

Introducing an Edge Layer between Cloud MES and Shop Floor in Decentralized Manufacturing

Badarinath Katti
TU Kaiserslautern, Germany
katti@rhrk.uni-kl.de

Michael Schweitzer
SAP AG, Walldorf, Germany
Michael.Schweitzer@sap.com

Christiane Plociennik
DFKI GmbH, Kaiserslautern, Germany
Christiane.Plociennik@dfki.de

Abstract—The decentralized manufacturing is the research topic in current smart and open integrated factories. This is also the future state of practice in both the process and manufacturing industries. The Manufacturing Execution Systems is a comprehensive automation software solution that coordinates all the responsibilities of the modern production systems. These MES solutions are designed as a centralized manufacturing control units that goes against the principle of decentralized manufacturing paradigm. Moreover, they also experience the problem of connectivity and the network latency when operated as cloud based solutions. To this end, this paper addresses the problem of network latency in the context of cloud based MES by reducing the geographical distance between the shop floor and the MES with an introduction of an edge layer near shop floor. In other words, the production control data containing the routing details and other related data should be cached near the shop floor. Subsequently, the cached data enables the decentralization in manufacturing.

Keywords—Decentralized Manufacturing; Edge Computing; Cloud MES; Generic Shop Floor Connector.

I. INTRODUCTION

Traditionally in pursuance of lean manufacturing methodologies, simple tools like paper or spreadsheet based manufacturing was the norm. Even large companies followed this principle where line managers and operators walked the entire production line length to observe the work process. The production environments could not embrace the software that aid the smooth production process as the machinery in the shop could not support machine-to-machine and machine-to-business integration. The next stage evolution was computer aided manufacturing where home grown software was employed to plan, schedule and run the production processes. This phase was followed by a systematic production paradigm where the production was conceived to be a top down approach comprising of different layers such as Enterprise Resource Planning (ERP), Manufacturing Execution Systems (MES), Supervisory Control and Data Acquisition (SCADA) and Shop Floor.

However, with the advent of low-cost and smart sensors and subsequently Cyber Physical Systems (CPS), the sensors that are connected to the machines are now reachable as they have online capability. Thus, the MES can directly co-ordinate with the plant machines. This development has given rise to the possibility of omitting the SCADA layer, the responsibilities of which can be taken over by the MES. In many cases, SCADA systems and the connectivity solutions from the MES layer through the SCADA down to the shop floor have been characteristically vendor-specific. They do not follow industry standards and thus make it difficult to replace machines on the shop floor level. The trend of moving towards standardized

communication protocols on all layers of the automation pyramid is also fostering the development of circumvention of SCADA layer. The figure 1 illustrates the above explanation of transformation of automation pyramid.

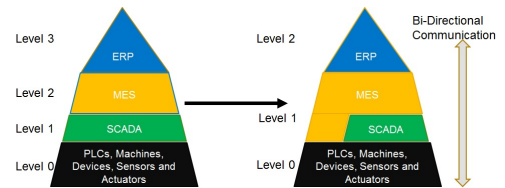


Figure 1: Evolution of classical Automation Pyramid

II. BACKGROUND

A. Centralized and Decentralized Manufacturing

In centralized manufacturing, a central entity is responsible for the entire system planning and hence, manages the operations at all stages of the manufacturing. The central entity takes unbiased decisions that are aimed at optimization of the objectives of an entire organization [1]. The centralized system is often complex in design which is tailor-made to solve a specific class of problems. The solution algorithms are problem specific and essentially complex, since the data is gathered from the whole system. In cases of unexpected events and product customizations, centralized systems have proven to be inflexible [1]. For example, if a product should change, it necessitates retooling of the entire system which is expensive in terms of both time and money. The decision-making process is delegated to the local decision-making bodies of the shop floor in decentralized manufacturing. At each step of the manufacturing process, a decision-maker is identified out of all the participating entities involved in the decision-making. The individual entity makes the decision with an express intent of optimizing its own objectives [2]. The decision-maker also acts speculatively to arrive at a decision with an assumption of the decisions of other entities. But the extent of such assumption depends on the degree of collaboration of different entities. This necessitates the coordination of the supply chain where the operational decisions and activities are shared accurately and in time bound manner with all the entities to avoid uncertainties. Furthermore, the decision variables of each entity are generally influenced by the decisions of other entities[3]. Both the above-mentioned modes of control in manufacturing are suited based on the manufacturing circumstances. For instance, the centralized manufacturing is preferred when the manufacturing process involves complex but static procedures [4] such as the determination of best locations for a set of

warehouses and cross-docks. When the manufacturing process expects dynamic changes such as varying customer orders and machine breakdown during the production, decentralized systems are preferred.

