

# CloudMan: A Platform for Portable Cloud Manufacturing Services

Soheil Qanbari\*, Samira Mahdi Zadeh\*, Soroush Vedaeei<sup>†</sup> and Schahram Dustdar\*

\*Distributed Systems Group, Vienna University of Technology, Vienna, Austria

{qanbari, dustdar}@dsg.tuwien.ac.at, e1329639@student.tuwien.ac.at

<sup>†</sup>Baha'i Institute for Higher Education (BIHE), Iran

soroush.vedaeei@bihe.org

**Abstract**—Cloud manufacturing refers to “as a Service” production model that exploits an on-demand access to a distributed pool of diversified manufacturing services and resources. It forms elastic and reconfigurable production lines, which enhance efficiency, by allowing optimal resource allocation in response to demand changes and market dynamics. This paper studies these challenges and proposes a portable cloud manufacturing platform, entitled “CloudMan”, aiming at achieving a portable deployment of cloud manufacturing services to any compliant distributed production line in the cloud. The stakeholders of CloudMan are detailed together with their API requirements, where each stakeholder has an interest in. Having this rigorous analysis in mind, we present a holistic architecture for CloudMan, as it considers the manufacturing data, material and event flow from sensors and shop floors, through services to end products. In architecting such platform, there is a lack of agreed standard for the portability and orchestration of manufacturing services, as well as their definition. The proposed platform incorporates OASIS Topology and Orchestration Specification for cloud Applications (TOSCA) policies, plans and templates as a mechanism for dynamic configuration, portability and deployment of manufacturing services across multiple collaborating manufacturers. Thereby, the architecture provides a set of abstraction levels for various types of manufacturing services in which encapsulates and addresses specific requirements to satisfy the needs of stakeholders.

## I. INTRODUCTION

Industrial sectors now seek for distributed manufacturing control systems that provide satisfactory, adaptable and robust systems, rather than optimal solutions that require several hard assumptions to be met [1]. They are inspired by the convergence of utility-driven opportunities of novel distributed technologies, such as cloud computing and Internet of Things (IoT), coupled with collaborative engineering and applications that are transforming the manufacturing processes like supplement, consumption, and production models over the web. This contributes to the ability of dynamic provisioning of elastically scalable manufacturing resources. Cloud technologies efficiently enable integrating geographically diverse production lines to share knowledge and collaborate through virtualization layers and well-defined interfaces. Thereby, this model facilitates the inter- and intra-factory communication and collaboration in cloud manufacturing environment. In this context, the distributed execution of shop floor jobs is applicable. Underneath all the technologies involved, the essence of cloud computing is the cornerstone of distributed and sustainable manufacturing. According to the definition of National Institute of Standards and Technology (NIST)

[2], cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. As thus, a cloud Manufacturing system is capable of serving multiple companies to planning, control and collaboration over the web. It makes production planning and manufacturing work in progress (WIP) distributable, controllable, composable and portable. In comparison to traditional production services, cloud manufacturing is more complex and encompasses a broader diversity of services to adapt to ever-changing *customer-specific* requirements. In particular, CloudMan services are subject to realize concepts of “Manufacture on-Demand” and “Manufacturing as a Service” to meet unforeseeable events such as change of orders in an on-demand fashion.

Moving forward, the aim of this paper is firstly to conduct a detailed analysis on the CloudMan platform stakeholders; then to identify and present a set of architectural API requirements to meet the stakeholders needs in such environments. Secondly to propose an adequate CloudMan architecture by presenting the layers, including its service abstractions, with their associated core services. Lastly, we describe the major components and how they interact with one another to fulfill each requirement. CloudMan handles dynamic configuration, portability and deployment of manufacturing services across multiple collaborating manufacturers, by incorporating TOSCA standard. Henceforth, the product Bill of Material (BOM) is virtualized on the cloud and is referred to as Bill of Manufacturing Services (BOMS). The product designers model their products TOSCA-based BOMS by composing both physical manufacturing resource, e.g. devices, machines, sensors, materials, and the unphysical services, computing resources and capabilities e.g. product documents, data and required software services, device drivers using TOSCA standard. CloudMan deploys the distributed manufacturing processes and plans of such TOSCA-based BOMS. The paper ends with certain conclusions, focused on explaining the applicability of the proposed architecture for cloud manufacturing systems. However, at the moment, to the best of our knowledge, its not trivial for cloud and business developers to fully utilize manufacturing resources from multiple factories to realize the cloud manufacturing platform.

To this end, our contribution is twofold: (i) Incorporating

OASIS TOSCA standard for modeling cloud manufacturing Bill of Materials (BOM). (ii) A portable cloud manufacturing platform architecture together with its big data processing model.

The paper continues with the specification of portable cloud manufacturing services at section II. Next, we model the TOSCA-based Bill of Manufacturing Services (BOMS) at section III, to demonstrate the feasibility and applicability of TOSCA in our proposed framework. Section IV is devoted to the core stakeholders and their relationships in cloud manufacturing ecosystem. At section V, the API requirements associated with the corresponding stakeholders for CloudMan are derived. Subsequently, section VI presents a detailed view on CloudMan platform architecture, and a focus is put on implementing this new model with the API requirements and the supporting system architecture. Then, section VII presents the platform big data processing architecture. Section VIII surveys some scientific related work. Finally, section IX concludes the paper and presents an outlook on future research directions.

## II. PORTABLE CLOUD MANUFACTURING SERVICES

We define cloud manufacturing as a ‘*distributed manufacturing execution model, where underlying resources envisaged in the Internet of Things (IoT), are elastically exposed and utilized as cloud services, then composed and orchestrated for a manufacturing task in an on-demand fashion*’. The cloud virtualization technology abstracts away the complexity of underlying manufacturing physical resources and their associated operations from the product designer and developer. The IoT, takes the responsibility of controlling the sensor network and its data and event flow with the closer touch on shop floor operations. Collaborative engineering, addresses the orchestration and collaboration side of the manufacturing flows. Then the manufacturing services and encapsulated resources (devices, sensors, materials, drivers, etc.) are composed into the product BOMS that can be accessed, configured, invoked, deployed and coordinated on distributed production lines in a near real-time manner.

