Research School for Operations Management and Logistics

Business Process Management Technology for Discrete Manufacturing

Irene Vanderfeesten
Paul Grefen

Beta Working Paper series 486

BETA publicatie	WP 486 (working paper)
ISBN	
ISSN	
NUR	804
Eindhoven	September 2015

Business Process Management Technology for Discrete Manufacturing

Irene Vanderfeesten and Paul Grefen

School of Industrial Engineering, Eindhoven University of Technology, the Netherlands.
{i.t.p.vanderfeesten, p.w.p.j.grefen}@tue.nl

Abstract. Contemporary business process management technology has successfully been implemented to streamline and coordinate administrative business processes. The use of this type of technology to automate processes in complex manufacturing is however less mature. In this paper we discuss the research challenges and requirements for the application and extension of business process management concepts and theories to discrete manufacturing processes as encountered in e.g. the automotive industry. In these processes humans and robots collaborate to produce complex physical products. Awareness of physical constraints during execution, such as safety issues, is very important in these processes. Integration of the process management with real-world events, such as safety alarms and failure of machinery, has to take place. We present an architectural solution showing the extensions needed to current business process management technology for adoption in the high tech manufacturing domain and we validate this solution by construction. From this we conclude that the application of business process management to discrete manufacturing is possible but requires substantial additions to contemporary systems. This paper contributes by providing the framework for these additions.

Keywords: industrial application of CIS, discrete manufacturing, robotic actors, real-world events, situation-aware, collaborative processes, business process management technology, reference architecture.

1 Introduction

Over the past decades, business process management (BPM) technology has become very popular to automate business processes [1, 2]. It has proven to have great potential in the efficient management of administrative business processes by streamlining and coordinating the tasks that have to be executed by various actors, which are typically humans or administrative information systems. Since the products in these administrative processes are mainly informational and easy to digitize and copy, there are almost no physical constraints to the management of the process. The uptake of this type of technology in domains with complex physical products and various types of physical actors is low however. Several efforts have been made to implement the BPM concepts and technology in healthcare, e.g. [3], where the 'products' are patients, the actors are mainly human, and many physical resources (e.g. MRI-scanner, hospital beds) are used or consumed. Only a limited number of efforts are reported yet on the adoption of BPM in the high tech manufacturing domain. In fact, the known initiatives in this domain, such as the Crosswork [4] and ADVENTURE [5] projects, focus on high level process choreographies between partners in a virtual enterprise, but none of them aims for the management of the detailed, physical production process within one factory. This paper intends to make a conceptual and architectural step in this direction.

Complex, high-tech, discrete manufacturing processes have specific characteristics that make the straightforward adoption of BPM technology from the administrative domain difficult. First of all, in these collaborative processes we find *mixed actor types*. Actors executing the manufacturing process tasks can be humans, robots, autonomous guided vehicles, or hybrid (machines directly controlled by humans). This collaboration puts constraints on the management of the process. For instance, when a human actor comes too close to a robot safety measures have to be taken and the robot has to be paused. Or in case of machine failures, the execution of a task has to be passed to another actor, possibly of another type. Secondly, the process needs to be *integrated with real world technical events* in order to create a *situation-aware process*. For instance, in the example of explicit safety management, sensors are used to detect dangerous situations. These sensors notify the process execution engine about an alarm and the engine in turn decides which measures (e.g. sending a warning to the human who comes too close, pausing a single robot, or stopping the whole production line) should be taken. Also, the process engine has to be able to communicate the worklist with a robotic actor through technical events and execution scripts. Finally, as a consequence of mass customization, we also find individualized complex products in these manufacturing processes that require a more flexible way of modeling the process and its actors. Certain product configurations may for instance be assembled fully automatically while others require human interventions. Therefore, discrete manufacturing processes have to be managed in a different way than administrative processes.

In this paper, we propose the use of concepts and technologies from the BPM domain to approach the above, and we discuss which extensions to these concepts and technologies are needed for the adoption in discrete manufacturing processes with mixed actors and individualized products. The concrete context of this research is the start of a European project in the Horizon 2020 – Factories of the Future program, called HORSE, in which research and development partners, and high tech manufacturing companies work together on the development of a framework for the efficient management of ‘smart factories’.

The application of BPM concepts and technology to this high tech manufacturing domain brings a number of challenges. In section 2 we will motivate our case with an example. Next, we list the identified challenges. In section 4 we design an architectural solution for this problem and validate it by showing that all requirements are covered by this design (correctness by construction). We conclude this paper in Section 5 with an outlook into further research.

Fig. 1. A car manufacturing production line with robots [10]

Fig. 2. Collaborative actor in car manufacturing: human working hand-in-hand with robot [11]

2 Motivating example

As a motivating example of the industrial application of BPM technology to discrete manufacturing we discuss a factory of the future for the assembly process of cars. In our factory different car models with their various configurations and specific user demands are assembled at the same production line. Parts of the assembly process are fully automatically executed by robots (see e.g. **Fig. 1**) while other parts are done by humans. In contemporary factories, mostly humans and robots are strictly separated by fences, in order to prevent dangerous situations or even fatal accidents where human workers get killed by a robot, e.g. [14]. Technological developments in these productions lines require collaborations between robots and humans in which e.g. the human is in charge (see e.g. **Fig. 2**). This requires a better and more detailed safety management where sensors control the safety situation on the work floor and signal (possibly) dangerous situations, and an overall process management module decides on the actions to be taken in case of any safety issues.

