Enhancement of Transparency and Adaptability by Online Tracking of Enterprise Processes

M. Grauer, S. Karadgi and D. Metz
Information Systems Institute,
University of Siegen, Germany.
{grauer, karadgi, metz}@fb5.uni-siegen.de

ABSTRACT

Enterprises are seeking novel approaches to reduce cost in complying with regulations and requirements from original equipment manufacturers. Consequently, enterprises are investing in manufacturing execution system (MES) solutions for realizing these requirements. However, most of the MES solutions do not support processing of real-time process data acquired from shop floor for online monitoring and control of enterprise processes. Further, monitoring of enterprise processes can be classified into online tracking and passive tracing. In the contribution, a framework is envisaged for online tracking of enterprise processes based on MES concepts. This framework has been validated in an industrial scenario.

Keywords

Enterprise entity, manufacturing execution system, monitoring, offline tracing, online tracking, traceability.

1. INTRODUCTION

Enterprises need to manufacture complex products with high quality, and reduced lead times to sustain competitive advantages. In addition, enterprises (i.e. tier suppliers) are seeking innovative techniques to reduce cost in adhering to various regulations, satisfying stringent requirements from overall equipment manufacturers (OEMs), and minimizing component recall cost, among others. Overall, these conflicting requirements lay emphasis on enterprises to achieve higher level of transparency, flexibility and adaptability in enterprise processes (i.e. business and manufacturing processes) [1]. This necessitates enterprises to enhance their monitoring and control of their enterprise processes within and across different enterprise levels.

According to VDI 5600 [2], an enterprise can be classified into different manufacturing execution system (MES) levels as illustrated in Figure 1: (i) enterprise control level, (ii) manufacturing control level, and (iii) manufacturing level. Similarly, IEC 62264 [3] or ISA-95 categorizes an enterprise into several enterprise levels which can be mapped onto corresponding MES levels as depicted in Figure 1. In the article, various terminologies are adapted from VDI 5600.

Business processes are predominantly located at the enterprise control level. These processes are concerned with achieving enterprise's long term strategies essential to sustain competitive advantages. Business applications (e.g., enterprise resource planning (ERP) system) are transaction-based and generate planned performance values (i.e. TO-BE values) periodically in weeks or months [1]. On contrary, manufacturing processes are employed to accomplish the objectives set at the enterprise control level. Automation devices and their corresponding programming logic controllers (PLCs) are event-based [1] and available at manufacturing level (i.e. shop floor) to execute manufacturing processes. Enormous amount of process data (e.g., sensor readings, resource status, product positions) is generated by these systems during execution of processes in real-time (i.e. seconds or milliseconds). In addition, operators provide necessary data related to automation devices, orders and products like pre-defined reasons for a resource breakdown, order details during start of an order execution and scanning of barcodes. Overall, these values (i.e. AS-IS values) indicate the actual performance at the manufacturing level.

Attempts are being made to integrate enterprise's MES levels along vertical and horizontal direction based on enterprise reference architecture ISO 15704 [4]. Enterprise integration (EI) enhances monitoring and control of enterprise processes, and thereby, elevating transparency, flexibility and adaptability of enterprise processes. ISO 15704 provides different abstract views of an enterprise, specifies modeling approaches and defines life cycle phases of enterprise activities necessary to realize an

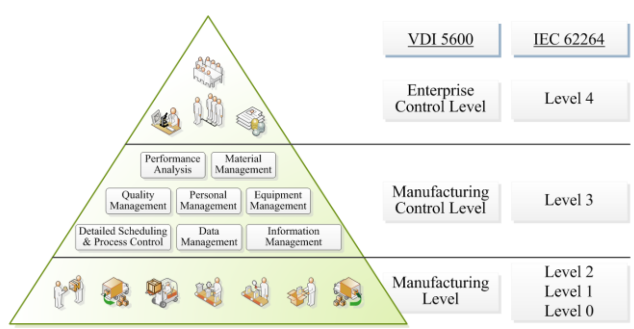

Figure 1. Enterprise levels as defined in VDI 5600 and corresponding IEC 62264 levels.

10th International Conference on Wirtschaftsinformatik,
16th - 18th February 2011, Zurich, Switzerland

integrated enterprise [5], [6]. However, realization of these architectures in terms of technologies still needs lot of attention [7]. Conceptually, service-oriented architecture (SOA) paradigm has become a de facto standard for enterprise application integration (EAI) [8]. EAI can be employed to realize *horizontal integration* of various business applications (e.g., ERP system, supply chain management (SCM) system, customer relationship management (CRM) system) at the enterprise control level. Also, an enterprise service bus (ESB) can be employed as a backbone of an SOA to realize an integrated enterprise [9].

Several software vendors have developed MES solutions to bridge the *vertical integration* gap between various MES levels (e.g., MES HYDRA [10], production monitoring and control (PMC) Provis.Agent [11], Plex Online [12]). However with these MES solutions, major issues still exist with respect to the interface between enterprise control level and manufacturing level [7], [13]. The realization of an enterprise-wide monitoring and multi-loop control system within and across all MES levels is not adequately reached [1], [13]. Different time granularities associated with enterprise levels in the vertical direction of an enterprise result in a temporal gap or asynchronization of enterprise planning and manufacturing execution [14]. Consequently, the exchange of data between different MES levels is done manually or at most semi-automatically due to inflexible and proprietary interfaces [13], [15]. In addition, processing of data in real-time for various purposes (e.g., tracking) is still a major issue with MES [7]. Overall, current MES solutions do not provide adequate integration across different MES levels [16].

The task of monitoring includes online tracking and offline tracing, among others. The focus of the research in this article is on online tracking of enterprise processes and their entities (e.g., products, orders, resources [17]) and thereby, enhancing the major activities of a manufacturing enterprise (e.g., production). The contribution is structured as follows. Section 2 highlights issues related to online tracking of enterprise entities. Section 3 presents state-of-the-art related to tracking and tracing in manufacturing and supply chains. A framework is introduced in Section 4 to enhance online monitoring and control of enterprise processes. Further, research results related to online tracking, an extension to the aforementioned framework, are elaborated. Section 5 describes an industrial case study to validate the elaborated framework. Enterprise in consideration is a casting enterprise (i.e. batch manufacturing). Hence, examples or scenarios required to clarify the terms associated with the framework will be based on casting processes. Nevertheless, the framework can be used for different types of manufacturing processes. Finally, conclusions and future works are discussed in Section 6.