The monitoring service instances are responsible for sensing and interpreting the operational data received from the deployed manufacturing services across various units. If malfunctioning detected, the monitoring instance resumes the corresponding services, changes the context status and invokes recovery plan. One potential fault-recovery process can be incorporating portable cloud manufacturing services. Being able to port the whole package of dependent manufacturing services, serves better proactive maintenance of defects detected in one production line. The fault-recovery process will port the BOMS to other corresponding and authorized production lines and resume the job completion.

## III. TOSCA-BASED BILL OF MANUFACTURING SERVICES

The TOSCA-based BOMS of a configurable product contains all the components that are required to manufacture the product. It is composed of comprehensive list of raw materials, components and assemblies required to build or manufacture a product. BOMS is usually in a hierarchical format, with the topmost level showing the end product, and the bottom level

displaying individual components, resources and materials. Each item in BOMS describes the relationship between a parent (assembly) item and a child (component) item. In order to establish a cross-factory manufacturing governance and ultimately integrate distributed production lines, a standard shall be used to define product structure covering the relationship between objects, components and their configuration recipes.

Listing 1: Excerpt of a TOSCA-based Cloud Manufacturing Plan

```
<?xml version="1.0" encoding="UTF-8"?>
<Plans>
  <Plan id="DeployManufacturingService"
    name="Car_Manufacturing_Build_Plan(CMBP)"
    planType="http://docs.oasis-open.org/tosca/ns/2011/12/PlanTypes/BuildPlan"
    planLanguage="http://www.omg.org/spec/BPMN/20100524/MODEL">

    <Precondition expressionLanguage="www.example.com/text"> ?
      Invoke if Order is Paid.
    </Precondition>

    <PlanModel>
      <process name="Produce_10_BMW_X6" id="p1">
        <documentation>
          Deploy new instance of the CMBP to produce 10 BMW cars.
        </documentation>

        <task id="Task1" name="Check_the_Inventory_for_BOMS_Resource_availability"/>

        <task id="Task2" name="Ship_materials_to_shop_floor_from_warehouse"
          isSequential="true"
          loopDataInput="Task2Input.CarCounter"/>
        <documentation>
          Assumption: Task2 gets 10 as an Input, sets the CarCounter that
          indicates execution number of task.
        </documentation>

        <task id="Task3" name="Assemble_Car_Items"
          isSequential="true"
          loopDataInput="Task3Input.CarCounter"/>

        <task id="Task4" name="Colour_the_Car"
          isSequential="false"
          loopDataInput="Task4Input.CarCounter"/>

        <sequenceFlow id="s1" targetRef="Task2" sourceRef="Task1"/>
        <sequenceFlow id="s2" targetRef="Task3" sourceRef="Task2"/>
        <sequenceFlow id="s3" targetRef="Task4" sourceRef="Task3"/>
      </process>
    </PlanModel>
  </Plan>

  <Plan id="Halt_Car_Production"
    planType="http://docs.oasis-open.org/tosca/ns/2011/12/PlanTypes/TerminationPlan"
    planLanguage="http://docs.oasis-open.org/wsbpel/2.0/process/executable">
    <PlanModelReference reference="Mfg:HaltMfgSrv"/>
  </Plan>
</Plans>
```

There are couple of standards, which each does utilize the data exchange but lacks supporting the cloud and portability features. PDX<sup>1</sup> is the *Product Data eXchange* standard for the e-supply chain. Its standardization effort is focused on the problem of communicating product content information between Original Equipment Manufacturers (OEMs), Manufacturing Services providers and component suppliers. The standard is based on XML as a flexible way to encode structured data into a format that is both human and machine-readable. It provides a way to describe product content (BOM, Approved Manufacturer Lists (AML), Drawings, etc.), Engineering Change Requests (ECR), Engineering Change Orders (ECO) and Deviations in an XML format. This enables a total product definition to be described at a level appropriate to facilitate supply chain interactions. The standard is designed to transfer technical information including BOMS, AML, as-built product configuration, and change (Engineering, Manufacturing, Product) information, as well as manufacturing features, tolerance specifications, material properties and finish specifications. Another more in-use standard, called the Business-to-Manufacturing Mark-up Language (B2MML)<sup>2</sup> is an XML-

<sup>1</sup><http://webstds.ipc.org/standards.htm#x2570>

<sup>2</sup>B2MML (2003): Business to manufacturing markup language. The World Batch Forum, <http://www.wbf.org>

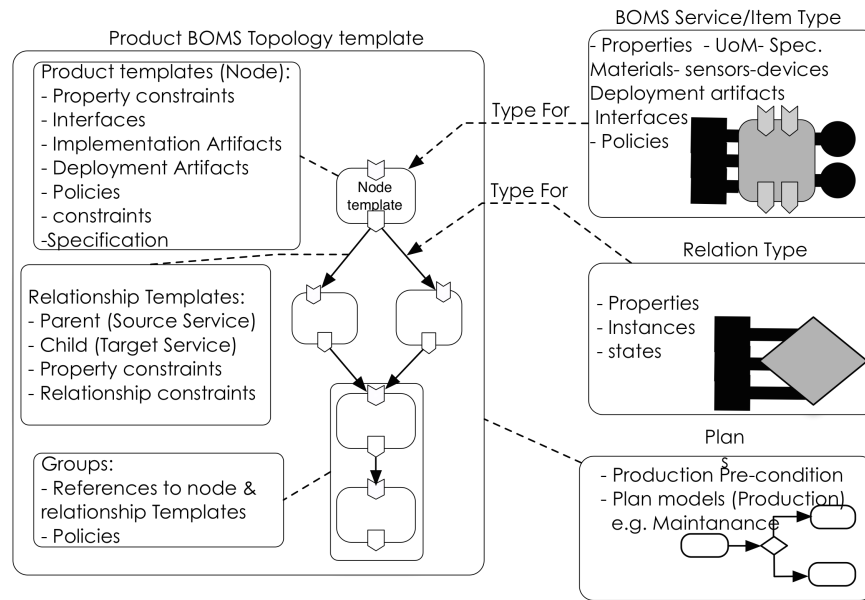


Figure 1: The core elements of TOSCA-based Bill of Manufacturing Services (BOMS) Template and the Topology.

