. Data collection and visualisation represents an important component of many of the identified solutions, illustrated in Table 1.

In order to approach the problem in line with the Shoestring ethos, our demonstrators had some additional requirements. Firstly, the solutions implemented needed to be incrementally scalable, so each solution would have minimal

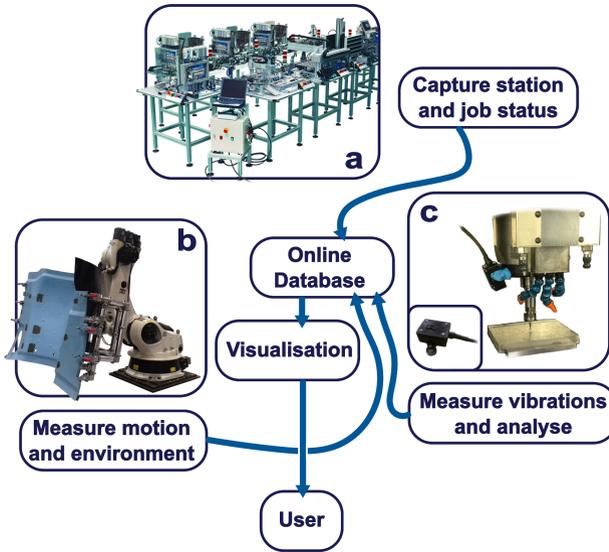


Fig. 1. Three demonstrators illustrate three different measurement types and use cases: (a) An assembly line produces discrete events representing the status of a particular station or specific part on the line. (b) A multi-sensor monitoring system measures the motion of a robot, the vibration and temperature of a held product and the environmental conditions in the robot bay, generating low frequency continuous time series data. (c) A 3-axis accelerometer mounted on a milling machine spindle produces high frequency continuous data which is processed to deliver low frequency vibration, status and condition data. All systems are connected to a cloud database and made accessible to the user through a visualisation front-end.

requirements for other systems to be put in place. Following from this, solutions needed to be inter-operable, to facilitate introducing additional functionality to the systems in discrete elements with minimum inter-dependency. Lastly, all demonstrators were based on low-cost commercial- off-the-shelf microcomputers (Raspberry Pi) and hobbyist grade preassembled electronic modules.

Three demonstrators were prototyped:

1. A tool condition monitoring (TCM) system on a milling machine.
2. A job and machine status tracker for an assembly line.
3. A motion and environmental sensing suite for a robot arm.

2 Architecture

The design and implementation of the proposed architecture was driven by the Shoestring vision, which introduced several constraints. Firstly, it should allow any implementation or solution to be incrementally scalable, so each solution

Table 1. The demonstrators deployed and the solutions which they fulfill, either fully (dark grey), or partially (light grey). Solutions listed here are selected from the top 15 solutions (of 52) identified through SME surveys [16].

	Tool condition monitoring	Job & machine status tracking	Robotic process monitoring
Real time tracking of internal jobs			
Capacity monitoring of resources			
Optimisation of machine setup times			
Predictive equipment maintenance			
Condition monitoring of equipment			
Process monitoring (vibration/energy/temp ...)			

would have minimal requirements for other systems to be put in place. This also required any solutions to be interoperable, so that it was as simple as possible to add additional discrete elements to generate new functionality.

Secondly, solutions needed to be cheap and widely used. Hence, all demonstrators used Raspberry Pi (*model 3B+*) microcomputers to log data from one or more sensors, and upload this data to the cloud. This introduced another limitation: the limited hardware resources of the Raspberry Pi and the non real-time operating system.

A final requirement was to use a common architecture for each demonstrator, so that in principle a new solution could be rapidly developed on top of this architecture.

The job status monitor recorded data from Programmable Logic Controllers (PLCs) from a Message Queue Telemetry Transport (MQTT) client over Ethernet, the robotic monitor had multiple modules all recording data from one or more sensors connected via the Inter-Integrated Circuit (i2c) bus, and the tool condition monitor recorded data from a sensor connected via Universal Serial Bus (USB). Data rates ranged from approximately 200 bits per second (bps) for one message per second over MQTT to roughly 1 Mbps for 3-axis accelerometer data encoded as byte strings and sent over USB serial. The higher data rates were sufficient to block the reading threads - in the case of the sensors on i2c, even lower data rates could block the thread as the sensors required continuous polling for data ready status.

