Proceedings of the ASME 2022
International Design Engineering Technical Conferences and
Computers and Information in Engineering Conference
IDETC/CIE2022
August 14-17, 2022, St. Louis, Missouri

DETC2022-XXXX

IN-PROCESS DATA INTEGRATION FOR LASER POWDER BED FUSION ADDITIVE MANUFACTURING

Milica Perisic[1], Yan Lu[2], Albert Jones[2]

[1]Associate at National Institute of Standards and Technology, Gaithersburg, MD
[2] National Institute of Standards and Technology, Gaithersburg, MD

ABSTRACT

Additive manufacturing (AM) is a powerful technology that can create complex metallic parts and has the potential to improve the economic bottom line for various industries. However, due to process instabilities, and the resulting material defects that impact the part quality, AM still isn't as widely used as it could be. To overcome this situation, it is crucial to develop an environment for easy, in-process monitoring and real-time control to detect process anomalies and predict part defects as quickly as possible. AM in-process monitoring measures various process variables and the sensors generate large volumes of structured or unstructured, 1D, 2D, and 3D data, some of which are acquired at very high frequencies. Integration of such data and their analysis are necessary for effective in-process monitoring and real-time control, but they are facing many challenges due to the characteristics of AM in-process data. This paper provides an overview of different in-process monitoring data sources and their connection methods and addresses the integration issues associated with acquiring and fusing the data for both on-fly control and offline analysis. The paper also presents a guideline to help high-speed data integration. This guideline can help users to decide the best data-integration configuration for a specific use case.

Keywords: intelligent manufacturing, CAD/features technology, integration methods, data exchange

1. INTRODUCTION

Technology advancements have enabled the exponential growth of data that are being generated in the smart manufacturing field. Thousands of Exabytes of manufacturing data are generated annually and this amount is expected to grow [1]. Data provide insights for a better understanding of manufacturing processes and are critical for operation and business decisions. Consequently, data, as well as their analysis and control, have become a key aspect of smart manufacturing.

Nowadays, manufacturers are dealing with big data, which is far more challenging than the data managed and analyzed in traditional manufacturing. Big data refers to large amounts of multi-source, heterogeneous data, which are characterized by the 5 Vs: volume, variety, velocity, veracity, and value [2]. Integrating big data is needed for operation and business decision making and results in improved manufacturing performance and efficiency [28]. But existing, manufacturing-automation solutions face great challenges when integrating big data, especially when high-speed and high-dimensional data sources are present. Research engineers and industry practitioners are working to address these challenges.

Laser powder bed fusion (LPBF) additive manufacturing (AM) is a relatively new technology that generates significant amounts of data. This technology produces parts from computer-aided-design (CAD) 3D models by fusing together powdered material with a moving energy source. Unlike subtractive manufacturing, where quality control can be performed by controlling samples from a series of final products, AM quality control and part certification are required for every part due to many factors that can affect the part quality. LPBF-built part quality heavily depends on powder properties, machine's capabilities, process parameters, and their associated process characteristics. During the build, a material is being heated enough to change the material properties, which can lead to the part defects that can't be easily perceived without extensive quality control procedures.

Due to the multitude of factors that affect in-process stability and part quality, monitoring is critical for understanding and controlling an LPBF process. A wide range of sensors are instrumented on AM machines. These sensors generate large volumes of measurement data, which could be structured or unstructured, and at high frequencies. Successful integration of such data and their analysis will enable effective in-process monitoring and real-time control.

Currently, LPBF in-process data from various sources are often acquired independently and archived manually. Such isolated data acquisitions make real-time monitoring and decision-making hard. In addition, manual data curation relies heavily on reverse engineering and repopulating legacy or existing databases. Both processes are labor-intensive, error prone, and cost-inefficient. As a result, there is a need to streamline AM in-process, data integration, which is challenging for many reasons. Sensor data of high dimensions and sampling frequencies must be captured, stored, shared, and properly managed for quick analysis and easy querying. For example, melt pool images are generated with a frequency up to 20KHz. To perform real-time analysis and control, one must capture, analyze, decide, act, and store each image in 50 microseconds. Also, sample and event data generated by different systems are not synchronized in time, leading to additional difficulties in data integration and fusion.

This paper identifies the gaps that limit the integration of in-process AM data, and the requirements for data-source integration. It starts with an overview of varying data sources that are installed for LPBF AM process monitoring, covering the available interfaces, data acquisition characteristics, and use scenarios related to both on-fly control and offline analysis. To integrate the data, several steps are required, and the paper proposes options for implementing each step. The paper also provides a guideline to help users decide which data-integration configuration is best to follow for a specific use case.

The paper outline is as follows. Section 2 surveys what AM in-process data are collected and presents the challenges regarding their integration. Section 3 describes a methodology to address those challenges. Section 4 illustrates that methodology based on a specific use case. Section 5 discusses a use-case based integration requirements and standards issues related to integration. Section 6 summarizes the paper and proposes future work.

2. LPBF IN-PROCESS DATA SOURCE OVERVIEW

Large volumes of structured and unstructured data are generated through the AM part development lifecycle. Data that are produced from various measurements, are associated with material characterization, process monitoring, and part qualification. Integrating measurement data is critical for streamlining and accelerating part development and certifying parts for fast deployment.