based standard maintained by the World Batch Forum (WBF), which specifies data formats for information exchange between different enterprise control systems [B2MML03]. It provides an implementation of the ISA-95 standard in terms of XSD schema definitions.

Looking forward, having the essence of cloud computing in manufacturing domain, leads to an evolution in its manufacturing service delivery models. Cloud-enabled manufacturing services can be identified, specified, designed, composed, configured and realized or manufactured on-demand. Noted this, the BOMS can now be considered as cloud manufacturing virtual product, applicable to be deployed in multi-production lines, resulting in actual “as-built” assets. In this respect, the need for a cloud standard or specification to represent BOMS seems to be vital. The following offerings are observed and the appropriate one is ensured.

Cloud standards like TOSCA and OVF are for packaging the cloud services and resources as marketable entities. OASIS TOSCA approaches composing the cloud resources from the design perspective, rather than the actual resources that support the application, as OVF does. The OVF structure, on the contrary, emphasizes on what must be installed and how. TOSCA refers to OVF packages as deployment artifacts. A component of a TOSCA service can be implemented by deploying an OVF package. TOSCA’s approach is higher level than OVF. In fact TOSCA represents a business-oriented development methodology, which can be a significant achievement in IT-business alignment. Thus, TOSCA is a better choice in our mappings.

The TOSCA language introduces a grammar for describing service templates, by means of *Topology Templates* and *Plans* in which can be utilized to define the BOMS as well. The focus is on design time aspects, i.e., the description of manufacturing services, resources, materials or in broader view, “things” to ensure their exchange. The runtime aspects of TOSCA are addressed by providing a container for specifying models of

plans. In cloud manufacturing case, these plans can address supply chain plans, production plans, maintenance plans, etc. which can support the management of instances of manufacturing services. In fact, the designers will be more focused on the TOSCA-based BOMS topology design, which is a logical relationship between product components, equipments and assembly items. The TOSCA-based BOMS topology defines its item’s specification regardless of any explicit development and manufacturing operation details. With this approach, the manufacturing operations of a product can change without any effect to its BOMS design. Hence, both the manufacturer and the user are able to compare various actual manufacturing solutions and in terms of cost, quality and timely production to make a reasonable choice.

Figure 1, shows the meta model of the TOSCA-based BOMS. The root of a TOSCA-based BOMS is the *Product Template*. The product template contains a directed graph that represents the structure of the product called a *Product BOMS Topology*. Every product template has at least one product TOSCA-based BOMS topology. The topology graph is composed of nodes and edges. Edges in a directed graph are links with a direction from node to node. The edges in a product BOMS topology graph are binary relationships between nodes. The nodes represent the logical components, items or objects of the product. These nodes and relationships are templates that are patterns for the actual resources, objects or materials and their relationships instantiated in a deployed manufacturing service. Relations capture, represent and quantify associations between objects. Plans orchestrate various aspects of a manufacturing service life cycle. The TOSCA specification defines *Build Plans* and *Termination Plans*. Build plans orchestrate the deployment, installation and production operations of a BOMS. Listing 1 implements an excerpt of a TOSCA-based BOMS build plan in cloud manufacturing. The complete implementation of a TOSCA-based Bill of Manufacturing Services (BOMS) is available on

the CloudMan site<sup>3</sup>. Termination plans orchestrate decommissioning a production line. Designers of TOSCA-based BOMS can add plan types as needed. Therefore TOSCA-based BOMS encapsulates the required information for product life-cycle management. The utility of TOSCA-based product BOMS maximizes the portability of manufacturing document, and guarantees a smooth data flow.

In this paper, we propose CloudMan platform which deals with the realization and provisioning of all types of manufacturing resources as services, for all phases of the production life-cycle, from product TOSCA-based BOMS topology design to its distributed deployment. These manufacturing resources such as physical resources, software services and data units are classified in [3] and can be dynamically configured and composed into the TOSCA-based BOMS to be utilized through manufacturing services.

#### IV. STAKEHOLDERS IN CLOUDMAN

To investigate and elicit the requirements, we first detail a comprehensive study about the cloud manufacturing ecosystem stakeholders, who have business interest in this environment. Specifically, their participation roles, the services they provide and the APIs they consume. Their dependencies are drawn in Figure 2. However, we need to have a systematic classification of stakeholders in order to understand interfaces that they could be directly consuming. In our work, we classify stakeholders into six main groups of *Original Equipment Manufacturers*, *Platform provider*, *Product Developer*, *Manufacturers*, *Coordinators* and *Auditors* as illustrated in Figure 2:

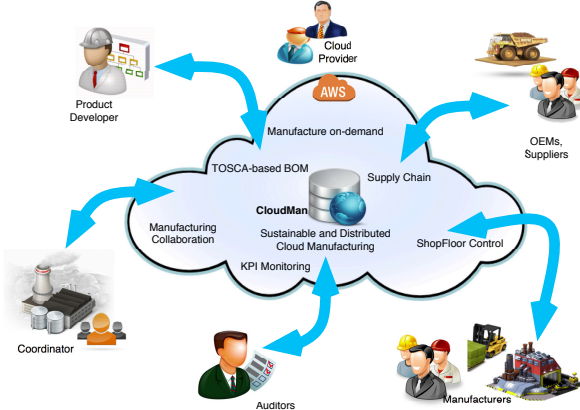


Figure 2: Main stakeholders in CloudMan Manufacturing System.