However, recent trends in manufacturing indicate an increasing tendency towards decentralized manufacturing. The decentralized systems are based on distributed control in which individual machines react to local conditions at real time. These machine components are linked to neighboring components to form a network that display the desired self-organizing behavior. The self-organization is the ability to tune proactively to improve the existing processes in response to environmental conditions [5]. Since the centralized system employ complex algorithms and analyze more information to arrive at a decision, this translates to slower response times in comparison to decentralized systems. Another advantage of decentralization is its property of provision of flexibility and robustness to the system. In a typical centralized system, a failure of central entity can potentially cause the catastrophic failure of the entire system [4]. The decentralized systems, on the other hand, tend to be more robust to the failures. The failure of the machines at the lower level of automation pyramid does not cause the whole system to fail. The decentralized systems adjust quickly and hence are resistant to perturbations. These systems continuously adjust in the pursuit of finding an optimal solution. These are also designed as highly redundant systems. These arguments evidently support the inclination towards decentralized control in manufacturing.

B. Functionalities of MES

The MES is a comprehensive automation control software solution that coordinates all the responsibilities of the modern production systems. The typical functionalities include sequencing the operations, monitor the production and to determine the states of different entities involved in production with respect to real time. The MES historically has been a vendor and industry specific solution and hence, is also called by other names such as Collaborative Production Management (CPM) and Manufacturing Operations Management (MOM) [6]. The horizontal and vertical integration in manufacturing improves the operation flow, which in turn increases the productivity. Such productivity gains are significant to the manufacturing companies to compete in the future and stay relevant in their fields [7]. The MES, which fundamentally is a centralized control structure, helps achieve this goal of the manufacturing companies. The MES also helps in better understanding the internal and external value chains in the manufacturing companies. The IEC 62264-3:2016 [8], which is an International Organization for Standardization (ISO) standard, defines activity models of MES in the view of enabling integration of enterprise system and the control systems. It divides the entire manufacturing operations management activities into the following four functional areas:

- Production operations management
- Maintenance operations management
- Quality operations management
- Inventory operations management

This paper focuses only on the production operation management aspect of MES.

C. Drawbacks of traditional MES

Historically the MES has mostly been an on-premise software solution, i.e., they are nursed close to production sites. MES is inherently difficult to own and maintain and even more rigid to evolve owing to the tight coupling of IT infrastructure to the manufacturing operations [9]. The self-owned MES is also difficult to enhance, in terms of business functionalities, in an efficient and time bound manner due to shortage of skilled resources. Both these factors are hindrances to the adoption of new technological innovations that enable complete horizontal and vertical integration. The inability of early adoption of new innovative technologies contributes to a return on investment (ROI) that is not commensurate with the manufacturers expectations.

The proprietary production control system that is part of the automation hardware is another facet of MES. These production control systems convert the ERP orders to technical production orders for the assembly lines. Such production control systems are tightly coupled to the machinery and hence, even a small change in production creates ripple effect across the automation layers. When the hardware and subsequently the production control software is discontinued, the future manufacturing maintenance is not safeguarded. In such situations, the implementation of expansions and changed requirements at the production execution is highly impractical and expensive.

A third classification of MES is tailor-made version from a third-party vendor who builds the MES as per the functional model specified by the manufacturer. The selection of an MES generally results in long term relationship with the MES vendor in the interest of protection of investment. The vendor guarantees long term maintenance and further development of MES modules, and integrate the future customer requirements in the product design and development. The continuation of status-quo after successful installation of MES is expensive since it involves upgradation of hardware components and IT solutions owing to their short innovation cycles. The additional difficulties such as platform dependency, license model, costs to maintain the MES software and work force that needs to be trained to use the software come to the fore and further increase the cost pressure on the manufacturers.

To that end, a detailed analysis of investment is necessary before investing in aforesaid solutions taking into the account the life cycle and cost of maintenance during the feasibility evaluation of an MES vendor.

D. Cloud based MES and Advantages

To address the above described difficulties, the traditional MES should be replaced by a comprehensive MES setup that can quickly adapt to the newer innovative technologies and offer the significant cost benefits to the manufacturer at the same time. The cloud based MES [10] is one such solution. The cloud based MES is a blend of various IT technologies such as distributed computing, internet technology, hardware virtualization and open source software. Cloud based solutions, in general, are best described as web based solutions that run on remote servers and accessed via internet on standard web browsers [11]. The cloud MES solutions are offered as IaaS (Infrastructure as a service), PaaS (Platform as a service) and SaaS (Software as a service) layers in the cloud architecture that are demand driven and charged as per usage [12].

The services in cloud based MES are generated by virtualizing and encapsulating the perceived manufacturing resources and capabilities [13]. Instead of building as individual projects, these MES Solutions are mostly assembled from configurable software components. The generic set of functionalities is built as per the customers requirements and typically, the functionalities provided by cloud based MES are richer than on-premise counterparts [14] and are also simple, fast and cheap [15]. Another main benefit of the cloud based MES is that it requires nearly no IT resource investment [11]. This lowers the entry costs for smaller firms that try to benefit from compute-intensive business analytics that were previously available only to the large corporations. This also lowers the IT barriers to innovation in the manufacturing processes [14]. The cloud based MES helps smoothly face the uninformed challenge of peak production demand without additional investment on on-premise resources [16]. The dearth of skilled resources that are acquainted with MES technology, achieving the ROI and technology compatibility are no longer the problems in the cloud scenario. Since the cloud servers are run as per the necessity, licenses can be increased or decreased accordingly. This decision need not be made upfront.