To resolve these issues the program separate processes were used to read data, analyse data, assemble messages and then post data to the cloud server. The principle is illustrated in Fig. 2. Code was written in Python, and each thread was created using the multiprocessing library. This spawns a new instance of the python interpreter for each thread, which can now run in parallel on

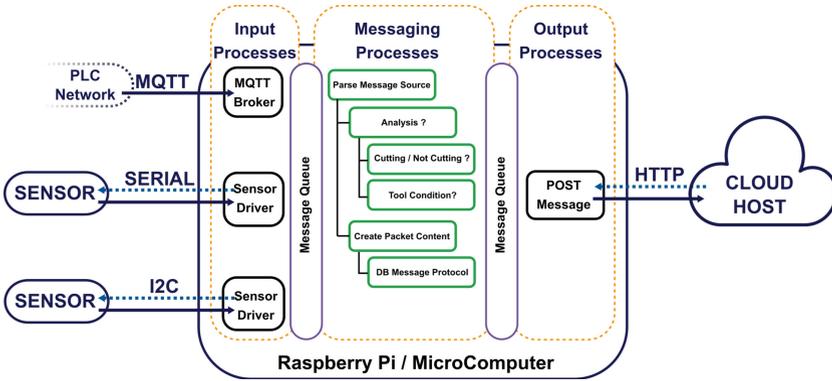


Fig. 2. Functional diagram of the sensor backend for all three demonstrators.

different cores of the Raspberry Pi, solving issues with limited processing power and thread blocking. This general approach also meant that new adding a new sensor or functionality simply meant writing a new Python process which could send and receive data on message queues.

One input process handles an interface, reads data from the sensor device, parses out the data and tags it with the sensor name, a key, and a timestamp. This data is put onto a shared message queue. The shared queue is read by the processing thread which parses the key and then takes appropriate action. After analysis, any results (e.g. cutting status) are put back on the message queue. All messages are eventually packaged into the correct string for database input and then put in another message queue. The output process sends all messages received out via http post to the cloud database.

Data is posted a database running on a cloud server. We used an InfluxDB database running on an Amazon Web Services (AWS) Elastic Compute Cloud (EC2) instance. All data can be posted as plain-text using the InfluxDB Representational State Transfer (REST) Application Programming Interface (API) line protocol. This requires more bandwidth than raw data but is more flexible for different data types and labelling schemes. It can handle continuous numerical data as well as discrete tags for different categories (for example, assembly line stations names or container numbers.)

The server also hosted a Grafana instance - an open source visualisation engine - which queried the InfluxDB instance for data. Queries were made with HyperText Transfer Protocol (HTTP) GET via InfluxDB REST API. Visualisations were manually built using the Grafana web interface, to summarise the status of each demonstrator. The full structure is illustrated in Fig. 3.

3.3 Job and Machine Status Tracking

The test-bed for job and status tracking was a Highly Automated System from SMC International Training (*SMC HAS-200*), which is a assembly line environment for training (Fig. 1a). The HAS-200 stations operate on barcoded containers, whose codes contain the product information and so are used to determine the station behaviour as each coded container moves through. The PLCs on each station were networked through an Ethernet switch, and were modified to output a simple MQTT message in response to: scanning a barcode and initiating the action, a change in station status, or a reject event (if a container failed a test.)

An MQTT broker was installed on a Raspberry Pi, to read MQTT messages, which were then parsed and the information posted to the database.

3.4 Robotic Process Monitoring

The robotics test-bed chosen was a KUKA industrial robot (Fig. 1b), modified with a flexible fixturing for aerospace panel handling, and programmed to manipulate the panel into programmed positions for a worker to undertake assembly tasks. This was fitted with two sensor modules, with a further unconnected environment sensor module nearby.

Each sensor module consisted of a Raspberry Pi, a 5400 mAh Li-Ion battery pack and the relevant sensors. The three different sensors were a motion module, a ranged module and an environment module, listed in Table 2. Sensors were chosen to plug directly on the Raspberry Pi General Purpose Input/Output (GPIO) pins and were predominantly from the Mikroe ‘Click’ family of sensor boards, which are low-cost hobbyist grade boards.