Other developments in manufacturing focus on the management of the overall production process in terms of efficiency and efficacy. Different type of cars with different individual configurations are produced on the same production line (instead of separate production lines or even separate factories for each car type), meaning that the car subassembly itself moves along the production line and actors (robots, collaborative robots and humans) are called in to execute tasks on the subassembly. This means that e.g. car A needs a simple plastic dashboard that may be assembled by a robot, while the next car on the line, car B, needs a luxurious wooden dashboard that has to be assembled by a human to prevent it from being damaged. In these more diverse and flexible production lines robots also become capable of executing several different tasks. Therefore, the allocation of tasks (including the passing of execution scripts and work instructions) to the right actor then becomes crucial.

As illustrated above, the management of the car production process of the future, on the level of the factory floor, brings a number of challenges that can be supported by BPM technology. This motivating example sets the stage for our general idea to apply BPM technology in discrete manufacturing as further outlined in the next section.

3 General idea

Discrete manufacturing processes are business processes in which individual products are produced in discrete series production. Products have parameters that can be set on the individual product level. A typical example is car manufacturing, discussed above, where each car can be parameterized to suit a specific customer (to support demand chain operation). In complex, high-tech discrete manufacturing, we find mixed actor types: actors can be human, robotic (e.g. autonomous robots, autonomous guided vehicles or other machinery) or hybrid (machines directly controlled by humans, such as exoskeletons). Depending on the parameterization of individual products, the assignment of actor types to manufacturing steps can vary. Also, the management of exceptions in the manufacturing process can lead to dynamic re-planning and dynamic re-assignment of actor types. For example, where a robot shows to be incapable to deal with a specific parameterization of an individual product (e.g. because of physical constraints), or in case of an (ad hoc) machine failure, a human may be re-assigned. Also, when a safety issue occurs (e.g. a human comes too close to a robot), the robot may be paused and the task(s) to be executed may be re-assigned to another actor or re-planned for execution at a later point in time.

In order to support these complex discrete manufacturing processes with BPM technology, these processes are to be modelled as business process types, the manufacturing of individual parameterized products as business process instances. A business process type models the manufacturing process of a series of products, a business process instance models the manufacturing cycle of an individual product. The use of process variants accommodates for product parameterization. The business process type describes the tasks that have to be executed to produce a product, the order in which these tasks have to be performed and the actor types or abstract roles that are qualified to execute the task. The task requirements are extended with actor capabilities, such as the weight an actor is able to lift or move, or the specific skills that are needed for the execution of a task.

Human, hybrid and robotic actors are included in the role model of the actor as abstract actor classes, which can be further specialized in different classes of factory workers, machines, and robots. Specific attributes of the actors, such as their skills and capabilities, are also modeled. When a process instance is executed according to the business process type each task has to be performed by a specific actor. The assignment of actors to tasks is called dynamic role resolution.

The run time selection of the right actor for the right task is done based on the specific capabilities and specific attributes of the human, robotic, and hybrid actors, and on the allocation constraints as described in the business process type. These attributes and constraints may concern availability, reliability, cost, time, experience, precision, safety, training of human actors, etc.

The process engine dynamically assigns and re-assigns the most suitable actor to a task and monitors task execution progress, but does not support the execution of the task by the actor itself. For the actual execution of the task the actor knows how to execute the task himself, or scripts may be available that may be sent with the task assignment to the actor. Therefore, support has to be developed for interactions between the actor and the process execution environment within a task (e.g. starting, pausing, resuming, and stopping an actor; and task passing between human and robotic actors) and the task can therefore be seen as a ‘half open box’ [6] with regard to its execution management.

While the process is executed and controlled by the process execution environment, unexpected events may occur. These unexpected events can for example be safety dangers, failures, and ad hoc unavailability of actors. They are, for instance, indicated by signals and warnings and involve real-time event processing. The process management system to be developed processes these signals and handles the exceptions in a structured way, e.g. by performing the rollback of specific activities or the activation of alternative process paths based on e.g. explicit safety parameters in the tasks and processes. To properly handle these exceptions also dynamic process change may be required, i.e., the on-the-fly changes of the process specification of running business process instances. Structured exception handling should therefore be supported both at design time (at the time of specifying the business process type) and at run time (when the process instances are executed). This involves elements of dynamic change [5, 7], i.e., the on-the-fly change of the process specification of running business process instances.