The need for in-process monitoring in AM is motivated by the fact that a defect in any layer, if not detected and promptly corrected, could remain permanently embedded during the deposition of subsequent layers. These defects in AM are linked to anomalies in process control, chamber environment variables and process variables. A single sensor is not capable of detecting all these anomalies, such as deviating laser power, fluctuating gas flow and poor melt pool geometries. Nor is a single measurement data capable of controlling the process stability. Thus, researchers use heterogeneous sensing modalities. [3]

These heterogeneous sensing modalities have led to more diverse and richer AM data types. Table 1 below gives a brief description of nine different in-process data types, which are separated into three groups - Process Input, Environment Monitoring, and Build Behavior Monitoring. Table 1 presents their characteristics including data formats, sampling rates, data sizes and data usages.

As shown in the table, some commonly used, in-process, monitoring sensors generate up to 600 MB of data per second, so we need to deal with terabytes of data daily. Also, there is a variety of data types, from simple integers to complex images to 3-D tensors. Next to being different formats, they are generated and acquired by different systems, which makes integration more challenging. Data are being generated in frequencies of up to many kilohertz, which means that we have to process data on a microsecond level. For data to be useful, both data and metadata should be considered for integration. Metadata are data that are used to describe and give information about other data [27].

Process Measurements				
Measurement Type		**Data Format and Rate**	**Data Transfer Speed**	**Data Usage**
Process Input and Environment Monitoring	***Laser Beam Position and Actual Power***	Time series; ~100KHz	100MB/s	Machine/process anomaly detection[Krauss]; part defect diagnosis
	Chamber Monitoring	Temperature, pressure, humidity, flow measurement in Time series; ~1KHz	1MB/s	Machine/process anomaly detection; part defect diagnosis
Build Behavior Monitoring	***Acoustic/ Ultrasonic Emission***	Time series; 100Khz-10 MHZ	10MB/s-100MB/s	Process anomaly detection, part defect detection
	Melt Pool Temperature	Time series; ~100KHz – 1MHz	1MB/s-10MB/s	Process anomaly detection; Process physics study; feedback control
	Meltpool Imaging	2D images, sub 10-100 micron/pixel, small FoV; ~1-20KHz	14.4MB/s-288MB/s	Melt pool characteristics; Process anomaly detection; Process physics study; feedback control
	Exposure Optical Tomography	Captures the entire build space and measures the quality-relevant heat emissions in real time. ~10Hz	~100MB/s	Process anomaly detection; part defect prediction; part defect diagnosis
	Surface Morphology	2D images, surface height profile; ~10 um width, ~20-100 um height; per layer	600MB/layer	Powder bed defect detection; Part defect diagnosis; Part qualification
	Powder Bed Imaging	2D images; One or a few per layer	16MB/layer	Powder coating quality monitorin
	Tomographic Images – Optical coherence	A stack of images, penetrating 200-400um, from 512 × 480 pixels up to 1024 ×885 pixels; 150 frames/sec; per layer	320MB/s	Void, crack or un-melted powder detection [UK]
	High-speed synchrotron X-ray imaging and diffraction	X-Ray images; ~50KHz	3.28GB/s	AM process understanding; AM process modeling

TABLE 1: IN-PROCESS DATA CHARACTERISTICS

Sensor types and integration methods vary from vendor to vendor. Some offer monitoring based on photodiodes, others combine photodiodes and cameras. Still others use a pyrometer that is integrated directly into the build chamber. Each sensor type can be on- and off-axis. On-axis refers to the technique of installing sensors in the beam path of the laser. Off-axis refers to installing sensors onto the roof of, or directly inside, the build chamber. Most machine-embedded, in-process sensing systems have lower sampling frequencies compared to customer-built monitoring systems. For example, one vendor provides a camera that captures 10 frames per second, which gives images of the whole layer. [4]

Questions remain. How to integrate data with different sampling frequencies? How to select and collect useful data? How to process them after the collection? What are the available storages? This paper will present a guideline for high-speed data integration which can help users decide which data-integration configuration is best to follow for a specific use case.

3. IN-PROCESS DATA INTEGRATION METHOD

The challenges of automating data integration are the top roadblocks to AM real-time monitoring and control. Data integration is the process of combining data residing in different sources and providing users with a unified access and view of them. This section will present a guideline to help high-speed data integration by discussing existing, industry, integration standards and pointing out the missing ones associated with various manufacturing and enterprise applications.

Figure 1, which relies heavily on our earlier seven-step framework [7], shows four groups of activities involved in data integration: Data Identification, Data Acquisition, Data Processing and Data Archiving. When combined, these four groups provide a big-data, integration system that 1) enables real-time monitoring and control as well as 2) long-term data archiving for offline analyses. In addition to discussing these groups, this paper also discusses available standards and approaches from real-time, near real-time, and offline points of view.