◇ *OEMs*: Original Equipment Manufacturers produce devices or components that can be consumed and utilized within other companies' manufacturing process. Their sustainable equipments can leverage and optimize the distributed manufacturing process. They can also provide logistics services and take charge of all the shop floor issues involved.

◇ *Cloud Platform providers*: They offer computing resources for enabling cloud manufacturing solution. They can be categorized as service providers (i.e., WSO2 ESB) or infrastructure providers (i.e., AWS EC2).

◇ *Product Developers*: In order to initiate a manufacturing process, BOMS should be modeled and designed. The product developers or designers process the user requests and then look for the most appropriate and applicable manufacturing resources that fit the request criteria. Then they model their product BOMS by composing both physical manufacturing resource, e.g. devices, machines, sensors, materials, and the unphysical services, computing resources and skills like product documents, product specification and required software services. In our platform, the product BOMS should be modeled using Oasis TOSCA standard and then the TOSCA-based product BOMS will be deployed to the distributed production lines for the actual manufacturing operations. The TOSCA-based BOMS components, will be mapped and deployed to actual services and manufacturing resources, devices and equipments at the production layer which then can be mapped, allocated and deployed on multiple production platforms.

◇ *Manufacturers*: They provide production lines with the associated and required human interactions in manufacturing at the shop floor. This workforce collaboration (required manpower) includes functions such as assembly, testing, quality control. Manufacturers may collaborate as well as compete.

◇ *Coordinators*: They take care of orchestration of cloud manufacturers in terms of serial and simultaneous production lines. In particular, they manage the portability and interoperability of manufacturing resources among multiple manufacturers. They also manage the resource and product delivery by dealing with SLA negotiation between manufacturers and a consumer. Specifically, they are hiding the diversity and heterogeneity of cloud manufacturers to consumers and coordinate a federated manufacturing.

◇ *Auditors*: In principle, they might be a third party partners, who watch and audit the manufacturing performance and the usage of resources on production lines through standard Key Performance Indicators (KPIs). Auditors make sure of SLA and the quality assurance of product delivery.

#### V. API REQUIREMENTS FOR CLOUDMAN

Our approach is to offer a cloud manufacturing platform and a set of APIs so that the product developers/designers can design their products seamlessly and circumvent the abstraction layers over the production lines and gain access to the original APIs for resource utilization. From the API engineering perspective, we delve into our perception of the APIs for the cloud manufacturing paradigm. To clarify the expectations of the stakeholders as consumers of these APIs, we classify them into three main categories: (a) TOSCA-based BOMS Formation APIs; (b) Resource Match-making APIs; (c) Manufacturing Provisioning APIs. Thus back to the developer of a cloud manufacturing product, they compose and connect diverse services, resources and raw materials from various manufacturers. They also make sure of clients' requirements to be satisfied by the chosen ingredients. we observe the following main API requirements for architecting the CloudMan:

##### A. RQ1. APIs Runnable on Multiple Cloud-enabled Manufacturing Systems

The goal is to provide a layer of abstraction between the cloud-enabled product BOMS designers and the multiple pro-

<sup>3</sup><https://github.com/soheil4TUWien/CloudMan>

Actors, Requirements →	RQ1	RQ2	RQ3	RQ4	RQ5	RQ6
Cloud provider (e.g. IaaS, PaaS)	•					•
OEMs (e.g. Devices, Equipments)			•	•		
Manufacturers (e.g. Workforce)		•	•		•	•
Coordinators (e.g. Control flow)	•			•	•	•
Auditors (Observe Mfg flow)				•	•	
Product Designer (Compose BOMS)	•	•	•	•		•

Table I: Relationships between Requirements and Stakeholders in CloudMan.

duction lines, which basically abstracts the differences among diverse manufacturers. Designers compose abstract TOSCA-based BOMS at this layer, which then can be distributed and deployed on multiple production platforms. This will grant the designers to an “on-Demand” and “at-Scale” manufacturing service provisioning on multiple and distributed production lines with a single platform.

#### B. RQ2. Discovering Product Resources and Materials through Query Interface

This requirement deals with manufacturing resource discovery procedure. First, it manages the resource discovery process among all the available manufacturers. Considering the *Publish-Subscribe* design pattern, the manufacturers publish their profiles in a knowledge base or manufacturing registry. Then the discovery agent broadcasts demand for the actual resource availability among multiple manufacturers. This is required for designing TOSCA-based BOMS.

#### C. RQ3. Match-making APIs to Map/Utilize TOSCA-based BOMS to Services to Production Resources

This requirement deals with production resource rating and reservation procedure. After having the manufacturing resources discovered, first the agent rates them based on the matching policy and deployment strategy, then chooses the more appropriate vendor/manufacturer and the resources, wherein the specification fits in upon requirements in terms of quality and cost constraints. Finally, the confirmed BOMS will be deployed for the actual production and will reserve the resource for future manufacturing provisioning on the shop floor. In summary, production resource matchmaking APIs provide users with a list of recommended resources indicating the best match between BOMS requirements and manufacturers’ offer.

#### D. RQ4. Ensuring end-to-end BOMS-centric Manufacturing Resource Coverage

The impetus behind the sustainable and distributed cloud manufacturing is driven by the dynamics of the market and ever-changing requirements and the need of product developers to optimize costs with the improved quality. Therefore, it is vital to ensure that the recommended and chosen resources and materials follow the constraints based on agreements and supports all the requested BOMS specification with the cost and quality demanded, while avoiding the dependency or lock-in to one vendor/manufacturer.