2. PROBLEM DESCRIPTION

In the age of globalization, manufacturing enterprises are facing the problem of cost escalation, among others. Enterprises need to simultaneously adhere to various regulations and standards. For instance, General Food Law of European Union - European Commission (EC) No. 178:2002 places requirement for tracing immediate supplier of products as well as immediate recipients [18]. Similarly, OEMs lay down tough requirements on Tier suppliers for tracing of products. This might be for auditing purpose (e.g., quality audit), calculating actual product cost, minimizing recall cost by identifying only the defect products and corresponding customers, and so forth. In short, tracking and tracing systems are indispensable for enhancing enterprise transparency, quality and efficiency, and at the same time improve inventory management [17].

Enterprises have numerous procedures or processes to collect data from various MES levels concerning different enterprise entities, either in real-time or offline. Data can be collected from PLCs of automation devices (see [19], [20]), barcode scanners [21], and radio frequency identification (RFID) tags [21], among others. The acquired data can be initially stored in process database for a certain period of time, and later in data warehouses. In most of the cases, the stored data is used for offline (or passive) tracing of enterprise entities, revealing embedded knowledge utilizing knowledge discovery in databases (KDD) [22], and calculating key performance indicators (KPIs) [23] and overall equipment effectiveness (OEE) [24]. In regards to offline tracing, online analytical processing (OLAP) can be utilized to perform multi-dimensional queries on the stored data for detection of relevant situations like compliance violations and deviations from planned performance values [25]. For example, analysis of material lot, resource parameters, and so forth can be employed in case of recall of products. In summary, offline tracing and the corresponding control approaches tend to be reactive.

Research in tracing as well as various available tracing systems in supply chains and manufacturing focus on passive tracing of products (e.g., finished products, work-in-progress (WIP)) along manufacturing level and hence, address only horizontal dimension of an enterprise [26]. However, enterprise members from enterprise control level and manufacturing control level, and customers are interested in online tracking of enterprise entities along horizontal and vertical dimension [26]. Hence, it necessitates integrating transactional data (i.e. TO-BE values) from enterprise control level along with the real-time process data (i.e. AS-IS values) from manufacturing level. For example, plant manager would like to track the performance of the shop floor in near real-time with the information from manufacturing level and associated financial information from enterprise control level. Overall, enterprise members based on their roles and responsibilities require aggregated online tracking information for decision making processes, and thereby, transforming the control approach from reactive to adaptive.

3. STATE-OF-THE-ART

Elaborate research has been carried out in the area of tracking and tracing along horizontal dimension in supply chains and manufacturing. In spite of this, tracking and tracing are often interchanged in most instances. Nevertheless, both terms have different meaning. In this section, definitions of tracking and tracing are provided to distinguish between them, performance of online tracking from computer science perspective is described and state-of-the-art of tracking and tracing is presented in supply chains and manufacturing.

3.1 Definitions, Properties, and Performance

Various definitions are available for tracking, and tracing and traceability. Tracking is an “act of observing, in most cases, the spatial movement of an entity” [27]. It can also be considered as “gathering and management of information related to the current location of products or delivery items” [28]. IEC 62265-3 regards tracking as an “activity of recording attributes of resources and products through all steps of instantiation, use, changes and disposition” [3]. Contrary, tracing refers to “storing and retaining the manufacturing and distribution history of products and components” [26], [28]. Similar to tracking, IEC 62264-3 defines tracing as an “activity that provides an organized record of resources and product use from any point using tracking information” [3]. In addition to tracing, traceability is defined as an “ability to preserve the identity of the product and its origins or more vividly as a possibility to trace the history and the usage of a product and to locate it by using documented identification” [29]. Researchers distinguish between tracing and traceability [30], [31]. Tracing refers to pursuing a particular enterprise entity (e.g., product) through a supply chain or shop floor. However, forward traceability identifies where a particular enterprise entity has been used (i.e. material implosion), while backward traceability identifies all enterprise entities (e.g., products, raw material) consumed by a particular enterprise entity in consideration (i.e. material explosion).

Traceability can be performed on various enterprise entity types - product, batch, quality, material lot, order, production plan, resource, and operation [17]. These different entities are linked through different relationships to develop a reference traceability model [17], [30], [31]. A traceability resolution can be defined at two levels - unit/item level and lot/batch level [17], [30]. A reference traceability model was presented at batch level resolution [17], [31]. This reference model was further optimized to have traceability resolution as unit/item [30]. Aforesaid enterprise entities can be categorized into resident entities and transient entities [32], [33]. Resident entity is active in the system (e.g., simulation, tracking) over a longer duration of time and described with fewer attributes. Similarly, transient entity is created, updated and destroyed frequently. Transient entity has detailed descriptions compared to a resident entity description.

In discrete event simulation (DES), researchers describe job-driven and resource-driven models [32], [33]. These models are having analogy to the aforementioned tracking entity types. In the job-driven model, jobs are part of an active system (i.e. transitive system). An individual job contains a separate record in the active system and contains a corresponding memory footprint [33]. This job record is created at the start of the manufacturing process and thereafter, record is updated while moving through different manufacturing process steps. Contrary, resources as in resource-driven model are part of the active system processing passive jobs. Therefore, a resource record contains fewer details compared to a job record. Each of the aforementioned models has its own advantages and disadvantages. In case of resource-driven model, execution of the system is fast and uses small memory footprint, which both do not change over a period of time [33], resulting in a system operating at maximum performance [30]. Job-driven model assists to track jobs with higher clarity but at the expenses of execution speed and exhaustive memory footprint. This model can be employed in manufacturing enterprise with low volume and high-mix production [32].

3.2 Track and Trace in Manufacturing and Supply Chain

Based on reference architecture ISO 15704, standards are available stressing on significance of tracking, tracing and traceability. Manufacturing enterprise solutions association (MESA) identifies product tracking and genealogy as one of the core functions of MES providing visibility to where work (i.e. products) is at all times and its disposition [34], [35]. VDI 5600 defines traceability process with the following sub-processes: documentation of production flows, acquisition of process data for complaint verification, acquisition of product data, analysis of product and process data for different objectives, and archiving of product and process data [2]. Finally, IEC 62264-3 identifies production tracking, maintenance tracing, quality resource traceability analysis, and inventory tracing.

Research has been performed for online monitoring of enterprise processes. For instance, PMC Provis.Agent integrates various IT-systems and machine control devices, and establishes the use of information between various systems [11]. Similarly, System Insights provides an open source framework for online monitoring and analysis of manufacturing enterprises [20]. The framework constitutes following components - data delivery, data collection, and data analysis. In aforesaid systems, real-time process data from the manufacturing level is visualized online using charts and gauges. Real-time process data is stored in database for offline analysis (e.g., forward and backward traceability, compliance violations) which is the case of most of the MES solutions.