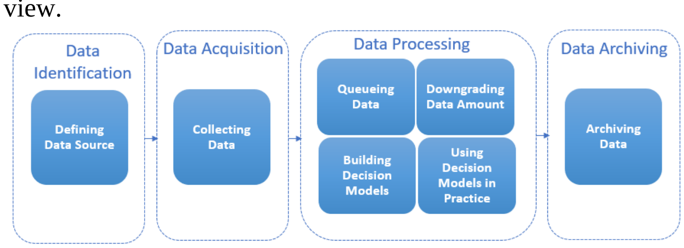

FIGURE 1: DATA INTEGRATION PROCESSES

3.1 Data Identification

In Data Identification, it's crucial to identify, define and characterize all the sources that generate in-process data. If these sources are not well defined and described, they can cause data integration and data usability issues throughout the whole AM process. For that purpose, a standardized definition can be created by providing three, important descriptions for each data source [7]: Source description, Data description, and Load description. Table 2 provides clear definitions of each.

Source Description provides information about the measurement devices being used for in-process sensing. This information highly impacts the type of data, which is defined in the Data Description part, and the available integration approaches. The Data Identifier field links to a specific data instance. Possible options include 1) AM system name or identifier, 2) device identifier, 3) the exact position of the device in the coordinate system of the source, and 4) a unique timestamp when a measurement is conducted. The final part of data definition is Load Description, which provides important information for data acquisition. This information is crucial for data integration.

Description	Data Definition Field	Data Definition Field Meaning
Source Description	Source Identifier	Identifier of a data source.
	Source Name	Name of a data source
	Source Type	Type of a data source
	Device Identifier	Identifier of a measurement device used.
	Device Type	Type of measurement device used.
	Device Manufacturer	Manufacturer of a measurement device used.
	Device Model	Model of a measurement device used.
	Device Configuration	Configuration of a measurement device used.
Data Description	Data Identifier	Identifier of a data instance.
	Data Category	How is data generated: Sample, Event, or Condition.
	Data Type	Type of data.
Load Description	Trigger Method	PUSH or PULL.
	Data Size	Size of one data instance.
	Protocol	Protocol used for data collection.
	Sampling Frequency	Frequency in which data is generated.

TABLE 2: GUIDELINE FOR DATA DEFINITION

Source Description and Data Description present metadata that can help users uniquely identify the specific data instance more easily. Load Description serves as a metadata also. But as noted, it's more important for data acquisition since it gives information about the data size and sampling frequency. Moreover, users can add their own metadata fields that can help identify their specific data instances more clearly. Other research presented a wide range of additional metadata fields for data registration for in-situ monitoring of LPBF processes [5].

3.2 Data Acquisition

In AM processes, in-process data are generated by in-situ, ex-situ, and machine sensors. New communication and sensor technologies have enabled ubiquitous, physical connectivity among intelligent devices, machines, sensors, and actuators [8]. This has resulted in many standards that can be used to collect data from the point-of-generation and pass them on to be further processed. Deciding on which standard to use highly depends on the use-case scenario. Comparing available protocols and standards for data acquisition provides insights for this paper.

Data generated from traditional field devices such as PLCs, process instruments, actuators and intelligent I/Os are collected using standards such as IEC 61158 [24], CAN [25], and Modbus [26]. These standards are not interchangeable, so they can create communication challenges when using IoT devices. Today, MTConnect and OPC UA are at the forefront of harmonizing data exchange between shop floor equipment and software applications. [8]

There are a variety of other standardized, session-layer protocols for IoT data exchange. The most important one is Message Queuing Telemetry Transport (MQTT). MQTT is a publish/subscribe protocol with minimal overhead and reliable communications. It is good for supervisory control and data acquisition (SCADA) and remote networks. It enables efficient data transmission, and it is quick to implement, due to its being a lightweight protocol. [8]

Acoustic and Ultrasonic Emission sensors have sampling frequencies that vary from 100KHz to 10MHz (see Table 1). For collecting this kind of data, a USB standard can be used. Deciding on which version of the USB standard to use depends on the data volume that is generated during the build process. Table 3 compares USB standard versions by data rate and transfer speed [16].

Standard	Maximum Transfer Rate	Maximum Data Rate
USB 3.0	5 Gbps	625 MB/s
USB 3.1	10 Gbps	1250 MB/s
USB 3.2	20 Gbps	2500 MB/s
USB 4	40 Gbps	5000 MB/s

TABLE 3: COMPARISON OF USB STANDARDS

During the LPBF process, a melt pool forms by laser beam irradiation on the metal powders, and then solidifies to the consolidated structure [15]. Melt pool monitoring is a necessary and important part of AM process monitoring and control. The easiest way to monitor its geometry and features is by using optical cameras that collect a variety of different images. Integrating these different images is challenging due to the combination of the high-sampling frequencies and the large amount of data they generate. The NIST AM Metrology Testbed [ammt.nist.gov] has instrumented a CMOS camera that generates melt pool images at 20KHz, where each frame is 120x120 pixels approximately 15KB in size. This leads to almost 300MB of just optical imaging data per second. Table 4 compares currently available standards for optical-imaging data acquisition.