III. RELATED WORK

The manufacturing control supervises the progress of product as it is being processed, assembled, moved and scrutinized in the factory. Due to this complexity, high number of interactions between different components and the execution of variety of operations, manufacturing control systems were designed as hierarchical or centralized control approach [9]. With the rise of the consumer need for higher product variety with shorter life-cycles that are available at affordable prices and right time without the compromise on the quality, the concept of mass production was replaced by mass customization [17]. The tight coupling of the automation which exists in centralized manufacturing hinders the flexibility and self-organization skills that are required to stay relevant in the era of mass customization offered by smart factories. Owing to this rigidity and low receptiveness to changes in the manufacturing, centralized manufacturing practices were replaced by decentralized manufacturing models [18]. [19] proposes a decentralized work-in-progress (WIP) manufacturing control that serves as an alternative to the centralized manufacturing systems. But the research is carried out on the premise that machines in the factory shop floor at best are partially connected and the decision-making rests entirely on employees on the shop floor. The RFID enabled MES was introduced for mass-customization in manufacturing that faced challenges of manual and paper-based data collection, production plans and schedules [20]. However, the topic of this research is not just restricted to the idea of machine data collection, but also on the ways to accommodate MES in to the decentralized production paradigm. With the aim of operating on a global scale and to meet the demands of consumer market, the research was carried out on several paradigms for the factory of the future such as holonic [?], bionic [21] and fractal [22] manufacturing. To date, a huge volume of literature has been published on this subject. The agent based manufacturing [9] and holonic manufacturing introduced concept of artificial intelligence in manufacturing with an aim to respond promptly and correctly to change in production order. [9], [23] and [?] convey the idea

that the agent based manufacturing and holonic manufacturing are complimentary.

Despite the favorable view towards decentralized manufacturing, there are conflicting arguments towards the notion of decentralization. [24] recognizes that localization of decision-making with an obligation to decentralize has the risk of losing the global vision of the network. [25] argues that even though the decentralization of manufacturing is the norm in the future, there are cases where a centralized entity is obligatory to overwrite the lower level definitions in the event of redefinition of production processes at higher levels of automation pyramid. [26] also contends that the absence of central decision-making body necessitates continuous harmonization of objectives among the agents leading to high coordinative complexity. Interestingly, [1] claims that decentralized production planning results in loss of efficiency w.r.t. centralized production planning. [27] also supports the idea of presence of a centralized entity that overrides the decisions taken by lower level entities to achieve the global optimization under certain circumstances. Therefore, there is a renewed interest in incorporating centralized paradigm concepts to overcome the above-mentioned difficulties.

There have been several works, for example [28], [29], [30], [10] in the domain of cloud manufacturing that combine the emerging advanced technologies such as cloud computing, virtualization [31], internet of things and service oriented architecture. In a broad categorization, two types of cloud computing adoptions in the manufacturing are proposed, namely, direct adoption of cloud computing in manufacturing and centralized management of distributed resources that are encapsulated as cloud services [29]. The latter categorization is also known as distributed manufacturing. The potential as well as the relationships among the cloud computing, internet of things and cloud based manufacturing is investigated in [14]. It also proposes a reference architecture. [32] illustrates the concept of cloud based MES, but its application area is distributed manufacturing which is outside the purview of this paper. Although the flexible management of resource service composition that strives to enhance the information and intelligence in manufacturing [33] is based on centrally managed MES, it emphasizes on the geographically dispersed networked manufacturing model. Although cloud computing has gained wide acceptance during the past decade, there have been no research projects on production process planning at machine and execution control level, which is an imperative research area for developing a comprehensive Cloud based manufacturing environment for factories of the future [34]. Although cloud providers claim near 100% availability, there are instances in the life cycle of cloud solutions where the services are disrupted due to many reasons such as electric failure, hardware failure, cascading failure on routers and cloud downtime arising out of data center migration, server update against vulnerability et cetera. These incidences, on an average, reduce the availability to 99.91%, which in other words a non-availability of 7.884 hours per year [35]. Such network outages are not acceptable in the event of manufacturing a priority order. In general, the focus has shifted from centralized manufacturing systems - and MES in particular - with the advent of decentralized paradigm in the manufacturing. This research paper is novel in the aspect that it focuses on the adaptation of cloud based MES, which traditionally runs on

centralized manufacturing paradigm, to the context of decentralized manufacturing. In other words, it attempts to retain a degree of centralized aspects of manufacturing to strike the right balance.