Association for automatic identification and mobility (AIM) identifies diverse family of technologies that share the functionality of identifying, tracking, recording, storing and communicating transactional, process, or product data [21]. These technologies can include barcodes, RFID tags, and data matrix codes, among others. These technologies either as single or in combination can be chosen based on requirements and feasibility. For more detailed description of tracking and tracing scenarios using automatic identification refer to [29], [30], [31].

RFID technology has been widely accepted in many industries for identification of products. However, it is mostly used to passively read the identification of the associated product and further, manually control the concerned process. To overcome this issue, decentralized product-based production control utilizing RFID was proposed [36]. Products and resources have installed RFID transponder. These products and resources exchange information between them as well with the planning level for continuous monitoring. European Union funded project TraSer provides open-source platform for tracking and tracing on an item level across different enterprises in a supply chain [27].

Available AIM technologies cannot be employed in all industrial scenarios. In case of casting enterprise, RFID tags cannot be attached to molding box as molding box will be destroyed to segregate castings and sand. Similarly due to high thermal conditions of the castings, RFID tags cannot be attached. Nonetheless, AIM technologies can be utilized in casting enterprise for online tracking of sand cores, and molten material, and so forth at batch level. To overcome the aforesaid drawbacks, Philips dotcode can be encoded on casting components – directly on components or through molding box [37]. Nevertheless, this encoding has many drawbacks. First, special resource has to be introduced into existing casting process for encoding and identification. Second, dotcode will be available for tracking of components almost at the end of the casting process. However, customers can utilize this coding for subsequent tracking and tracing. Third, component shape and size severely restrict the

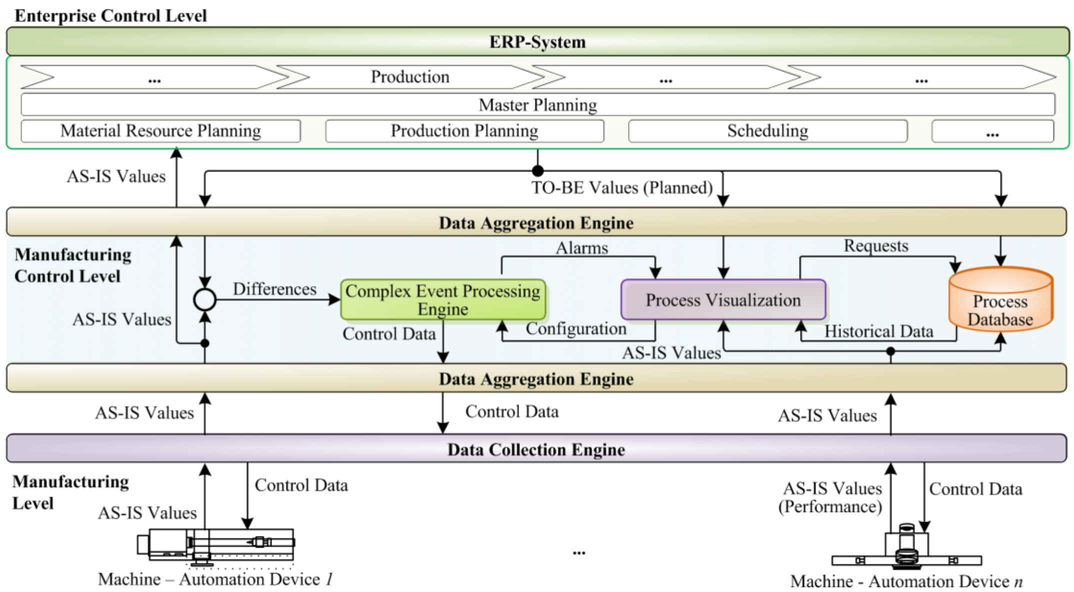

Figure 2. Framework overview for EI, and online monitoring and control of enterprise processes (adapted from [38], [39]).

encoding process. Finally, (critical) components cannot be encoded due to safety regulations, and component specifications, among others. In short, suitable AIM technologies need to be identified for the enterprise in consideration.

4. FRAMEWORK FOR ONLINE TRACKING OF ENTERPRISE PROCESSES

An overview of the envisaged framework for enabling EI, and enhancing online monitoring and control of enterprise processes, adapted from [38], [39], is depicted in Figure 2. This framework encompasses following components: (i) data collection engine for integrating physical resources located at manufacturing level (see Section 4.1), (ii) data aggregation engine for relating transactional and real-time process data from different MES levels (see Section 4.2), (iii) data aggregation engine facilitates online tracking of enterprise processes and their entities (see Section 4.3), (iv) online control of enterprise processes using complex event processing (CEP) engine and subsequently, dispatching control objects to achieve strategic objectives of an enterprise (see Section 4.4), and (v) process visualization clients provide interfaces for displaying real-time process data, online tracking information, and support forward and backward traceability of enterprise processes (see Section 4.5). These components can be seen as indispensible for establishment of closed loop control of enterprise processes and realization of knowledge feedbacks within and across various MES levels [1], [11].

4.1 Data Collection Engine for Physical Resource Integration

Data collection engine has been designed for physical resource integration at manufacturing level [19]. Existing standard and proprietary protocols (e.g., OLE for process control (OPC) servers, Modbus) for data collection are embedded into data collection engine. Due to modular design, data collection engine can be extended to include new protocols. The engine provides windows communication foundation (WCF) interface for subscribing to acquired process data from automation devices and their PLCs. Process data can include various resource parameters (e.g., temperature), unique identification tags, events, and so forth [3]. Identification tags are crucial for online tracking and control of enterprise entities. Identification tags can be collected in numerous ways: AIM techniques (e.g., by scanning barcode representing molten material batch number; reading RFID tag attached to a container with sand cores), resource generated (e.g., unique molding box number is generated once the physical lower molding box is produced by molding machine), and operator input (e.g., operator key-in order number before start of an order execution through a terminal).

Process data can be retrieved from automation devices and their PLCs using three distinctive *communication patterns*. First, the data collection engine utilizes polling mechanism with a pre-defined polling interval for requesting the data from automation devices (e.g., request temperature value of melting pot every 500 ms). Second, publish-subscribe mechanism can be used to automatically publish the data on its update, as it is preferred for online control activities (e.g., publish molten material batch number whenever molten material is poured into a molding box). Third, isochronous communication pattern can be employed. In this case, publish-subscribe mechanism is used only for subscribing to the value of a certain primary key from the automation device. On change of this value, corresponding block of data can be requested from the automation device using request-reply mechanism. For instance, block of data (e.g., pressure, weight) is requested whenever molding box number is changed indicating a new molding box has been produced. Isochronous communication pattern is not as rigid as polling mechanism, but not as lenient as publish-subscribe mechanism.