#### E. RQ5. Manufacturing Monitoring through multi-Production Lines Watch Interface

The platform should provide monitoring services for manufacturing flows running on multiple production lines. Developers can benefit of these APIs in order to collect data and track metrics, gain insight, and react immediately to keep their product development running smoothly. The match-Maker agent can also monitor and process custom Key Performance Indicators (KPIs) to gain cloud-wide manufacturing visibility into resource utilization, application performance, and operational health for future production planning.

#### F. RQ6. Managed Interoperability of Production Flow among Multiple Factories

The variety of the service interfaces design makes the product development and the hosted production lines dependent together, which poses the interoperability challenges between multiple manufacturing services. We need models for interoperability and orchestration of the services designed for cloud manufacturing environments.

As listed in Table I, CloudMan requirements are directly associated with a stakeholder, who consumes an API and performs a specific operation to the fulfillment of manufacturing requirement.

## VI. CLOUDMAN PLATFORM ARCHITECTURE

This section presents the CloudMan architecture that realizes the aforementioned core concepts. Before documenting the architecture, we study the term Manufacturing Execution System (MES) defined by the Manufacturing Execution System Association (MESA)<sup>4</sup> as follows: “Manufacturing Execution Systems (MES) deliver information that enables the optimization of production activities from order launch to finished goods. Using current and accurate data, MES guides, initiates, responds to, and reports on plant activities as they occur. The resulting rapid response to changing conditions, coupled with a focus on reducing non value-added activities, drives effective plant operations and processes. MES improves the return on operational assets as well as on-time delivery, inventory turns, gross margin, and cash flow performance. MES provides mission-critical information about production activities across the enterprise and supply chain via bidirectional communications.”

As a blueprint for designing the architecture, we briefly review the information flow between the MES and various

<sup>4</sup>www.mesa.org

heterogeneous connected systems as defined by ISA-95<sup>5</sup> standard. The Product Lifecycle Management (PLM) sends product information like its bill of materials, work instructions, equipment configurations, operations list and their execution order to the MES for the operational purposes. This information has been defined using B2MML(Business-to-Manufacturing Markup Language)<sup>6</sup>. In our architecture, we model this PLM information using TOSCA standard. The MES sends the manufacturing test results to PLM, then sends the performance results together with produced and consumed resources to the legacy applications like ERP system. On a shop floor side, the work instructions and recipes are sent from MES to Programmable Logic Controllers (PLC) or Gateways for actual operations. As a result, production data are sent back to MES. The MES layer, which is responsible for managing the factory, sits below the ERP, manages the business. The industry standard for plant floor connectivity and process control is called Open Platform Communications (OPC). The standard specifies the communication of real-time plant data between control devices from different manufacturers.

The architectural model proposed for the CloudMan platform as illustrated in Figure 3, is organized on five interconnected layers: (i) Manufacturing Vertical Applications Layer, (ii) Manufacturing Core Services Layer, (iii) Manufacturing Execution System Layer, (iv) Manufacturing Service Bus Layer and (v) Manufacturing Infrastructure Resource Layer. Next is focused on each layer's specification and capabilities. In the architecture big picture, we also provide a sample WSO2<sup>7</sup> service which can implement the layer's capabilities.

#### A. Layer 1. Manufacturing Virtual Applications (MVA) Layer:

At this layer, clients are involved. CloudMan provides interfaces for users to invoke the services for their business needs, like on-demand manufacturing requests. Following the request, required services will be composed and orchestrated to fulfill the order. Business users can also create web components like dashboards or gadgets to have a close look at their ongoing manufacturing orders. This layer simply represents the "face" of the CloudMan.

On the other side, the manufacturers can benefit of their enterprise store to advertise their manufacturing services, APIs and applications, so that the users can rapidly discover, subscribe and make use of them upon their demands. CloudMan is capable of outsourcing the manufacturing processes in an on-demand fashion using the cloud gateway service. This opens a secure channel to other manufacturing services in the cloud for more elasticity and collaboration. Last but not least, 3rd party applications like ERP/CRM can also be integrated to the CloudMan and render their manufacturing orders on cloud. They can model their requirements by product definitions, bill of materials, work instructions and the equipment settings. In return, CloudMan will respond with the manufacturing results, including the finished products, material consumptions and metering data for charging, billing and future planning purposes.

<sup>5</sup>ISA-95 is an international standard for developing an automated interface between enterprise and control systems, specifically for global manufacturers.

<sup>6</sup>The B2MML standard defines a format for exchange of ISA-95 information and the specific method (XML documents) for exchanges.

<sup>7</sup><http://wso2.com/products>

#### B. Layer 2. Manufacturing Core Services (MCS) Layer:

This layer is the foundation and "body" of the architecture, which serves all parts of the ecosystem. It relies on scalability of distributed service bus. At this layer, the Enterprise Service Bus (ESB) is in place to act as connecting layers and implement interactions among various components within layers. ESB delivers monitoring, routing and queuing of messages plus choreography and queuing of events. Another responsibility of this layer is the Identity Server, which provides the authorization service over resource usage, grants the authorized users to access the manufacturing service catalog and invoke registered services in Governance Registry.

This layer also deals with monitoring manufacturing system performance upon predefined KPIs. The monitoring mechanism consolidates the collected data around KPIs for the manufacturing system and provides insights to the stakeholders for their future planning. Monitoring is applicable on various granularity of the manufacturing units. On a higher level of granularity, the web services that are exposed for internal usage, should meet the performance criteria, like availability and response time for the prospective manufacturing load. Looking at the production level, for instance, manufacturing resources like equipments and devices (e.g., 3d Printer) running on the shop floor, need to be checked periodically to ensure their healthy operation, such as the resource status, its energy consumption and the resource utilization. These monitoring tasks might lead to a change in manufacturing flow and request the production line to elastically scale and adapt by allocating more or releasing manufacturing resources to newly detected situation.