Five standards for collecting data from industrial cameras address speed, cable length, receiver device and connector. Camera Link provides a speed of up to 850 MB per second when used with two cables [17]. CoaXPress provides high-speed rates and longer cable possibilities; because of that, it outperforms the CameraLink standard. The highest rate that it offers is more than 1500 MB per second with CXP-12 version [18]. For even higher demands, the CameraLink HS offers the highest speed of up to 2100 MB/s, with CX4 connector [19]. The last two standards differ from the previous one by having a direct connection to a PC. While there is no need for a frame grabber device, they will have a much lower speed rate.

These standards are suitable for cameras that are producing 125 to more than 400 MB per second of video or image data [20][21]. Which standard to use depends on both the amount of data that must be integrated and the decision regarding chosen receiver device. If the camera generates less than 125 MB/s then there is no need to invest in a receiver device because both, Gig-E and USB3 Vision can be used. If more data are generated, other options must be evaluated. Here, the decision will be made based on the speed and cable length needed.

Standard	Speed	Receiver Device	Cable Length	Connectors
CameraLink	255 MB/s for one cable and up to 850 MB/s for two cables	Frame grabber	7-15m	MDR 26-pin connector; SDR, HDR 26-pin connector (Mini Camera Link); HDR 14-pin connector (PoCL-Lite).
CameraLink HS	2100 MB/s	Frame grabber	15-300m	SFP, SFP+, CX4
CoaXPress	1562.5 MB/s	Frame grabber	30m	BNC Connector and smaller DIN 1.0/2.3
Gig-E Vision	125 MB/s or 250 MB/s with two cables	PC (direct)	100m (copper) and 5000m (fyber optic) using a single camera	Copper Ethernet; Copper Ethernet with vision locking screws; 10 Gigabit Ethernet direct attach cable; Ethernet fiber optic cable
USB3 Vision	437.5 MB/s	PC (direct)	Standard passive copper cable 3-5m; active copper cable 8+m; fiber optic cable 100m	USB3 Vision type connectors: host side (standard A locking) and device side (micro B locking)

TABLE 4: OPTICAL IMAGING DATA INTEGRATION STANDARDS

3.3 Data Processing

Data Processing is an essential part of data integration. It includes retrieving, transforming, or classifying raw, in-process, measurement data into useful information. During the data-processing phase, many activities are conducted, starting with collecting data from receiver devices to registering, fusing, analyzing, and sending data to long-term storage.

Type of Analysis	Batch	Streaming	ETL	ELT	Grouping Data	Parallel/ Distributed Computing	Lambda Architecture	Queueing
Offline	Yes	No	No	Yes	Layerwise Partwise	No	No	No
Real-Time	No	Yes	Yes	No	No grouping	Yes	Yes	Yes
Near Real-Time	Yes (micro batch)	Yes (micro batch or native)	Yes	No	Layerwise	Yes	Yes	Yes

TABLE 5: GUIDELINE FOR USING DATA PROCESSING APPROACHES FOR DATA ANALYTICS PURPOSES

There is no universal approach to processing high-velocity, AM measurement data. What approach should be followed depends on the use case. As noted, AM in-process data are collected and used mainly for control and monitoring. This paper proposes three data-processing approaches: offline, near real-time and real-time analysis. Table 5 presents a guideline for each approach. Based on the use case, users would want to perform one of the mentioned analyses. This guideline can help users understand what requirements they need to fulfill to conduct the intended analysis. Before discussing each proposed approach in detail, a short explanation of the methods and paradigms is provided in Table 5.

The batch-processing approach is used when a large volume of data is collected all at once [9]. The stream-processing approach is used when data are being generated continuously [9]. The ETL (extract-transform-load) data integration method includes 1) extracting data from various heterogeneous sources, 2) applying different process rules to the extracted data in the middle class, then 3) loading the transformed data into storage [11]. In the ELT (extract-load-transform) data integration method, after the extract process, stores data in a large storage device where scalable computations can be made [11].

Parallel computing on a single computer uses multiple processors to execute tasks in parallel. Distributed parallel computing, however, uses multiple computing devices to execute those same tasks. The Lambda architecture combines batch and stream processing paradigms to implement multiple paths of computation [9]. There is a streaming path for fast and, possibly, approximate results. And there is a batch, offline path that could be used for long-term archiving purposes [9].

3.3.1 Offline Analysis

Offline analysis has no time constraints. So, it can be conducted after a build is finished. AM in-process monitoring data are used to provide, or improve, the current understanding of AM processes through offline data analysis. Not having time constraints potentially relaxes the time requirements for data-integration when compared to real-time or near real-time data analysis. Still, executing that integration within a certain time constraint can be challenging, if a manual operation is involved. Figure 2 presents a proposed architecture for an offline-analysis data-integration approach.

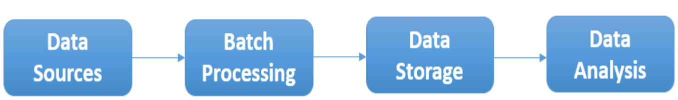

FIGURE 2: OFFLINE ANALYSIS DATA INTEGRATION APPROACH

Since there is no need for real-time data processing, the batch data-processing paradigm is a better option than the streaming one. It is an extremely efficient way to process large amounts of data that are collected over a long period of time, such as an AM build. But there is no need to process each data sample in real-time. A batch data-processing task can be scheduled as soon as the build ends. Or they can be scheduled, if needed, a few times during the build.