4.2 Data Aggregation Engine for EI

The data aggregation engine subscribes to data collection engine and is in charge of preliminary processing of delivered process

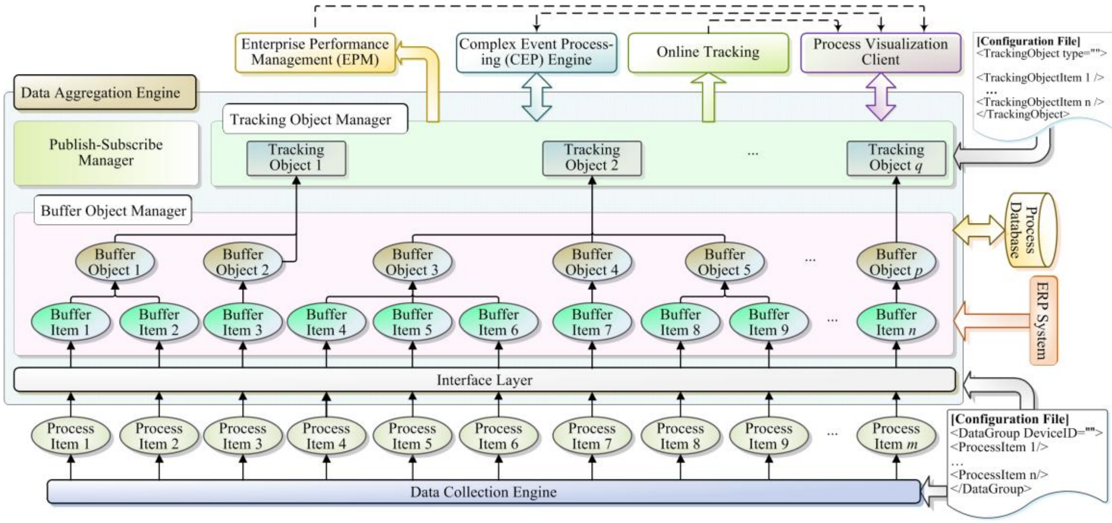

Figure 3. Simplified view of data aggregation engine.

data (i.e. AS-IS values) as depicted in Figure 3. Further, it integrates the AS-IS values with the corresponding transactional data (i.e. TO-BE values) from business applications. Integrated data is committed to a process database for offline analysis (e.g., KDD process, forward and backward traceability). In addition, selected subset of the integrated data is consumed to derive tracking objects.

On startup, data collection engine and data aggregation engine are initialized from an XML-based configuration file (see Figure 3). An editor is available to system administrators to manage the configuration file. This configuration file defines necessary information required by the aforesaid engines during run-time. The information contained can be related to resources, data groups, buffer objects, process items, buffer items, database settings, and so forth. A resource is identified by a unique device identification tag which can have one or more data groups denoting possible operations that can be performed by the resource. For example, molding machine will contain data groups for operations to manufacture lower molding box and upper molding box. A data group consists of process item definitions (e.g., temperature) and communication pattern settings. In turn, each process item is characterized by item name, memory location in the automation device and data type. Data groups along with the process items are initialized in the data collection engine and process items are delivered to the subscribed data aggregation engine through WCF interface.

Buffer objects are managed by buffer object manager in the data aggregation engine. A buffer object corresponds to a certain data group in an automation device instantiated through data collection engine and signifies a certain activity in the enterprise processes. Similar to data groups, buffer object consist of number of buffer items. A buffer item is akin to a process item but contains additional information like readable name, column name for mapping onto a process database table, unit. The relationship between process items, buffer items and buffer objects is depicted in Figure 4. The configuration file explicitly defines the relation between buffer items and corresponding buffer object, when (i.e. trigger condition) and where (i.e. database table name) a buffer object has to be stored in the process database.

Data collection engine publishes process items from different automation devices which have been updated. A process item is mapped onto a buffer item by interface layer (see Figure 3). On creation of a buffer item, buffer object manager creates or updates corresponding buffer object, as shown in Figure 5. A buffer object is created if it does not exist; otherwise, it is updated with the arrival of new buffer item's value. Hence, a certain buffer object (e.g., a lower molding box) is allocated memory only once. As depicted, buffer items arrive at different time and frequency, and update the corresponding buffer object. In addition, buffer object manager retrieves transactional data (i.e. TO-BE values) from ERP system corresponding to a buffer object. Based on a trigger condition which has been predefined in the configuration file, a buffer object is processed by buffer object manager for committing it in process database. Simultaneously, the buffer object is forwarded to tracking object manager for further processing. Once the buffer object is committed to the process database, it is reset and waits for arrival of new buffer items.

Trigger conditions define when buffer objects have to be stored in the process database and forwarded to tracking object manager. These trigger conditions depend upon automation device vendors

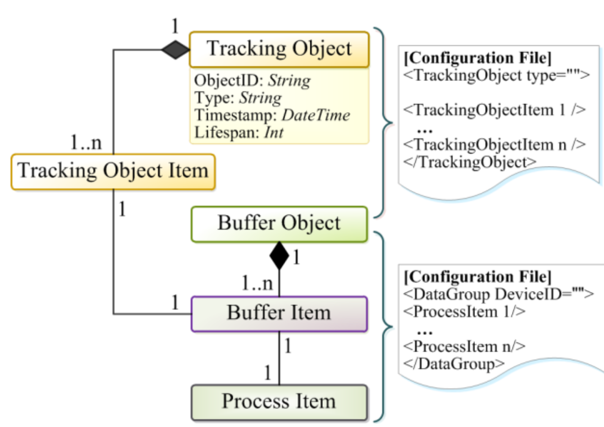

Figure 4. Relationship between process item, buffer item, buffer object, tracking object item and tracking object.