Table 2. Table of sensor modules used in robotic process monitoring, including parameters measured and the specific sensor products used.

Module	Parameter	Sensor
Environmental	Temperature ($^{\circ}\text{C}$)	pimoroni enviroPHat
	Pressure (mbar)	
	Colour (RGBI)	
Motion	3-axis acceleration ($\pm 2\text{ g}$)	IMU 6DoF 8 Click
	3-axis gyroscope	
	3-axis acceleration ($\pm 100\text{ g}$)	Accel 3 Click
Ranged	Laser Rangefinder	LightRanger 2 Click
	IR ranged thermometer	IRThermo Click

4 Discussion

The monitoring system has limitations owing to its nature as a proof-of-concept demonstration and prototyping platform. Firstly, the hardware implementation was not suitable for full industrial deployment. The wireless sensor modules on the robotic process monitor were battery powered from a 5400 mAh source. Based on specified current consumption of the Raspberry Pi and connected sensors we should not expect a lifetime over one day - unsuitable for a deployed application. However, many small, low-cost wireless sensor modules are available and a future deployment would use these, potentially with another device as a bridge to the cloud database, or to perform any analysis which is computationally expensive (and hence power hungry).

We used a cloud deployment for the database and visualisation. Though cloud data storage is commonly more secure than local data storage due to a robust backup policy and regular security updates (depending on the cloud provider) it's applicability to industrial circumstances does require some considerations. A future deployment will either need greater security or physical isolation from mission-critical industrial networks owing to the difficulty in securing and updating large numbers of embedded devices. This is simple when using deployed sensors on their own network but will require new solutions for systems similar to the MQTT monitor for assembly line monitoring.

A more prosaic problem with our specific implementation was the inflexibility of the database to variable demand. It could easily accommodate continuously writing data from multiple sources. During live demonstrations when multiple users were accessing data visualisations however, the database crashed due to inability meet multiple simultaneous requests for data. Future implementations will also need to take advantage of flexible demand management and scaling technologies to maintain functionality.

Finally, whilst the software programs are broadly similar for each demonstrator, each is a bespoke solution with the sensor drivers, MQTT client, and data parsing all coded in the same application. These could be separated into different programs for each function in a service-oriented architecture. In this schema, standalone Python software processes would be separated out into entirely different programs which could communicate via publish/subscribe message queues (*e.g. MQTT/Data Distribution Services etc.*). This is a mature architecture already proposed for low-cost industrial digitisation schemes [18].

5 Conclusions

We demonstrated how digital manufacturing technologies can be built from cheap hobbyist-grade sensors and computers, can be quick and easy to install in principle, and require no changes in the operation of the equipment to function.

The system we have implemented is a simple low-cost industrial monitoring solution. Operational and environmental data is gathered from a sensor system monitoring an aerospace panel fixturing, as a robot arm moves it through a

cycle of positions suitable for enabling a worker access to undertake assembly procedures. A tool condition monitoring system has been deployed on a 5-axis precision machining centre which can capture high frequency vibration data, and reduce this to machine status and tool condition data. A job and part status monitor has been developed to track part progress and station status on an assembly line.

All data from these demonstrators is captured in a common cloud database, with an open-source visualisation engine providing access to graphs and analysis of this data from any location. Data can be presented as live values, such as current factory floor temperature; time series, for instance, robot arm 6-axis motion; or historic operational data, like a chart displaying the time spent cutting, idle or powered off.

The demonstrators produced so far show data capture and analysis for three distinct use cases. However, a real manufacturing and assembly SME will require returns in terms of key metrics. To show value it will be essential to connect gathered operational and sensor data to product quality and assembly precision; process it into higher level inferences about machine utilisation, maintenance and downtime; and potentially to tie these into business tools to synthesis complex quantities like return on investment from a process change.

Our future work will investigate low cost methods for collecting data on product quality, and develop analytic tools for linking the finished quality data to condition data acquired during manufactured. Similarly we will aim to build tools linking manufacturing status data to business model, all within the Digital Shoestring ethos of low-cost flexibility, enabling SMEs to keep pace in a rapidly developing industry.