A distributed-computing approach is a useful way to split a single, large, data-processing task into several smaller ones. Each smaller task can be executed on a different computer, thereby speeding up the entire task. A Lambda architecture and queueing are unnecessary as well.

For offline analysis, it's easier to use the ELT integration approach to avoid performing high computational tasks on data before storing them – assuming there is available storage. If that's not the case, some transformations and data cleaning must be performed before sending data to long-term storage for archiving. After storing data, data registration and data fusion tasks should be performed. Researchers proposed methods for doing both, when 1) those data are melt-pool monitoring data [12] [13] and 2) the analyses are done on layer- and part-wise data.

3.3.2 Real-Time Analysis

Real-time analysis and control of AM processes is still in its infancy, but it continues to be an important AM goal and research topic. Figure 3 proposes an architecture for a data integration approach that can help perform real-time data analysis.

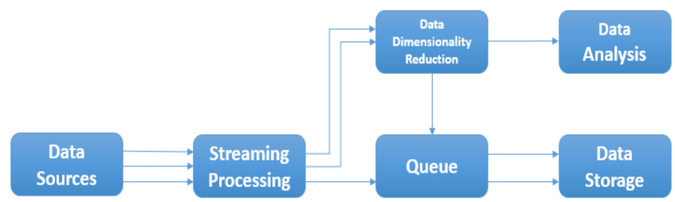

FIGURE 3: REAL-TIME ANALYSIS DATA INTEGRATION APPROACH

Clearly, batch processing isn't useful for processing high-speed, streaming data because it cannot process incoming events fast enough [10]. For that reason, a streaming paradigm should be used whenever every data sample must be analyzed. Images are one example, especially since they are collected continuously in real-time. The high dimensionality of the monitoring data makes it extremely difficult to get real-time analysis for many AM control applications.

To overcome this processing problem, distributed or parallel computing algorithms must be incorporated into the system architecture. Which algorithm to choose depends on the time available for one data sample to be processed. Section 4 presents a system where real-time analysis is achieved using a parallel-computing approach to process data with a frame rate of 2500 FPS.

In this use case, the Lambda architecture is proposed. Data are collected, then sent to a data dimensionality reduction component and a message queue. The data dimensionality reduction component serves for reducing the dimensionality of collected data. For example, it can calculate the design features in the data and sends them to the data-analysis component and to persistent, longtime storage. In parallel, to prevent overwhelming the storage, raw, high frequency data are sent to the message queue. From there, the queued data are used as inputs for offline analysis.

The proposed Lambda architecture associated with Figure 3 represents a form of edge computing. Edge computing plays an enormous role in using AM data because it is extremely fast. Edge computing takes parts of the computing process to various physical locations, moving the processed data close to where there are needed and collected. Edge computing is needed for control. But users need persistent storage for more sophisticated analysis, based on tools such as machine learning. Note, there are some new, vendor solutions that offer edge computing for process monitoring and cloud integration [14].

3.3.3 Near Real-Time Analysis

Near real-time analysis of AM build data allows some latency, which makes data integration slightly easier. The range for latency can vary and mostly depends on the use case. Latency can be defined by how many points, tracks, or layers behind data analysis can happen. The approaches for this analysis are the same as the real-time analysis approaches. So, Figure 3 is relevant for this use case. The difference in this case is that the streaming paradigm isn't always the only option. If time allows, batch processing could be also used; but only by processing the data in micro-batches. This will, to a certain point, imitate the streaming paradigm. For example, Apache Spark leverages micro batching for streaming, which provides near real-time processing [29]. Micro batching divides event streams into small batches and triggers the computations. In AM, latency is allowed to a certain extent, so data can be analyzed per layer.

3.4 Data Archiving

Data integration is considered finished when data are in a persistent storage. Subsequently, data fusion and software integration, which are outside the scope of this paper, happen. This paper will compare three types of persistent storage that can facilitate better data analytics: time series databases, data lakes and cloud storage. Table 6 presents a guideline on when to use each of the proposed storage methods based on user needs.

Time Series Database	Data Lake	Cloud Storage
Meaningful time-series measurement	Storing unstructured, semi-structured, and structured data	Storing unstructured, semi-structured, and structured data
Data analysis in specific time periods	Scalability and flexibility	Storing all data in one place
Storing image features, no images	Storing all data in one place	Easy data access by multiple users
Avoid queueing data during the integration	In-house solution	Easy maintenance and fast recovery
Fast data querying	Full control over data access	Cost-efficiency

TABLE 6: GUIDELINE FOR CHOOSING A PERSISTENT STORAGE

Time-series databases are a good option for dealing with IoT data. Most IoT devices collect data constantly and report that data at regular intervals. Time-series analysis can provide timestamped data points, making it possible to identify unusual behaviors during AM builds. Also, since all data samples have a timestamp value, it's easier to fuse various data sources if their acquisition systems are synchronized. Time-series data can be analyzed for a specific time period. Time-series databases can handle highly concurrent and high throughput "writes" so the user could avoid using message queues during data integration activities [22]. Data are organized in a way that allows users fast and easy data querying. The main disadvantage of using this type of persistent storage is that it's not recommended to store raw, image data for further analysis since the purpose of time series data is to monitor measures acquired in time. One can store image features instead of images, but that requires 1) calculating all possibly needed features before storing them or 2) storing raw image data in another storage, which causes data-federation problems.