IV. USE CASE

The direct client and server communication between the cloud based MES and shop floor, as illustrated in figure 1, encounters the network latency due to a variety of factors such as Nodal processing delay, Queueing delay, Transmission delay, Propagation delay and Packet Loss, and thus affect the throughput of the network. These delays are explained in the context of figure 2. The data packets are sent from source to destination via routers r_1 and r_2 . Each router has an incoming queue and an outbound link to each of the connected routers. The packet arriving at a router goes through the queue and the router determines the outbound link after examination of the packet header. An incoming data packet is immediately bound to outbound link if the router queue is empty and there are no packets being sent on the outbound link at the time. If the router queue is non-empty or the corresponding outbound link is busy, the incoming packet joins the router queue. When the data packet arrives at a router, the router examines the packet header for redirection to the appropriate destination. This causes a delay which is known as Processing delay d_{proc} and is the key component of network delay. The node also checks for bit level errors in the packet arising while transmitting from the previous node. After this nodal processing, the router directs the packet to a queue that precedes the outbound link.

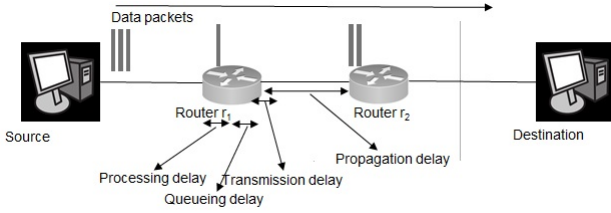


Figure 2: Illustration of network delays

The time a packet spends in the queue while earlier packets are transmitted at the node is called queueing delay d_{queue} . The incoming packet experiences zero queueing delay when the router queue is empty and no other packet is being transmitted by the router. Alternatively, the incoming packet experiences a queueing delay in direct accordance with the length of the router queue. The router transmits the data at a rate known as transmission rate. When the data packets arrive for a sustained period at a given router at a rate more than its transmission rate, these data packets will queue in at the router. To gain some insight, let A denote the average number of packets that arrive at the router queue per unit time. Let R be the transmission rate of the router; that is, it is the rate at which the bits are pushed out of router queue. For the sake of simplicity, suppose all the packets consist of B number of bits. Then, the average number of bits that arrive per unit time at the router queue is $A * B$. The ratio of $(A * B)/R$, called network traffic intensity, plays an important role in determining the queueing delay. If network traffic intensity is less than 1, the nature of arriving data packets influences the queueing delay. If a data packet arrives every A/R units of time, each

of these packets then arrives at an empty queue and will not encounter the queueing delay. Conversely, if the packets arrive in bursts due to traffic congestion, it then results in substantial average queueing delay. For example, assume P packets arrive simultaneously every $(A/R) * P$ units of time. The first packet that is transmitted will encounter no queueing delay. Nonetheless, second packet encounters a queueing delay of (A/R) units of time. Similarly, the third packet experiences a queueing delay of $2 * (A/R)$. In general, n^{th} data packet will experience a queueing delay of $(n - 1) * (A/R)$ units of time. However, the packet queueing does not follow a pattern in practical situations and the packets are spaced apart by an arbitrary amount of time. Therefore, the above quantity alone is not adequate to fully characterize the queueing delay. Nevertheless, it is a useful tool in the estimation of queueing delay.

The qualitative dependence of average queueing delay on the network traffic intensity is demonstrated in figure 3. It

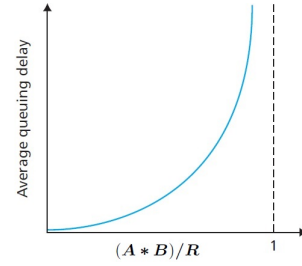


Figure 3: Dependence of average queueing delay on traffic intensity [36]

can be observed from figure. 4. that as the traffic intensity tends to 1, the average queueing delay grows exponentially. The queueing delays are typically of the order of microseconds to milliseconds [36].

When the packet arrival rate is greater than router transmission rate, the size of packet queue grows at the router. However, this cannot continue indefinitely due to the finite capacity of the router queue. Therefore, the router drops the packet when it finds no place at its queue. Such a dropped packet is lost and this phenomenon is called Packet loss. At this juncture, the client that transmitted the packet to the network core expecting the delivery acknowledgement from the server re-transmits the packet after waiting for a specified amount of time. This reduces the throughput of the network connection. The router takes a finite time to transfer the bits of a data packet onto the outbound link. This time is known as transmission delay d_{trans} and mathematically, it is defined as B/R . The packet on the outbound link propagates to the next node in a time known as the propagation delay. If l is the length of the physical link and v is the propagation speed of the data packet in the physical link, the propagation delay d_{prop} is then given by l/v .

The total nodal delay d_{nodal} is then given by,

$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop} \quad [37]$$

If there are N number of similar routers between the source and destination spaced apart at equal distances, then the end-

to-end delay $d_{end-to-end}$ is measured as,

$$d_{end-to-end} = N * (d_{proc} + d_{trans} + d_{prop}) + \sum_{n=1}^N d_{queue_n}$$

where the last part of the above equation is sum of the queueing delays experienced at each of the routers.