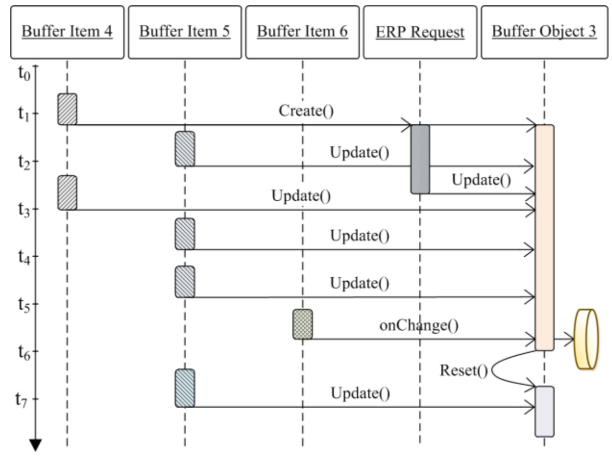

Figure 5. Sequence diagram for depicting interaction between buffer items and buffer object.

who define conditions when the process data will be stable for further processing. There exist different trigger conditions: *onChange()*, *always()*, and *onCondition()*. These trigger conditions can be extended to include new conditions. *onChange()* trigger condition is valid when a certain predefined buffer item's value is changed. For example, details of lower molding box are committed to process database when the sequence number of the lower molding box is increased. *always()* trigger condition can be utilized when buffer object has to be continuously committed to process database on arrival of a buffer item. To analysis melting pot, it is necessary to plot a graph with temperature against time, necessitating storing temperature value every time it arrives. *onCondition()* trigger condition is an extension to *onChange()* trigger condition with an additional condition that a certain predefined buffer item has changed to a preset value. Molten material before pouring into molding box is treated with gas to obtain certain chemical composition of molten material. Buffer items (e.g., gas flow rate) arrive continuously but the buffer object is committed when a particular status buffer item is set to one, indicating the metal treatment process is in progress.

Finally, publish-subscribe manager of data aggregation engine manages subscription of various components of the framework (e.g., CEP engine, process visualization clients). Similar to data collection engine, WCF interface is available for subscribing. The aforementioned tasks (e.g., update of buffer object, storing of buffer object) are handled in dedicated threads. Hence, it necessitates implementing data aggregation engine functionalities in a thread-safe manner (e.g., mutual exclusion).

4.3 Tracking Objects and Online Tracking of Enterprise Entities

On trigger conditions, buffer objects are committed to a process database. Simultaneously, these objects are forwarded to tracking object manager as illustrated in Figure 3. Tracking object manager is responsible for supervising tracking objects and their life cycle. Tracking objects are aggregated for deriving online tracking information of enterprise processes and their entities, and utilized in numerous ways: (i) tracking objects are forwarded to CEP engine for online control of enterprise processes, (ii) tracking objects are exploited for enterprise performance management like calculating online KPIs, and (iii) all subscribed process visualization clients receive tracking objects for visualization.

Tracking is described as "activity of recording attributes of resources and products through all steps of instantiation, use, changes and disposition" [3]. Like traceability, tracking can be performed on numerous enterprise entity types - product, batch, quality, material lot, order, production plan, resource, and operation [17]. Further, these entities can be classified into transient and resident entity types [32], [33]. In case of enterprise with low volume and high mix manufacturing, resource can be considered as resident entities, while other enterprise entities can be considered as transient entities. In short, it is necessary to employ both transient and resident entity types for achieving enhanced enterprise transparency.

A traceability reference model at the batch level was proposed [17], [31]. This reference model describes various possible relationships with other enterprise entities. In the envisaged framework, online tracking is performed at batch level as most of the raw materials (e.g., sand, molten material) as well as (semi-) finished products are viewed from batch perspective. In addition, batch level tracking resolution will result in reduction of memory footprint, and faster system execution and responsiveness. Hence, the aforementioned reference model was adapted and extended to include resource entity, and batch - raw material and products are segregated as illustrated in Figure 6. Item-centric traceability reference model proposed in [30] can be adapted and extended as mentioned previously in case to obtain item level tracking resolution.

Data members of a tracking object can be categorized into two sets. First set represents data required by tracking object manager to manage tracking object. This data include unique identification tag, and timestamps, among others. Each tracking object is identified by a unique identification tag. For example, molding box tracking object is identified by a unique number and incremented whenever a lower molding box is produced. Similarly, identification tag can be generated for various enterprise entities. Second set represents data required to derive the status of an enterprise entity (i.e. tracking object items). These two sets together are depicted in Figure 4.

A tracking object contains condensed status information of an

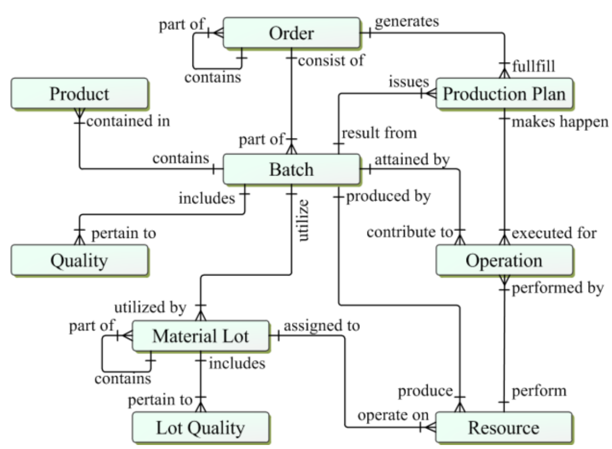

Figure 6. Reference tracking model (modeled as enterprise relationship model with crow's foot notation).

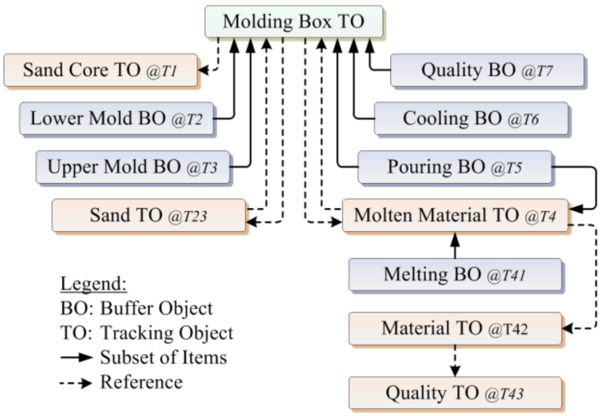

Figure 7. Data members of molding box tracking object and its relationship to other tracking objects.

enterprise entity necessary to derive different perspectives. A tracking object is composed of numerous tracking object items and references to other tracking objects. A buffer object denotes a certain operations within an enterprise process (e.g., pouring operation) consisting of buffer items. However, only subset of these buffer items are mapped onto tracking object items as illustrated in Figure 4. This subset of items represents critical control related parameters/attributes. Data mining methods and structured interviews with the domain experts can be employed to identify these control related parameters [40]. Similar to configuration file described in previous sub-section, an XML configuration file contains explicit relation between buffer items, buffer objects, tracking object items and tracking objects. In addition, a tracking object contains references to other tracking objects, accessed through unique identification tags. An example of a molding box tracking object is depicted in Figure 7 which includes buffer objects and references to other tracking objects. Here, molding box tracking object contains buffer objects denoting operations (e.g., production of lower molding box, upper molding box) performed on the physical molding box and references to other tracking objects (e.g., batch number of the inserted sand cores).