The other two storages, Data Lake and Cloud Storage, are similar. Both can be used for storing unstructured, semi-structured, and structured data in one place [23, 24]. By choosing these storage types, one can avoid data federation problems and it would be easier to make connections between different data sources. Also, both storages provide unlimited scalability. What makes the difference in these options are cost and ownership. As an in-house solution, a Data Lake can be more flexible, but it requires knowledge to set it up and further administration. It also requires a higher investment at the beginning. Everything is developed in-house, so data privacy is higher than in cloud storage. Cloud storage is more cost-efficient, because users pay for the storage and for each executed query [24]. Moreover, and importantly, the maintenance and recovery, in case of failure, are the responsibility of the storage provider.

Data archiving comes with the challenges of synchronizing different frequencies and associating the different dimensions and resolutions of data sources. As noted previously, data can be joined based on the timestamp value, or added id value for each instance. For example, if a user has one coaxial image and three pyrometer values at the same time frame there are two options for the resulting data set. A data set could have one row with two columns: first column for image data, and the second one for some aggregated value of pyrometer values. Another option is to have three rows with repeated values for image column, and raw values of pyrometer data. More challenging is to associate different dimensions and resolutions of data sources and that represents a data fusion problem that was analyzed in the paper [30].

4. AN AM USE CASE

The data-integration ideas and processes presented earlier in this paper will be applied to the "Intelligent Metrology Architecture (IMA) for Metal Additive Manufacturing" proposed in [6]. This architecture helps extract significant features from the large amount of in-process AM data collected by the metrology systems (a collection of sensors). These features are used to effectively estimate the intermediate part quality. Here, integrating those sensor data is necessary to create the inputs for the virtual metrology software that analyzes in-process measurement data and makes layerwise, part-quality estimates. In this use case, that estimate is the surface roughness.

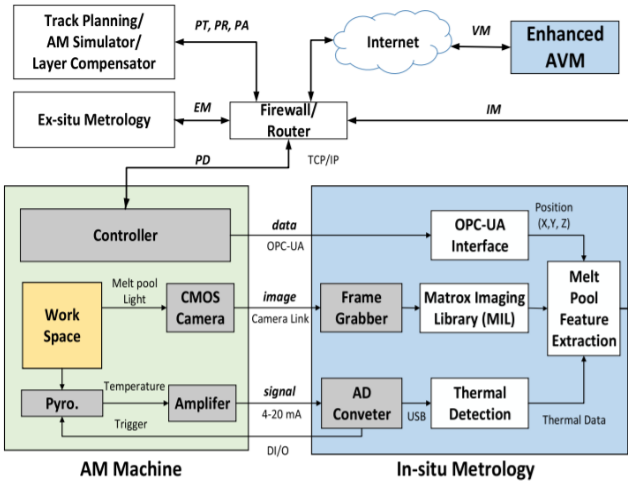

FIGURE 4: INTELLIGENT METROLOGY ARCHITECTURE FOR METAL AM [6]

Figure 4 shows the IMA architecture [6] for our AM use case. The main parts of the architecture are the AM Machine with instrumented sensors, In-situ Metrology for extracting features from data, and Automated Virtual Metrology (AVM) for analyzing those features. AM Machine has two sensors: CMOS camera and pyrometer. In-situ Metrology collects data from the sensors, extracts features needed for prediction and sends those features to AVM. AVM uses a model to predict surface roughness from the features sent from In-Situ Metrology part of the architecture. More details of the architecture are explained in the paper [6].

4.1 Data Identification

Two types of data must be integrated before they are analyzed for the purpose of estimating surface roughness and generating layerwise rescan strategy. As shown in Figure 4, the AM machine is equipped with a 500W laser with a wavelength 1070 ± 10 nm, a coaxial CMOS camera, and a pyrometer. The coaxial CMOS camera can capture images during a build at the rate of 2500 frames/sec; and each image size is 160x160 pixels. The temperature measured by the pyrometer is sampled at 100 kHz. Both sampled data series are uniquely defined by timestamps.

4.2 Data Acquisition

As noted in section 4.1, those two data inputs must be collected at 2500 frames/sec and 100 kHz; the first is for the coaxial CMOS camera and the second is for the pyrometer. Next, the actual, laser-spot position is collected. Yang [6] states that the laser-spot position is provided by the controller via an OPC-UA interface to the ISM module. In parallel, the image and temperature of the melt pool are captured using the CMOS camera and the pyrometer respectively. Images are collected using the CameraLink protocol; temperature signals from the pyrometer are in the range of 4–20 mA. After the analog pyrometer signal is converted to digital, the USB communications standard is used to send the resulting data to a data store. These data are used as inputs to data processing.