#### C. Layer 3. Manufacturing Execution System (MES) Layer:

MES layer represents the "brain" of the CloudMan. MES receives the product definition, TOSCA-based BOMS, the manufacturing instructions, the equipments configurations and finished product quality constraints from the MVA layer or from 3rd party ERP/CRM applications. Then this layer checks the availability of manufacturing resources and does the inventory and supply chain control through the MSB layer. Having that in mind, the MES schedules the production planning and reserves the resources through Match-making APIs furnished in requirement section accordingly. The resource scheduling process deals with the distributed allocation of planned manufacturing operations associated to all orders on the available resources. This layer is also capable of defining rules and constraints for performance measurement and quality assurance.

As scheduled, the production plan will be deployed on the production lines, allocating resources and invoking manufacturing services leading to actual shop floor operations. Through Complex Event Processing (CEP), CloudMan processes events, including material consumption; resource usage and product assembly will be tracked and analyzed, then the production results will be sent to upper layers for aggregation and computation. The data services in this layer will store, retrieve and manipulate the manufacturing data to straightforward integration with manufacturing flows.



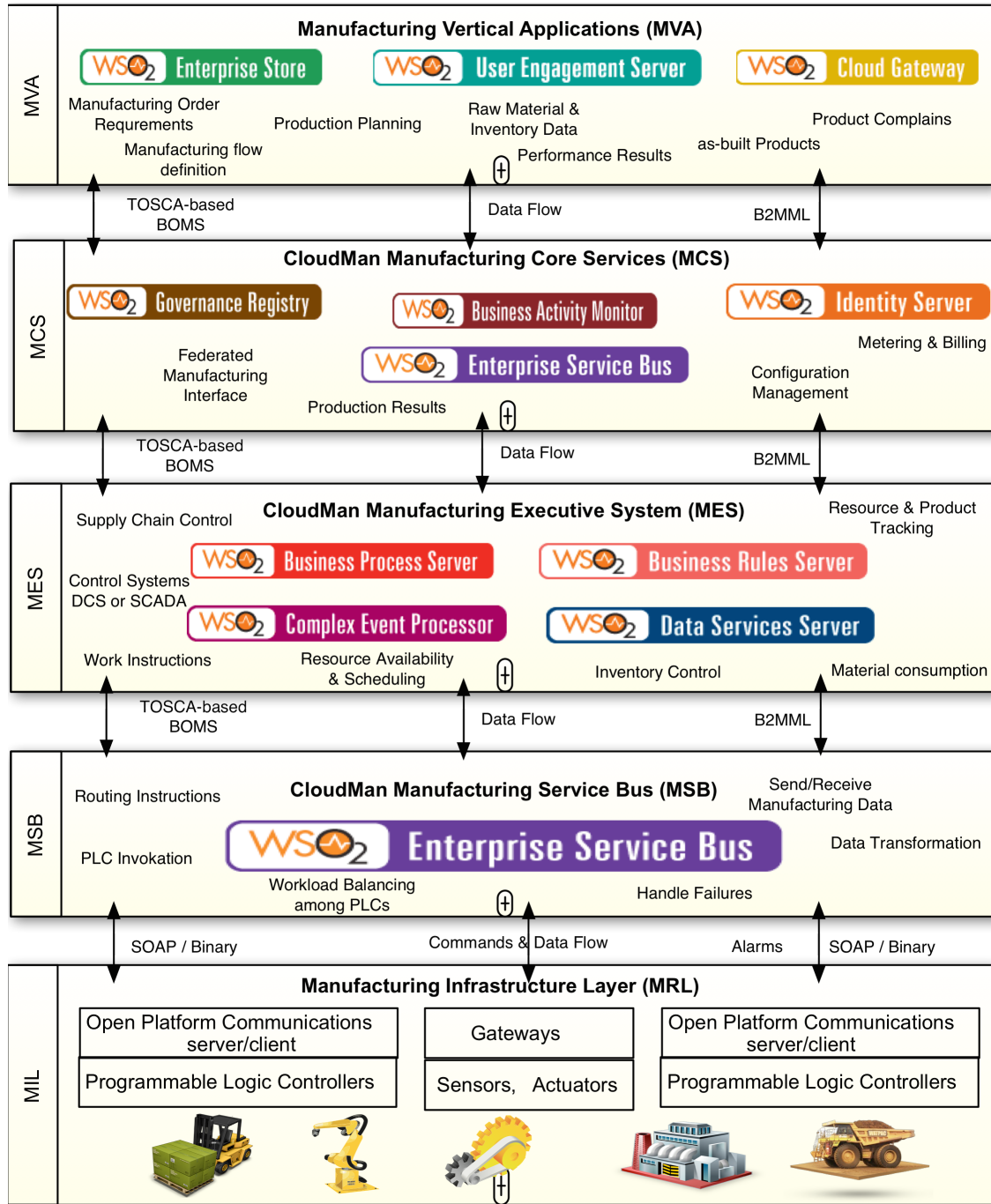


Figure 3: CloudMan Framework Layered Architecture

#### D. Layer 4. Manufacturing Service Bus (MSB) Layer:

After the production scheduling is confirmed, the execution commands and allocation messages are sent from MES layer to MSB. The MSB receives the messages/events, then after analyzing them decides which end points (e.g., machine, equipment, human) is the recipient, so that it will route the message like instructions to the desired receiver. It is in fact the “heart” of the CloudMan as it pumps pure manufacturing messages, data and events into various end points. MSB handles the transactions and can load balance among multiple

PLCs to perform under high loads. It is also responsible for handling the production failures in case of occurrence.