The virtualization principle of cloud computing enables sharing and dynamic allocation of resources. The virtualization concept can be applied at different levels such as computer hardware, operating system, storage and network. This virtual network infrastructure also introduces its own series of packet delays and causes further performance degradation. The delay factors explained in this section form one of the important aspects during the design of network elements and have a direct impact on the network latency.

During the production execution, the shop floor constantly seeks the information from MES. The work stations at the shop floor request MES for routing details at every stage of the production. Each work station collects the operation, bill of materials (BOM), set points and other resource configuration details. Once this information is collected the machine is instructed on how to proceed with that step of the production process. Once that step of the production is completed, the work station informs MES the same along with the generated results. The MES then processes the results and accordingly sets the next operation of the production. This process continues until all the planned operations are executed to manufacture the planned component. During exceptional cases if the need arises, the routing path is changed, as instructed by MES, to accommodate the exceptional situations. For example, the work in progress is diverted to rework station if the concerns regarding the quality of the products are raised.

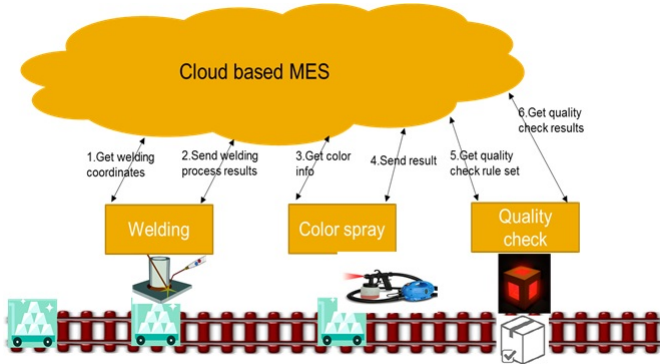


Figure 4: MES-Shop floor connectivity in production execution

The figure 4 illustrates this situation where there are three operations that are required to be performed to produce the planned component. The three operations are welding, color spraying and quality check. The work stations constantly communicate with MES to seek process parameters, machine configuration values and push the results during production control. The problem of network latency which is encountered each time the request is created to fetch the next operation details from MES does not auger well in high speed manufacturing scenarios. Furthermore, in the state of the art industries, the machines constantly question MES for recipe, task and the

corresponding set points that require quick response from the MES.

V. INTRODUCING AN EDGE LAYER

It is possible to achieve reliable packet transmission with bounded transmission delay only in the switched Ethernet with guaranteed sub-milliseconds delays [38]. As per the Open Systems Interconnections (OSI) model, Ethernet provides services up to and including the data link layer [39]. On the contrary, when the communication takes place over Wide area network (WAN), the transmission delay is no longer bounded. The WAN technologies generally function at the lower three layers of the OSI reference model. This includes physical, data link and network layers of OSI reference model. In case of cloud based MES, the data communication needs to take place between the shop floor and cloud based MES. As explained in section IV, the network latency is directly proportional to the geographic distance. The MES in cloud is not guaranteed to be close to the site of production. To this end, the problem of network latency in the context of cloud based MES can be addressed by reducing the geographical distance between the shop floor and MES. Hence, the data which is required for production execution should be close to the shop floor. In other words, this data should be cached near the shop floor. The cached data should contain the routing details and the production data that is required to manufacture the planned component.

Historically, functionality of MES was limited to data collection and tracking while the raw materials transform to finished products via proprietary SCADA systems. With the evolution of manufacturing technologies, the trend has been to move towards the standardization of the communication protocols. With this intention of adopting the standardized industrial protocols across the automation pyramid, this research paper proposes to replace vendor-specific SCADA systems with the *Generic shop floor connector* (GSFC) solutions that are not tightly coupled to the shop floor. Such GSFCs should control the production processes and collect the data to and from the shop floor and enterprise software. These GSFCs also help in enabling the plug and work feature of today's smart factory, since they can connect to wide variety of industry specific data sources of diverse manufacturers such as OPC UA, classical OPC and http based web services. The GSFCs are very close to the machines at the shop floor. Due to this physical proximity, the data communication latency between the shop floor and GSFCs is short as data packets need not cross multiple routers. Consequently, they are ideal places to cache the routing and production control data from cloud based MES. The caching strategy also facilitates the implementation of *logical decentralization* of the production execution.

Based on the above arguments, the cloud MES and the shop floor communication evolution can be illustrated as in figure 5. Once the production routing and production control data is cached, the intention is to reduce the communication between the GSFC and cloud based MES as far as possible. In other words, GSFC should take the control of the production process execution from MES. Several exceptional situations may arise in the shop floor while the GSFC is in control of the production execution. The GSFC should either resolve or find an alternative course of actions to the prevailing exceptional situations. The objective of this exercise is the successful



Figure 5: Evolution of MES-Shop floor connectivity

completion of the production execution. The cloud based MES should make it conducive for the successful completion of production execution.