4.3 Data Processing

The first, data-processing task extracts features from the image data by using the Matrox imaging library. Details of feature extraction are described in [6]. After the image features are extracted, they are fused with current, laser-spot position and temperature data. Fusion is based on the corresponding timestamp in all three data sources. Data are then analyzed using the AVM component [6] to make surface-roughness predictions for each intermediate layer based on a predictive model.

Extracted features are stored in a relational database for the purpose of data analysis. Data analysis happens in both near real-time and offline. Offline, predictive-model training can be conducted on archived data. To create a training data set, a build was made to capture both in-process data and the surface measurement data from the as-built parts. Both the features of the in-process data and the surface measurement data are captured in the relational database and used for predictive model training. The resulting model is saved in the database as well for the AVM to make intermediate-layer, surface-roughness predictions. The estimation can be used for near-real-time, layerwise, decision making, for example, continue the build, pause the build, or rescan the layer.

Since this use case has a near-real-time, analysis component, the architecture's streaming approach is conducted, and data samples are processed in parallel. Data are sent directly for analysis in AVM component so there is no need for queuing data, which would only slow down the process. As noted, raw data are transformed (calculation of image features) and later fused with other sources, so this architecture presents an ETL process. The first step is data extraction from the sensors, then data transformation and loading into the storage.

4.4 Data Archiving

The architecture doesn't show storing of the raw data, only features extracted during the Data Processing. The goal of this architecture is to estimate the quality of AM parts during the build process; it doesn't focus much on the long-term archiving for future analysis. It is possible to create an additional architectural component that would archive the collected raw data before using them for offline analysis. Since data instances have a timestamp value, all three, proposed, storage options can be used. But, note, that original images shouldn't be stored in a time series database.

5. DISCUSSION

The general, data integration framework defines a workflow for AM in-process, big-data integration. The detailed configuration for each integration step depends on the application. To advance the AM process understanding, the current, process-monitoring strategy is to collect as much data as possible and as fast as possible. Better understanding of the fundamentals of AM processes will help identify which data are most important for part-quality control. Also, data analysis will help to evaluate the veracity and value of the collected data. These characteristics are interconnected because veracity impacts the value that data offer. Veracity and value can be evaluated during in-process data integration by, for example, using expert system rules, but the downside is that would affect the processing time. This represents a challenge for real-time or near real-time data integration approaches due to time constraints. Eventually, data integration requirements should be driven by use cases. Consequently, AM in-process data integration performance is going to be assessed based on the process control, optimization requirements, and the part yield rate in each use case.

Interoperability is a key issue for in-process data integration. Both AM machine builders and in-process, monitoring technology providers need to take responsibility for the integrability of their sensing systems. Standard communication protocols and information models should be developed and adopted in support of easy and cost-effective in-process data integration. In addition to standard data formats, standard metadata should be used to enable data sharing and reuse.

In addition, a standard reference architecture for in-process data integration will be helpful for part producers to adopt the framework, so are some reference implementations to provide best practices.

6. CONCLUSION AND FUTURE WORK

AM data integration faces many challenges due to the high volume, velocity, veracity, and variety that accompanies AM in-process data. These characteristics make automated data integration very challenging. To overcome the data integration challenges in LPBF additive manufacturing this paper proposed a guideline to help high-speed data integration. The guideline covers four essential processes for data integration – Data Identification, Data Acquisition, Data Processing and Data Archiving. Each process is discussed in detail, providing approaches to label data for further analysis, choosing appropriate standards for data acquisition, selecting what architectural approach to follow for data processing, and choosing the appropriate data storage based on the use case.

Future work will consist of identifying existing interoperable standards for data integration and the gaps to fill. We intend to develop a reference architecture, which can define the data integration functions and their interactions required for various in-process data pipeline setup. In addition, a reference implementation will be conducted at NIST to automate NIST AMMT data acquisition, transfer, and archiving, as well as to enable intelligent process monitoring and alarm management.

DISCLAIMER

Certain commercial systems are identified in this paper. Such identification does not imply recommendation or endorsement by NIST; nor does it imply that the products identified are necessarily the best available for the purpose. Further, any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NIST or any other supporting U.S. government or corporate organizations.