#### E. Layer 5. Manufacturing Infrastructure (MI) Layer:

This layer encapsulates manufacturing resources like materials, machines, equipments, devices, sensors and human workforce. In CloudMan, the resources are connected, sensed and can be controlled through IoT technologies. Then these resources and their operations are virtualized and encapsulated into registered cloud manufacturing services, which will be

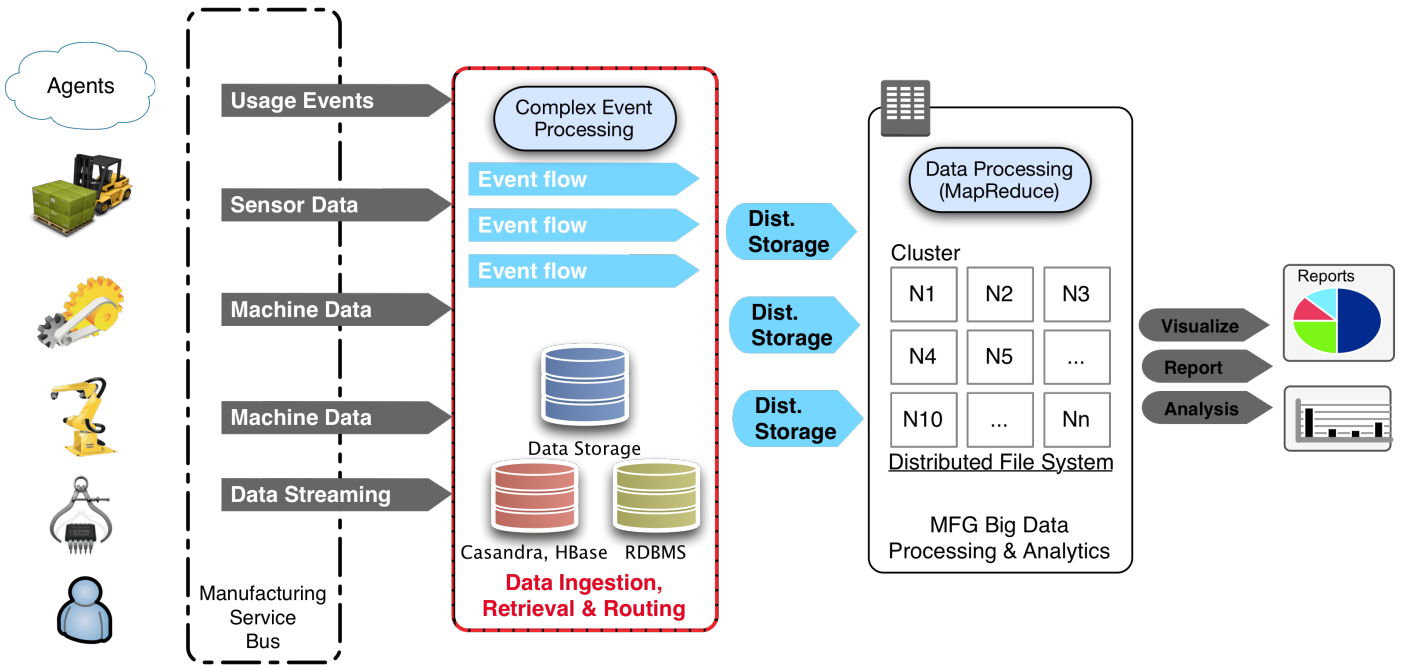


Figure 4: CloudMan Manufacturing Platform Big Data Architecture

scheduled upon the demand in MES. It is strongly the “skeleton” of the CloudMan as all other layers are functioning on top of this layer. Optimized resource scheduling is a key factor in this layer for increasing manufacturing productivity. Therefore there is a great need to develop a systematic real-time equipment health evaluation and dynamic preventive maintenance.

## VII. CLOUDMAN PLATFORM DATA ARCHITECTURE

From the time a request is submitted to the CloudMan till its fulfillment, a huge amount of data is produced by manufacturing resources such as equipments and sensors. In order to make use of this massive volume of data that is moving from bottom most layer to the top, we need to effectively collect them, analyze them, appropriately visualize them and accordingly take actions. Here, two aspects are considerable: (i) space: as we are dealing with Peta-bytes of data from different data sources, an effective way of storing them and correlate them plays an important role, and (ii) time: in some cases we are interested in analytics results over a time period (e.g. for monitoring purposes), and in some other cases a real time analytics is needed (e.g. for generating alerts). CloudMan data architecture as shown in Fig. 4 covers both aspects.

First stage is to collect data streams and store them in distributed large storages on HDFS (WSO2 Business Activity Monitor (BAM) is capable of doing this). Then the analysis engine such as BAM analyzer, processes the stored data and generates the results streams, which can be then stored in RDBMS databases, as the volume of analyzed data is much smaller than the original data, or we can store them again in our big data storage, if the volume is high. Our analysis engine is empowered by Hadoop framework, supporting powerful programming models such as MapReduce.

For real time analytics, we use a complex event processing engine such as WSO2 Complex Event Processing (CEP)<sup>8</sup> to analyze the data as they are coming and without storing them. The output stream from real time analytics generated by CEP, as well as batch analytics generated by BAM, will be sent to MVA layer to be visualized. At this layer we take benefit of tools, such as WSO2 User Engagement Server (UES)<sup>9</sup>, to build dashboards and visualize and monitor KPIs. For the outputs from CEP, we can also generate alerts for matching event detections.