A tracking object will undergo different phases of life cycle - creation, modification and destruction. Creation and modification can be explicitly defined in the XML configuration file. A tracking object is created by allocating memory. This object is modified when the corresponding buffer objects are created or updated. In addition, tracking objects are linked with other tracking objects on creation or update of a certain buffer object. For instance on pouring, reference of molten material tracking object is appended to corresponding molding box tracking object (see Figure 7). Tracking objects are available in main memory. Hence, it is critical to define the termination condition, especially for transient entities like products and orders. Destruction terminates a tracking object by removing the reference and freeing memory for new tracking objects. Consequently, performance is enhanced in terms of memory footprint and execution speed. Termination condition can be specified in numerous ways. The output of a (short-term) production planning is production plan containing quantity, bill-of-materials (BOM), material routing, and resources, among others [41]. In this case, completion of the last step of the material routing can be considered as the termination condition. Workflow associated with an order and managed in workflow management system (WMS) can also be considered. In addition, maximum expected lifespan of tracking objects can be defined in the configuration file. Aforementioned termination conditions can be employed for defining the termination condition. A special garbage collector manages the destruction of expired tracking objects.

Creation and modification timestamps of a tracking object as well as tracking object items are recorded. Timestamps are crucial for processing by other components of the framework, especially by CEP engine for control of enterprise processes. In few instances, two timestamps are recorded for an event [42]. First timestamp is noted when the actual event is triggered. Second timestamp refers to when the activity is processed by the system. This is required for critical applications. Here, only the timestamp when an event is triggered is considered.

Tracking objects contain transactional data, real-time process data (i.e. number of tracking object items) and references to various tracking objects. These objects can be utilized to create online reports from different perspectives defined by enterprise members' roles (e.g., supervisor, plant manager) and their corresponding privileges (e.g., defined in a lightweight directory access protocol (LDAP) server). Research has identified different ways to navigate between objects (i.e. tracking objects), operation (i.e. buffer objects) and attributes (i.e. buffer items) [29], [30]. For instance, object (e.g., sand batch) → object (e.g., molding box) tracking navigation can be utilized to identify all the molding box tracking objects where a particular sand batch tracking object is used. Aforesaid tracking object navigation can be utilized to derive higher level reports.

Online tracking assists enterprise members from enterprise control level and manufacturing control level with a tool to track enterprise processes and its entities in near real-time. Online tracking object contains manufacturing process data as well as financial information from business applications (e.g., ERP system). With the up-to-date information, enterprise members can proactively perform corrective actions. In addition, tracking object can be forwarded to CEP engine for online control of enterprise processes.

4.4 Online Control of Enterprise Processes

An event is "an object that is a record of an activity in a system" [43]. It has three aspects: (i) form - represents data components, (ii) significance - denotes an activity/operation, and (iii) relativity - describes relation with other events. Definition of an event and its aspects has been included in a tracking object. Tracking objects and tracking object items represent event form as well as event significance. Similarly, relations defined between tracking objects based on the reference model (see Figure 6) characterize event relativity. Therefore, a tracking object can be interpreted as a complex event which is composed of several simple events. For effective control of enterprise processes, temporal and causal relations between complex events should be considered [43]. On creation or modification, tracking objects are forwarded to a CEP engine (e.g., EsperTech [44], Drools Fusion [45]) which supports causal and temporal relations. CEP engine is also capable of dispatching control data to concerned MES levels to achieve strategic objectives of an enterprise.

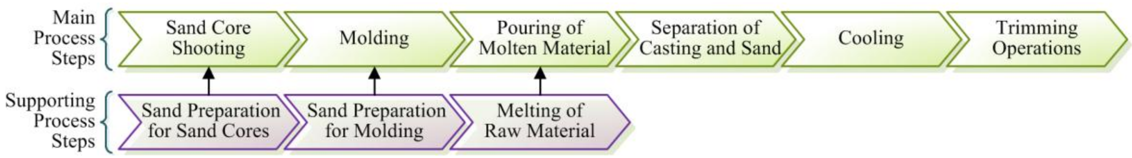

Figure 8. Casting process and its supporting process steps.

4.5 Process Visualization Client

Process visualization client has to fulfill following tasks: (i) visualize real-time process data and transactional data using charts and gauges, (ii) display alarms derived from the CEP engine, (iii) provides interfaces to configure the behavior of the CEP engine, (iv) display detailed online tracking report, and (v) supports forward and backward traceability. For these tasks, the visualization client takes enterprise member's roles and privileges into account.

5. A CASE STUDY

The framework for enabling EI and online control of enterprise processes elaborated in Section 4 can be put into practice in different types of manufacturing, especially in batch manufacturing (e.g., casting processes) and discrete manufacturing (e.g., sheet metal forming processes). Here, an attempt is made to realize the framework for casting processes. The enterprise in consideration is characterized by low volume and high-mix production. Casting process is considered as a flow job shop model of scheduling where each job has to pass through a fixed sequence of resources, defined as a workflow in a WMS (see Figure 8). The casting process is supported by special purpose machines with a high production rate (e.g., molding machine can produce approximately 250 molds per hour). To efficiently utilize capital intensive resources, online monitoring and control of enterprise processes is mandatory.