References

1. Eurostat. Key size class indicators, manufacturing (NACE section c), EU-28. Annual enterprise statistics for special aggregates of activities (NACE Rev. 2) (2019)
2. Maier, J., et al.: Made smarter - review 2017. Accessed 08 Jan 2020
3. Jensen, J.P., Remmen, A.: Enabling circular economy through product stewardship. *Procedia Manuf.* **8**, 377–384 (2017)
4. Schlegel, A., Langer, T., Putz, M.: Developing and harnessing the potential of SMEs for eco-efficient flexible production. *Procedia Manuf.* **9**, 41–48 (2017)
5. Mittal, S., Khan, M.A., Romero, D., Wuest, T.: A critical review of smart manufacturing & industry 4.0 maturity models: implications for small and medium-sized enterprises (SMEs). *J. Manuf. Syst.* **49**, 194–214 (2018)
6. Spena, P.R., Holzner, P., Rauch, E., Vidoni, R., Matt, D.T.: Requirements for the design of flexible and changeable manufacturing and assembly systems: a SME-survey. *Procedia CIRP* **41**, 207–212 (2016)
7. McFarlane, D., et al.: Digital manufacturing on a shoestring: low cost digital solutions for SMEs. In: Borangiu, T., Trentesaux, D., Leitão, P., Giret Boggino, A., Botti, V. (eds.) SOHOMA 2019. SCI, vol. 853, pp. 40–51. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-27477-1_4

8. Perzylo, A.: SMErobotics: smart robots for flexible manufacturing. *IEEE Robot. Autom. Mag.* **26**(1), 78–90 (2019)
9. Zheng, C., Qin, X., Eynard, B., Bai, J., Li, J., Zhang, Y.: SME-oriented flexible design approach for robotic manufacturing systems. *J. Manuf. Syst.* **53**(September), 62–74 (2019)
10. Zheng, C., Zhang, Y., Li, J., Bai, J., Qin, X., Eynard, B.: Survey on design approaches for robotic manufacturing systems in SMEs. *Procedia CIRP* **84**, 16–21 (2019)
11. Zhao, R., Drouot, A., Griffin, J., Crossley, R., Ratchev, S.: A low-cost automated fastener painting method based on machine vision. In: Ratchev, S. (ed.) *IPAS 2018. IAICT*, vol. 530, pp. 92–100. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-05931-6_9
12. Dassisti, M., Giovannini, A., Merla, P., Chimienti, M., Panetto, H.: An approach to support I4.0 adoption in SMEs: a core-metamodel and applications. *IFAC-PapersOnLine* **51**(11), 42–47 (2018)
13. Minoufekr, M., Driate, A., Plapper, P.: An IoT framework for assembly tracking and scheduling in manufacturing SME. *ICINCO* **2019**, 585–594 (2019)
14. Crossley, R., Ratchev, S.: Aerospace assembly gap measurement using low cost smart tools with machine vision. In: Ratchev, S. (ed.) *IPAS 2018. IAICT*, vol. 530, pp. 158–168. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-05931-6_15
15. Schmitt, R., Dietrich, F., Dröder, K.: Big data methods for precision assembly. *Procedia CIRP* **44**, 91–96 (2016)
16. Schönfuß, B., McFarlane, D., Athanassopoulou, N., Salter, L., de Silva, L., Ratchev, S.: Prioritising low cost digital solutions required by manufacturing SMEs: a shoestring approach. In: Borangiu, T., Trentesaux, D., Leitão, P., Giret Boggino, A., Botti, V. (eds.) *SOHOMA 2019. SCI*, vol. 853, pp. 290–300. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-27477-1_22
17. Martínez-Arellano, G., Terrazas, G., Ratchev, S.: Tool wear classification using time series imaging and deep learning. *Int. J. Adv. Manuf. Technol.* **104**(9), 3647–3662 (2019). <https://doi.org/10.1007/s00170-019-04090-6>
18. Hawkrigde, G., et al.: Tying together solutions for digital manufacturing: assessment of connectivity technologies approaches. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pp 1383–1387, September 2019

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