A. Challenges of Integration of GSFC: A Survey

The GSFC should assume the role of the MES after the production order from MES is transferred to its cache. The transfer of production control to the GSFC is smooth under normal circumstances when the production encounters no problems. However, the system should be designed such that it should be robust against production fluctuations and should mitigate or solve the problems that may arise under exceptional circumstances of the production and reduce the system productivity.

To this end, several experts in the field of manufacturing were scientifically surveyed in person with the well-formed fixed response questionnaire administered by the author and their thoughts w.r.t. to the caching the production control data were collected. Taking into consideration the results of this survey, there could be several challenges that GSFC may face during the execution of production (shop) orders and that include but not limited to the following:

- Determination of next routing step as most of the business rules that govern the routing decisions are present in the MES
- Translation of business data arriving from MES to technology and business agnostic solution such as GSFC
- Adaptation in GSFC in the event of change of the data model in centralized cloud MES
- Determination of the suitable resources to perform the current operation
- Path substitution in the event of machine breakdown [9]
- Determination of course of action in the event of quality defects
- Determination of course of action in the event of unavailability of raw materials
- Dealing with the change of the production order [9]
- Handling the production orders of high priority [9]
- Distributed manufacturing where components are being manufactured at different sites

B. Overall System Architecture

The solution architecture should be designed taking into the account the challenges mentioned in section V-A. To this end, The cloud based MES should consist of following components in the context of facilitation of production execution:

1) *Production Planning System*: This application layer enables the production planner to plan the production sequence in a generic way. It has different maintenance user interfaces that help define the various product, operation and shop floor related master data. This master data enables the design of the shop floor routing for a product variant. This component also facilitates the production planner to create and release the production order to the shop floor.

2) *Service composition*: This component mainly has three sub units, namely, resource virtualization, virtual resource servitization and dispatcher. The remote resource sharing and management is a challenge to cloud based MES since it is geographically separated from the shop floor. The virtualization and service oriented architecture are the two enabling technologies that address the above-mentioned problem. In other words, virtualization technology is the key idea behind building the cloud services in the context of manufacturing. The resource virtualization is the transformation of real manufacturing resource to a virtual or logical resource. Each manufacturing resource is modeled formally with a set of inputs and outputs according to its main functionality. The functional and non-functional capabilities of the resource can be modeled using a semantic model. The model is then subjected to real-to-virtual mapping methods to map to a logical resource as illustrated in figure 6. The GSFC can query for the set points of a specific manufacturing resource using this model at runtime in the shop floor.

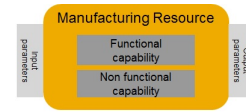


Figure 6: Resource Virtualization

The virtual resource servitization is the transformation of abstract concepts of capabilities provided by these manufacturing resources into formal services that are understandable by the cloud platform. This process involves several aspects such as definition of the service model, message model, ports and protocols. The service model includes the template for the service offered by cloud platform. The reception of inputs and generation of outputs of the service is defined in the message modeling process. Port modeling involves the definition of functional operation port used to accomplish the operation target. The protocol binding specifies the different protocols that are supported by the service. The production order created and released by the production planner is transferred from the MES to the shop floor by the dispatcher. The logic of transferring the priority order(s) is pre-loaded into the dispatcher. The parameters that expedite the release and subsequent transfer to the shop floor are production end date, priority customer, and inventory and manufacturing resource availability. The GSFC, introduced in this paper, is a technology and business agnostic solution. Therefore, the dispatcher should send the unambiguous data, for example, a collaborative product definition and operations semantic model. The GSFC translates this information to its compatible data model for further processing.

3) *Data mining and predictive analytics*: This component analyzes the current and past semi structured or unstructured data and extracts useful patterns and transfers this knowledge to GSFC. It is then helpful to solve or mitigate the problems arising in the shop floor during production.

4) *Information systems*:: This constituent stores the product genealogy including complete work instructions, components and phantom assemblies, operation flow and routing, manufacturing resources and work centers employed, bill of materials, activities on the shop floor, rework instructions and the discrepancies. This is realized using the Digital Object Memory (DOME) [40] which maintains all the information about a product instance over its production lifecycle, where each product is uniquely identified and tracked using RFID tag that contains the SFC number. Since DOME is centrally accessible to all the involved entities of production, it enables production coordination among these entities, compilation of the historic manufacturing report, quality investigations and process improvements.