Job-driven and resource-driven or transient entity and resident entity models are considered for online tracking. Enterprise entities in consideration are component's batch, molten material batch, sand batch, and orders and resources (see Figure 9). For

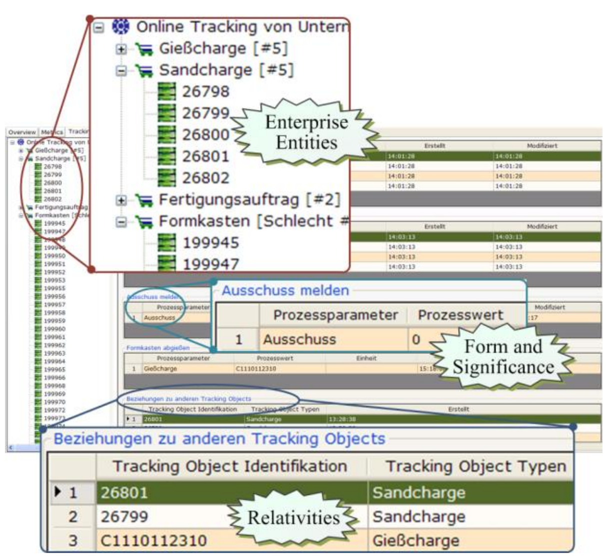

Figure 9. Screenshot of online tracking information.

instance, molding machine is considered under resource-driven model or as a resident entity, while other enterprise entities are considered under job-driven or as a transient entity. Online tracking is performed at batch level tracking resolution due to following reasons: (i) most of the raw materials (e.g., sand, molten material) are viewed from batch perspective, (ii) multiple components are simultaneously manufactured (e.g., a molding box contains one or more similar components), (iii) orders are executed in small batches as with low volume and high-mix, and (iv) enhance performance of the implemented application by reducing memory footprint and increasing execution speed.

Each tracking object contains details of the activities performed (i.e. event significance) and the values of the parameters employed to realize the activity (i.e. event form) as illustrated in Figure 9. In addition, tracking object contains references to other tracking objects (i.e. event relativity) which are necessary to generate online tracking report (see Figure 9). Process visualization client provides an interface to visualize the online tracking information. This information is updated during various phases of its life cycle. Further, it takes enterprise members' roles and privileges into account. The workflow, shown in Figure 8, defines the termination condition for tracking objects. For instance, molding box tracking object is removed from memory when it is made available for trimming operations (e.g., grinding operation). The framework has been implemented using Microsoft™ Visual Studio IDE and .NET framework 3.5.

6. CONCLUSIONS AND FUTURE WORK

Enterprises endeavor to overcome various challenges induced by globalization by enhancing their monitoring and control of enterprise processes or entities. Enterprise entities include products, orders, resources, and material lot, among others. Monitoring of enterprise entities can be classified into tracking, and tracing and traceability. Tracing or traceability are offline analysis tasks performed using historical data. Consequently, associated control of enterprise entities is reactive. However, real-time process data and transactional data can be employed for tracking of enterprise entities resulting in an adaptive control system. A framework has been presented to establish online monitoring and control of enterprise processes. Data aggregation engine pre-processes the process data from various resources. The pre-processed data is further aggregated to derive tracking objects. Online tracking information is generated from tracking objects using various navigation ways associated with tracking objects. Apart from visualizing real-time data, process visualization client provides an interface to display the online tracking report.

At the moment, the framework has been used in an enterprise for online monitoring and control of batch manufacturing (i.e. casting enterprise). Currently, research work is been carried out to apply the framework for discrete manufacturing processes especially for an automotive sheet metal component supplier.

7. ACKNOWLEDGMENTS

Parts of the work presented here have been supported by German Federal Ministry of Economics and Technology (BMWi) as part of "Central Innovation Programme SME" (ZIM) initiative (KF2111502LL0). Also, we are thankful to our industrial partner Ohm & Häner Metallwerk GmbH & Co. KG, Germany for the opportunity to implement the elaborated framework in a casting enterprise. Especially, we would like to acknowledge Dr.-Ing. Ludger Ohm and Dr.-Ing. Georg Dieckhues for their valuable comments and support.

8. REFERENCES

[1] Kjaer, A. P. 2003. The integration of business and production processes. *IEEE CONTR SYST MAG.* 23, 6 (Dec. 2003), 50-58. DOI= http://dx.doi.org/10.1109/MCS.2003.1251181.

[2] VDI 5600. 2007. Manufacturing execution system (MES) - VDI 5600 Part 1. Verein Deutscher Ingenieure (VDI).

[3] IEC 62264. Enterprise-Control System Integration. All Parts.

[4] ISO 15704. 2005. Requirements for Enterprise Reference Architecture and Methodologies. ISO 15704:2000/Amd 1:2005.

[5] Neaga, E. I., and Harding, J. A. 2005. An enterprise modeling and integration framework based on knowledge discovery and data mining. *INT J PROD RES.* 43, 6 (Mar. 2005), 1089-1108. DOI= http://dx.doi.org/10.1080/00207540412331322939.

[6] Giachetti, R. E. 2004. A framework to review the information integration at the enterprise. *INT J PROD RES.* 42, 6 (2004), 1147-1166. DOI= http://dx.doi.org/10.1080/00207540310001622430.

[7] Saenz de Ugarte, B., Artiba, A., and Pellerin, R. 2009. Manufacturing execution system – a literature review. *PROD PLAN CONTROL*. 20, 6 (Sept. 2009), 525-539. DOI= http://dx.doi.org/10.1080/09537280902938613.

[8] Linthicum, D.S. 2003. *Next Generation Application Integration: From Simple Information to Web Services*. Addison-Wesley Professional, Amsterdam.

[9] Chappell, D. 2005. *Enterprise Service Bus*. O'Reilly, CA.

[10] MES HYDRA. http://www.mpdv.de/en/default.htm.

[11] Sauer, O., and Sutschet, G. 2006. Agent-based control. *COMPUT CONTROL ENG J*. 17, 3 (June. 2006), 32-37. DOI= http://dx.doi.org/10.1049/cce:20060305.

[12] Plex Online - Manufacturing Execution System. 2010. http://www.plex.com.

[13] Panetto, H., and Molina, A. 2008. Enterprise integration and interoperability in manufacturing systems: trends and issues. *COMPUT IND*. 59, 7 (Sept. 2008), 641-646. DOI= http://dx.doi.org/10.1016/j.compind.2007.12.010.

[14] Gronau, N., and Lindemann, M. 2008. Ableitung von IT-Strategien für die Produktion (in German). In *Strategische Bedeutung der Produktion*, D. Specht, Ed. Gabler, Wiesbaden. 147-162. DOI= http://dx.doi.org/10.1007/978-3-8350-5486-8_9.

[15] Karnouskos, S., Baecker, O., de Souza, L., and Spiess, P. 2007. Integration of SOA-ready networked embedded devices in enterprise systems via a cross-layered web service infrastructure. In *Proceedings of the 12th IEEE Int. Conf. on Emerging Technology and Factory Automation ETFA 2007* (Patras, Greece, Sept. 25 - 28, 2007). 293-300. DOI= http://dx.doi.org/10.1109/EFTA.2007.4416781.

[16] Sauer, O. 2009. Trends in manufacturing execution systems. In *Proceedings of the 6th Int. Conf. on Digital Enterprise Tech. DET 2009* (Hong Kong, Dec. 14 - 16, 2009), Springer, 685-693. DOI= http://dx.doi.org/10.1007/978-3-642-10430-5_53.