## VIII. RELATED WORK

In relation to our work, there are some prominent approaches contributing architectures, platforms and models of complete frameworks for the cloud manufacturing system. Brecher et al. [4] proposed a module-based and configurable manufacturing platform based on Service-Oriented Architecture (SOA), called open Computer-Based Manufacturing (openCBM). STEP standards<sup>10</sup> are utilized to preserve the results of manufacturing processes that are fed back to the process planning stage. Li et al. [5] introduced a four-layer application service integration platform that is able to bridge multiple clouds and information systems. Interactions across organization boundaries are supported by collaboration point, which plays as an interface providing data exchange, command transferring, monitoring and so forth. The system integrated manufacturing business processes with the help of collaboration agents. This research work examined the possibility of integrating existing manufacturing applications in the cloud Computing environment. Schulte et al. [6], [7] analyzes requirements regarding process enactment for Cloud manufacturing

<sup>8</sup><http://wso2.com/products/complex-event-processor>

<sup>9</sup><http://wso2.com/landing/user-engagement-server>

<sup>10</sup>[http://en.wikipedia.org/wiki/ISO\\_10303](http://en.wikipedia.org/wiki/ISO_10303)



and provides a concept for an according software framework. They also elaborate the use of service-oriented virtual factories to establish, manage, monitor, and adapt virtual factories in a plug-and-play-like fashion.

Wang and Xu [8] proposed a Distributed Inter-operable Manufacturing Platform (DIMP) as an integrative environment among existing and future CAD/CAM/CNC applications. It is also based on SOA concept. In the platform, the requests and tasks from the users are modeled, collected and defined as "Virtual Service Combination", which echoes the manufacturing requirement at the Global Service Layer. Moreover, STEP data models are utilized as the central data schema. In a recent paper[9], they have proposed a service-oriented system called Interoperable cloud-based Manufacturing System (ICMS) which provides a cloud-based environment, integrating the existing and future manufacturing resources by packaging them using the "Virtual Function Block" mechanism and standardized description.

In contrast to existing approaches, our CloudMan platform is incorporating a cloud standard to elicit the manufacturing requirements using the TOSCA-based Bill of Manufacturing Services (BOMS).

## IX. CONCLUSION

The cloud-based design and manufacturing refers to a product realization model under the support of three core technologies of Cloud computing, Internet of Things (IoT) and Collaborative engineering. In this paper, we proposed a reconfigurable cloud manufacturing platform, called CloudMan platform, which considers these three technologies and deals with the portability and provisioning of all types of manufacturing resources as services. We incorporate TOSCA specification to model product Bill of Materials(BOM) as Bill of Manufacturing Services (BOMS) which leads to portability of such services. This elevates the manufacturing process to be more resistance against failure and defects detected in one production line since the manufacturing service can be ported to other certified production eventually. As an outlook, our future work will more focus on utilizing TOSCA-based BOMS specification in realization of virtual factories which are formed and constructed on-demand by a dynamic composition of distributed manufacturing services and resources with the CloudMan platform.

## ACKNOWLEDGMENT

The research leading to these results is sponsored by the Doctoral College of Adaptive Distributed Systems at the Vienna University of Technology as well as the Pacific Controls cloud Computing Lab (PC3L)<sup>11</sup>, a joint lab between Pacific Controls, Dubai, and the Distributed Systems Group at the Vienna University of Technology.

## REFERENCES

- [1] A. Thomas, D. Trentesaux, and P. Valckenaers, "Intelligent distributed production control," *Journal of Intelligent Manufacturing*, vol. 23, no. 6, pp. 2507–2512, 2012. [Online]. Available: <http://dx.doi.org/10.1007/s10845-011-0601-x>
- [2] T. G. P. M. Mell;, vol. The NIST Definition of Cloud Computing. NIST SP - 800-145, September 28, 2011.
- [3] L. Zhang, Y. Luo, F. Tao, B. H. Li, L. Ren, X. Zhang, H. Guo, Y. Cheng, A. Hu, and Y. Liu, "Cloud manufacturing: a new manufacturing paradigm," *Enterprise Information Systems*, vol. 8, no. 2, pp. 167–187, 2014.
- [4] C. Brecher, W. Lohse, and M. Vittr, "Module-based platform for seamless interoperable cad-cam-cnc planning," in *Advanced Design and Manufacturing Based on STEP*, ser. Springer Series in Advanced Manufacturing, X. Xu and A. Y. C. Nee, Eds. Springer London, 2009, pp. 439–462. [Online]. Available: [http://dx.doi.org/10.1007/978-1-84882-739-4\\_20](http://dx.doi.org/10.1007/978-1-84882-739-4_20)
- [5] Q. Li, C. Wang, J. Wu, J. Li, and Z.-Y. Wang, "Towards the business-information technology alignment in cloud computing environment: An approach based on collaboration points and agents," *Int. J. Comput. Integr. Manuf.*, vol. 24, no. 11, pp. 1038–1057, Nov. 2011.
- [6] S. Schulte, D. Schuller, R. Steinmetz, and S. Abels, "Plug-and-Play Virtual Factories," *IEEE Internet Computing Magazine*, vol. 16, no. 5, pp. 78–82, 2012. [Online]. Available: <http://doi.ieeecomputersociety.org/10.1109/MIC.2012.114>
- [7] S. Schulte, P. Hoenisch, C. Hochreiner, S. Dustdar, and M. Klusch, "Towards Process Support for Cloud Manufacturing," in *18th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2014) Ulm, Germany*. IEEE Computer Society, Washington, DC, USA, 2014, pp. NN–NN.
- [8] X. Wang and X. Xu, "Dimp: An interoperable solution for software integration and product data exchange," *Enterp. Inf. Syst.*, vol. 6, no. 3, pp. 291–314, Aug. 2012. [Online]. Available: <http://dx.doi.org/10.1080/17517575.2011.587544>
- [9] —, "Icims: A cloud-based manufacturing system," in *Cloud Manufacturing*, ser. Springer Series in Advanced Manufacturing, W. Li and J. Mehnen, Eds. Springer London, 2013, pp. 1–22. [Online]. Available: [http://dx.doi.org/10.1007/978-1-4471-4935-4\\_1](http://dx.doi.org/10.1007/978-1-4471-4935-4_1)

<sup>11</sup><http://pc3l.infosys.tuwien.ac.at/>