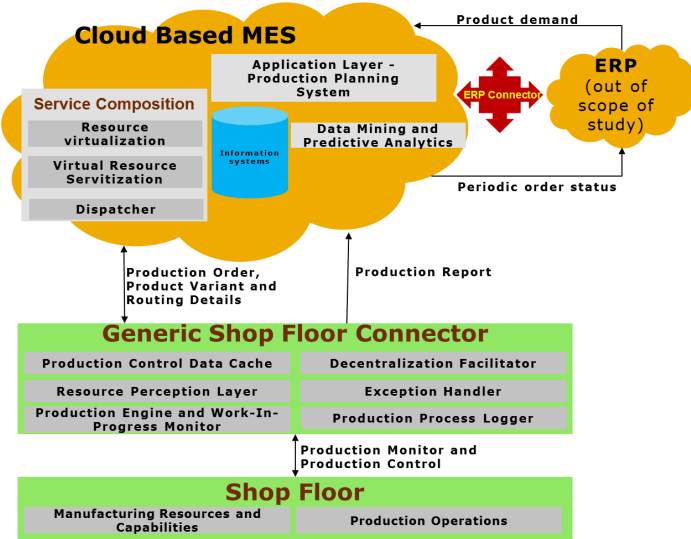


Figure 7: Integration of generic shop floor connector with Cloud based MES

C. Description of GSFC

In addition to providing the support for a variety of machine communication protocols of industrial automation, the GSFC should also consists of following components, each with a dedicated responsibility, to achieve the end goal of decentralization in production.

1) *Resource Perception Layer*: To achieve harmonization among various manufacturing resources, they need to be coupled together. The IoT technology is employed to perceive different manufacturing resources with an intent to enable intelligent identification, detection, communication, tracking, monitoring and management. The effectiveness of this exercise hinges on the ability of this layer to extract the key information from the real resources. The commonly adopted IoT techniques include RFID communication protocols for short distances and HTTP over TCP protocols for long range communication. The different manufacturing resources at the site also register

themselves as virtual resources to this layer. This data is transferred to decentralization facilitator component which enables it to take decisions at run-time.

2) *Production Control Data Cache*: This component stores the data delivered by the cloud based MES. It contains the blueprint of the production execution on the shop floor, which is the detailed routing information in the case of discrete manufacturing. Various entities of GSFC such as decentralization facilitator and production engine arrive at the decisions and actions based on this cached production execution data.

3) *Decentralization Facilitator*: This entity enables the decentralization in the manufacturing by coordinating with various manufacturing resources and cloud based MES. The layer maintains the virtual resource pool consisting of a collection of virtual manufacturing resources. It is used in run-time classification of resources that aids in on-demand resource capability matching. The virtual resource management helps GSFC identify capabilities intelligently by semantic searching of suitable services and the manufacturing resources on the shop floor to meet the production requirement.

4) *Exception Handler*: This block of the GSFC is accountable for overcoming any shortcomings that arise in the production environment. These shortcomings are explained in section 7.2. The exception handler either attempts to find alternate course of action by local coordination or seeks further instructions from the centralized entity which has global picture of the system.

5) *Production Engine*: In this context, the production order information and routing details are fetched from the production control data cache and the responsibility of matching the manufacturing resources for the given operation is delegated to decentralization facilitator. After the decision-making process, the production engine assigns the operation job to the real resources after the necessary configuration. To ensure the production is running as expected, it is necessary to monitor run-time status and respond to changes. In case of changes and exceptions, this layer coordinates with decentralization facilitator and exception handler to solve or mitigate the contingency. The production engine also has the intelligence to recognize the situations where GSFC cannot take the optimal decision based on local information. In such scenarios, it seeks the master data, the singular source of truth, stored in centralized cloud MES.

6) *Production Process Logger*: This component uploads the variety of knowledge it gathers during the production onto the cloud based MES. This unstructured data is subjected to analysis and an effort is made by cloud MES to find patterns and transform it into a structured data. This knowledge in turn can be provided as a feedback to the closed loop system in order to optimize the production in the long run. Based on the above arguments, the new automation architecture diagram, as illustrated in figure 7, is constructed which also shows how GSFC can be integrated into the scheme of things.

VI. IMPLEMENTATION(TO BEDISCUSSED)

The proposed solution need not necessarily be fully implemented for this paper. It would, however, be good if we could show the use case and how one or two challenges from above are addressed.

VII. CONCLUSION AND FUTURE WORK

This paper argues that the cloud based MES is suited to the changing production environments over the traditional MES solutions. It explores the problem of network latency associated with cloud based MES. To overcome this problem and also achieve the decentralization in manufacturing, an edge layer known as GSFC is introduced and a comprehensive architecture is designed to integrate this edge layer with the cloud based MES. The future work is further refinement in realization of decentralization, development of semantic model for the technology and business agnostic GSFC, research on the extent of caching under given conditions and handling of priority orders.