[17] Khabbazi, M. R., Yusof Ismail, M. D., Ismail, N., and Mousavi, S. A. 2010. Modeling of traceability information system for material flow control data. *AUST J BASIC SCI*. 4, 2 (Feb. 2010), 208-216.Will, M., and Guenther, D. 2007. Food Quality and Safety Standards. http://www2.gtz.de/dokumente/bib/07-0800.pdf.

[18] Will, M., and Guenther, D. 2007. Food Quality and Safety Standards. http://www2.gtz.de/dokumente/bib/07-0800.pdf.

[19] Grauer, M., Metz, D., Karadgi S., Schäfer, W., and Reichwald, J. W. 2009. Towards an IT-framework for digital enterprise integration. In *Proceedings of the 6th Int. Conf. on Digital Enterprise Tech. DET 2009* (Hong Kong, Dec. 14 - 16, 2009), Springer, 1467-1482. DOI= http://dx.doi.org/10.1007/978-3-642-10430-5_111.

[20] Vijayaraghavan, A. 2009. MTConnect for realtime monitoring and analysis of manufacturing enterprises, http://www.systeminsights.com/.

[21] AIM – Association for Automatic Identification and Mobility. http://www.aimglobal.org/.

[22] Choudhary, A., Harding J., and Tiwari, M. 2008. Data mining in manufacturing: a review based on the kind of knowledge. *J INTELL MANUF*. 20, 5 (Oct. 2008), 501-521. DOI= http://dx.doi.org/10.1007/s10845-008-0145-x.

[23] VDMA 66412. Manufacturing Execution Systems (MES) - Kennzahlen (in German). All Parts.

[24] Muchiri, P., and Pintelon, L. 2008. Performance measurement using overall equipment effectiveness (OEE): literature review and practical application discussion. *INT J PROD RES*. 46, 3 (July. 2008), 3517-3535. DOI= http://dx.doi.org/10.1080/00207540601142645.

[25] Chaudhuri, S., and Dayal, U. 1997. An overview of data warehousing and OLAP technology. *SIGMOD RECORD*. 26, 1 (Mar. 1997), 65-74. DOI= http://doi.acm.org/10.1145/248603.248616.

[26] Van Drop, K-J. 2002. Tracking and tracing: a structure for development and contemporary practices. *LOGIST INFOR MAN*. 15, 1, 24-33. DOI= http://dx.doi.org/10.1108/09576050210412648.

[27] Kemény, Zs., Ilie-Zudor, E., Szathmári, M., and Monostori, L. 2008, TraSer: An open-source solution platform for cross-company transparency in tracking and tracing. In *Proceedings of the 17th World Congress, The International Federation of Automatic Control* (Seoul, Korea, July 6 - 11, 2008), 4493-4498.

[28] Kärkkäinen, M., Ala-Risku, T., and Främling K. 2004. Efficient tracking for short-term multi-company networks. *INT J PHYS DISTRIB LOGIST MAN*. 34, 7, 545-564. DOI= http://dx.doi.org/10.1108/09600030410552249.

[29] Töyrylä, I. 1999. *Realising the Potential of Traceability - A Case Study Research on Usage and Impacts of Product Traceability*, Doctoral Thesis. Finnish Academy of Technology, Espoo.

[30] Rönkkö, M. 2006. *A Model for Item Centric Material Control in Manufacturing*. Master Thesis. Helsinki University of Technology.

[31] van Dorp, C. A. 2004. *Reference-Data Modeling for Tracking and Tracing*, Doctoral Thesis. Wageningen Agricultural University, the Netherlands.

[32] Schruben, L. W., and Roeder, T. M. 2003. Fast simulations of large-scale highly congested systems. *SIMULATION*. 79, 3 (Mar. 2003), 115-125. DOI= http://dx.doi.org/10.1177/0037549703255634.

[33] Fowler, J. W., and Rose, O. 2004. Grand challenges in modeling and simulation of complex manufacturing systems. *SIMULATION*. 80, 9 (Sept. 2004), 469-476. DOI= http://dx.doi.org/10.1177/0037549704044324.

[34] MESA. 2004. MESA's Next Generation Collaborative MES Model. White Paper. No. 8. MESA International.

[35] MESA. 1997. MES Explained: A High Level Vision. White Paper. No. 6. MESA International.

[36] Zaeh, M. F., Ostgathe, M., Lau, C., and Wiesbeck, M. 2009. Adaptive, product-based control of production processes. In *Proceedings of the 3rd Int. Conf. on Changeable, Agile, Reconfigurable and Virtual Production CARV 2009* (Munich, Germany, Oct. 5 - 7, 2009), 99-110.

[37] Sluiter, P. 2007. Spezifische Kennzeichnung von Gussstücken in Großserien (in German). *Giesserei*. 94 (Jul. 2007), 77.

[38] Grauer, M., Karadgi, S. Metz, D., and Schäfer, W. 2010. Online monitoring and control of enterprise processes in manufacturing based on an event-driven architecture. In *Proceedings of the 4th International Workshop on Event-Driven Business Process Management* (Hoboken, NJ, Sept. 13, 2010).

[39] Grauer, M., Karadgi, S., Metz, D., and Schäfer, W. 2010. An approach for real-time control of enterprise processes in manufacturing using a rule-based system. In *Proceeding of Multikonferenz Wirtschaftsinformatik MKWI 2010* (Göttingen, Feb. 23 - 25, 2010), 1511-1522.

[40] Grauer, M., Karadgi, S., Müller, U., Metz, D., and Schäfer, W. 2010. Proactive control of manufacturing processes using historical data. In *Proceedings of the 14th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems KES 2010* (Cardiff, UK, Sept. 8 - 10, 2010), 399-408.

[41] Stadtler, H. 2008. Production planning and scheduling. In *Supply Chain Management and Advanced Planning*. Stadtler, H. and Kilger, C. (Ed.). 4th edition, Springer, Berlin, 199-216.

[42] Seeger, B. 2010. Kontinuierliche Kontrolle - Complex Event Processing: Auswertung von Datenströmen (in German). In *iX - Magazin für professionelle Informationstechnik*, 2/2010, 131.

[43] Luckham, D. 2007. *The Power of Events - An Introduction to Complex Event Processing in Distributed System*. Addison-Wesley, Munich.

[44] EsperTech, http://www.espertech.com/.

[45] Drools Fusion, http://www.jboss.org/drools/.