REFERENCES

1. Yin, S., & Kaynak, O. (2015). Big data for modern industry: challenges and trends [point of view]. *Proceedings of the IEEE*, *103*(2), 143-146.
2. Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey. *Mobile networks and applications*, *19*(2), 171-209.
3. Yang, H., Rao, P., Simpson, T., Lu, Y., Witherell, P., Nassar, A. R., ... & Kumara, S. (2020). Six-Sigma Quality Management of Additive Manufacturing. Proceedings of the IEEE, 109(4), 347-376.
4. Fuchs, L., & Eischer, C. (2018). In-process monitoring systems for metal additive manufacturing. EOS Electro Optical Systems.
5. Feng, S. C., Lu, Y., & Jones, A. T. (2020, September). Meta-data for in-situ monitoring of laser powder bed fusion processes. In International Manufacturing Science and Engineering Conference (Vol. 84256, p. V001T01A026). American Society of Mechanical Engineers.
6. Yang, H. C., Adnan, M., Huang, C. H., Cheng, F. T., Lo, Y. L., & Hsu, C. H. (2019). An intelligent metrology architecture with AVM for metal additive manufacturing. IEEE Robotics and Automation Letters, 4(3), 2886-2893.
7. Perišić, M., Milenković, D., Lu, Y., Jones, A., Ivezić, N., & Kulvatunyou, B. (2021). A Data Integration Framework for Additive Manufacturing Big Data Management. In 2021 International Solid Freeform Fabrication Symposium. University of Texas at Austin.
8. Lu, Y., Witherell, P., & Jones, A. (2020). Standard connections for IIoT empowered smart manufacturing. Manufacturing letters, 26, 17-20.
9. Carbone, P., Katsifodimos, A., Ewen, S., Markl, V., Haridi, S., & Tzoumas, K. (2015). Apache flink: Stream and batch processing in a single engine. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 36(4).
10. Pfandzelter, T., & Bermbach, D. (2019, June). IoT data processing in the fog: Functions, streams, or batch processing?. In 2019 IEEE International conference on fog computing (ICFC) (pp. 201-206). IEEE.
11. Haryono, E. M., Gunawan, I., Hidayanto, A. N., & Rahardja, U. (2020, November). Comparison of the E-LT vs ETL method in data warehouse implementation: A qualitative study. In 2020 International Conference

on Informatics, Multimedia, Cyber and Information System (ICIMCIS) (pp. 115-120). IEEE.
12. Yang, Z., Lu, Y., Li, S., Li, J., Ndiaye, Y., Yang, H., & Krishnamurty, S. (2021, August). In-Process Data Fusion for Process Monitoring and Control of Metal Additive Manufacturing. In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (Vol. 85376, p. V002T02A072). American Society of Mechanical Engineers.
13. Lu, Y., Yang, Z., Kim, J., Cho, H., & Yeung, H. (2020, November). Camera-based coaxial melt pool monitoring data registration for laser powder bed fusion additive manufacturing. In ASME International Mechanical Engineering Congress and Exposition (Vol. 84492, p. V02BT02A045). American Society of Mechanical Engineers.
14. THE ULTIMATE GUIDE TO MACHINE DATA And How IoT Platforms Help You Make the Most of It Whitepaper- add website
15. Yeung, H., Yang, Z., & Yan, L. (2020). A meltpool prediction based scan strategy for powder bed fusion additive manufacturing. Additive Manufacturing, 35, 101383.
16. USB: Port Types and Speeds Compared, https://www.tripplite.com/products/usb-connectivity-types-standards; [accessed on February 11, 2022].Camera Link Standard, Opto Engineering, https://www.opto-e.com/basics/camera-link; [accessed on February 11, 2022].
17. CoaXPress Standard, https://www.coaxpress.com/; [accessed on February 11, 2022].
18. CameraLink HS Standard, https://www.automate.org/a3-content/vision-standards-camera-link-hs; [accessed on February 11, 2022].
19. Gig-E Vision Standard, Opto Engineering, https://www.opto-e.com/basics/gig-e; [accessed on February 11, 2022].
20. USB3 Vision Standard, Opto Engineering, https://www.opto-e.com/basics/usb-3-0; [accessed on February 11, 2022].
21. Zhaofeng Zhou (2019). Key Concepts and Features of Time Series Databases, https://www.alibabacloud.com/blog/key-concepts-and-features-of-time-series-databases_594734; [accessed on February 11, 2022].
22. Miloslavskaya, N., & Tolstoy, A. (2016). Big data, fast data and data lake concepts. Procedia Computer Science, 88, 300-305.
23. Wu, J., Ping, L., Ge, X., Wang, Y., & Fu, J. (2010, June). Cloud storage as the infrastructure of cloud computing. In 2010 International conference on intelligent computing and cognitive informatics (pp. 380-383). IEEE.
24. IEC 61158-1:2019 Industrial communication networks - Fieldbus specifications - Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series, https://webstore.iec.ch/publication/59890; [accessed on February 11, 2022].
25. ISO 11898, “Road vehicles – Controller area network (CAN) — Part 1: Data link layer and physical signalling”, https://www.iso.org/standard/63648.html; [accessed on February 11, 2022].
26. Modbus Specifications and Implementation Guides, http://www.modbus.org/specs.php; [accessed on February 11, 2022].
27. Hodge, G. M. (2001). Metadata made simpler.
28. Tao, F., Qi, Q., Liu, A., & Kusiak, A. (2018). Data-driven smart manufacturing. *Journal of Manufacturing Systems*, *48*, 157-169.
29. Apache Spark. https://spark.apache.org/docs/latest/structured-streaming-programming-guide.html; [accessed on February 14, 2022].
30. Yang, Zhuo, Yan Lu, Simin Li, Jennifer Li, Yande Ndiaye, Hui Yang, and Sundar Krishnamurty. "In-Process Data Fusion for Process Monitoring and Control of Metal Additive Manufacturing." In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, vol. 85376, p. V002T02A072. American Society of Mechanical Engineers, 2021.