REFERENCES

- [1] G. Saharidis, Y. Dallery, and F. Karaesmen, "Centralized versus decentralized production planning," 2006.
- [2] M. Marques, C. Agostinho, G. Zacharewicz, R. Poler, and R. Jardim-Goncalves, "An architecture to support responsive production in manufacturing companies," 2016.
- [3] Hong, J. Ammons, and M. Realf, "Centralized versus Decentralized Decision-Making for Recycled Material Flows," 2008.
- [4] C. Anderson and J. Bartholdi, "Centralized vs Decentralized control in manufacturing: lessons from social insects," 2000.
- [5] M. Rolon and E. Martinez, "Agent-based modeling and simulation of an autonomic manufacturing execution system," 2011.
- [6] "Manufacturing Operations Management from Siemens," URL: <https://www.plm.automation.siemens.com/mom/> [accessed: 2017-08-29].
- [7] B. Wangler and S. J. Paheerathan, "Horizontal and Vertical Integration of Organizational IT Systems," 2000.
- [8] "IEC 62264-3:2016," URL: <https://www.iso.org/standard/67480.html> [accessed: 2017-08-29].
- [9] P. Leitao, "Agent-based distributed manufacturing control: A state-of-the-art survey," 2009.
- [10] F. Tao, Y. Cheng, L. D. Xu, L. Zhang, and B. H. Li, "CCIoT-CMfg: Cloud Computing and Internet of Things-Based Cloud Manufacturing Service System," 2014.
- [11] A. Lenart, "ERP in the cloud benefits and challenges," 2011.
- [12] J.-J. Hwang, H.-K. Chuang, Y.-C. Hsu, and C.-H. Wu, "A Business Model for Cloud Computing Based on a Separate Encryption and Decryption Service," 2011.
- [13] F. Tao, L. Zhang, Y. Liu, Y. Cheng, L. Wang, and X. Xu, "Manufacturing Service Management in Cloud Manufacturing: Overview and Future Research Directions," 2015.
- [14] S. Marston, Z. Li, S. Bandyopadhyay, J. Zhang, and A. Ghalsasi, "Cloud computing The business perspective," 2010.
- [15] W. Voorsluys, J. Broberg, and R. Buyya, "Introduction to cloud computing," 2010.
- [16] T. Wood, A. Gerber, K. Ramakrishnan, P. Shenoy, and J. V. der Merwe, "The Case for Enterprise-Ready Virtual Private Clouds," 2009.
- [17] P. BJ, "Mass Customization: The New Frontier in Business Competition," 1992.
- [18] K. Ueda, A. Lengyel, and I. Hatono, "Emergent Synthesis Approaches to Control and Planning in Make to Order Manufacturing Environments," 2004.
- [19] H. Loedding, K.-W. Yu, and H.-P. Wiendahl, "Decentralized WIP-oriented manufacturing control," 2003.
- [20] R. Y. Zhong, Q. Dai, G. H. T. Qu, and G. Q. Huang, "RFID-enabled real-time manufacturing execution system for mass-customization production," 2012.
- [21] K. Ueda, "A Concept for Bionic Manufacturing Systems Based on DNA-type Information," 1992.
- [22] H. J. Warnecke, *The Fractal Company*. Springer, 1993.
- [23] M.K.Lim and Z. Zhang, "A multi-agent based manufacturing control strategy for responsive manufacturing," 2003.
- [24] B. Montreuil, J.-M. Frayret, and S. DAmours, "A strategic framework for networked manufacturing," 2000.
- [25] M. Marquesa, C. Agostinho, G. Zacharewicz, and R. Jardim-Goncalves, "Decentralized decision support for intelligent manufacturing in Industry 4.0," 2017.
- [26] D. Mourtzis and M. Doukas, "Decentralized manufacturing systems review: challenges and outlook," 2012.
- [27] B. Hubanks, *Self-organizing military logistics*. Cambridge, Mass., 1998.
- [28] F. Tao, L. Zhang, V. C. Venkatesh, Y. Luo, , and Y. Cheng, "Cloud manufacturing: a computing and service oriented manufacturing model," 2011.
- [29] X. Xu, "From cloud computing to cloud manufacturing," 2012.
- [30] D. Wu, M. J. Greer, D. W. Rosen, and D. Schaefer, "Cloud manufacturing: Strategic vision and state-of-the-art," 2013.
- [31] *Virtualization for Dummies*. Sun Microsystems, 2009.
- [32] P. Helo, M. Suorsa, Y. Hao, and P. Anussornnitisarn, "Toward a cloud-based manufacturing execution system for distributed manufacturing," 2014.
- [33] L. Zhang, H. Guo, F. Tao, Y. L. Luo, and N. Si, "Flexible management of resource service composition in cloud manufacturing," 2010.
- [34] L. Wang, "Machine availability monitoring and machining process planning towards Cloud manufacturing," 2013.
- [35] C. Cerin *et al.*, "Downtime Statistics of Current Cloud Solutions," 2014.
- [36] J. Kurose and K. Ross, *Computer Networking: A Top-Down Approach*. New York: Addison-Wesley. p. 41., 2013.
- [37] N. Weng and T. Wolf, "Characterizing Network Processing Delay," 2004.
- [38] J. Loeser and H. Haertig, "Low-latency Hard Real-Time Communication over Switched Ethernet," 2004.
- [39] "Data Link Layer (Layer 2), Charles M. Kozierok," URL: tcpipguide.com [accessed: 2017-08-29].
- [40] J. Hauptert, "DOMeMan : Repraesentation, Verwaltung und Nutzung von digitalen Objektgedaechtnissen," 2013.