To summarize this general description, the system to be developed should address a number of specific characteristics of the high tech manufacturing domain, such as:

1. A mixed actor role model, including detailed capabilities of the actors
2. A physically constrained process model, allowing for detailed task requirements, constraints, and process variants
3. A mixed actor role resolution mechanism
4. A physical process execution mechanism, incl. real-time process control and advanced exception handling

4 Research Challenges

In order to support the specific characteristics of the discrete manufacturing domain, we propose a number of extension to contemporary BPM concepts and technologies, both in design time and runtime environments. Below we list these research challenges as requirements to the system to be developed.

4.1 Design time

At design time the processes, actors, roles, tasks and other constraints are defined. Current BPM concepts and models should be extended to incorporate the additional information needed to manage a high tech manufacturing process in the way as described above. The additional requirements are as follows.

Requirement 1 –Robotic and hybrid actors.
The role model should be extended as to include human, robotic and hybrid actors as abstract roles, which can be further specialized in different classes of factory workers, machines, and robots. Also, the attributes (such as the capabilities, availability, reliability, costs, time, and safety) of each actor should be modeled.

Requirement 2 – Task requirements and allocation constraints.
Following the more detailed information on the actors, the constraints and rules on allocation of actors to tasks should also be part of the process definition. It should for instance be modeled which capabilities an actor should have to execute the task, which safety requirements should be respected, which are the time and cost constraints on the execution of the task, etc.

Requirement 3 – Process variants.
The process definition should allow the specification of process variants that accommodate for product parameterization. At design time several deviations from the blueprint production process for typical product configurations may already be envisioned and specified. For instance, a car with or without an airbag would follow a different path through the production line that may be specified at design time already.

4.2 Runtime

After the process has been designed, a process execution environment is used for the automated, discrete control of mixed-actor manufacturing processes at runtime. Contemporary BPM technology has to be extended to support the specific characteristics of mixed-actor manufacturing processes. The additional requirements for this runtime environment are listed below.

Requirement 4 – Assignment of tasks to robotic and hybrid actors.
Apart from the allocation of tasks to humans the system should also be able to allocate tasks to robotic actors. This may require a different way of communicating tasks, e.g. by pushing tasks to a robot's worklist and including execution scripts in the task assignment.

Requirement 5 – Structured safety exception handling
The execution environment should support the structured handling of exceptions, such as starting another process, rollback of activities, dynamically reassigning actors, or dynamically changing the process in case of e.g. safety warnings. Taking into account the task requirements, actor capabilities, actor availability, allocation constraints, etc. as given in the extended process definition.

Requirement 6 – Structured unavailability exception handling
When actors suddenly become unavailable, e.g. due to failures of machinery, or when parts of the process are held or suspended because of safety problems, the system has to determine what kind of measurements have to be taken in order to manage the running process instances. This may involve the re-assignment of the tasks to other actors (human, robotic and hybrid) or the dynamic change of the discourse of tasks for each process instance.

Requirement 7 – Monitoring of task executions.
The execution environment should be capable of monitoring and managing task executions by actors (e.g. start, pause, resume, abort). In case of a safety problem the system should e.g. be able to force the pausing, restart, or abortion of a task execution for any kind of actor. Also an actor should be able to indicate a change in the status of the task he is executing.

In the next section we show how these requirements can be projected on standard BPM concepts. We extend two well-known BPM system reference architectures to design a solution and validate this solution by construction.

5 Reference architecture for mixed actor manufacturing BPM

In order to design a conceptual solution that incorporates the requirements derived above, we present an extension of two well-known and widely used BPM system reference architectures: the WfMC model [8] and the Mercurius architecture [9].

5.1 WfMC reference architecture

The WfMC reference model [8] is a high level reference architecture and mainly describes the runtime aspect of a BPM system. It puts the workflow enactment service (also termed as process engine) in a central position and focuses on the different kinds of interfaces with other modules, e.g. the interface with workflow clients and the interface with other enactment services.

The proposed extensions to support mixed actor manufacturing processes are in the process definition tools and the interfaces to applications. The process definition tools have to be extended to support the modeling of:

- (i) robotic and hybrid actors with their attributes (cf. requirement 1)
- (ii) extended task requirements and allocation constraints taking into account the specific actor attributes (cf. requirement 2)
- (iii) process variants for individual products (requirement 3)

The interface to the client applications can be used for the communication with human and/or hybrid actors in which the human is in the lead of the execution of tasks (cf. part of requirement 4). The interface still contains a worklist, but may be extended with task execution management (e.g. the actor may be directed to stop his work because of a safety issue), specific work instructions for the execution of a task for particular individual product, and a safety module in which the actor may indicate a safety problem (e.g. alarm button). With these additional functionalities the input and output of information for the correct management of safety issues (cf. requirement 5), unavailability of actors (cf. requirement 6), and task execution management (cf. requirement 7) is guaranteed.

In addition to the extension on the client applications, a new interface needs to be added to the reference model in order to support the interaction with robotic actors. For these actors also a worklist, safety module (cf. requirement 5) and task execution management module (cf. requirement 7) have to be added. Furthermore, execution scripts may be forwarded to the robotic actor via this interface (cf. requirement 4).

Fig. 3. Indicates the proposed extensions to the WfMC reference architecture.

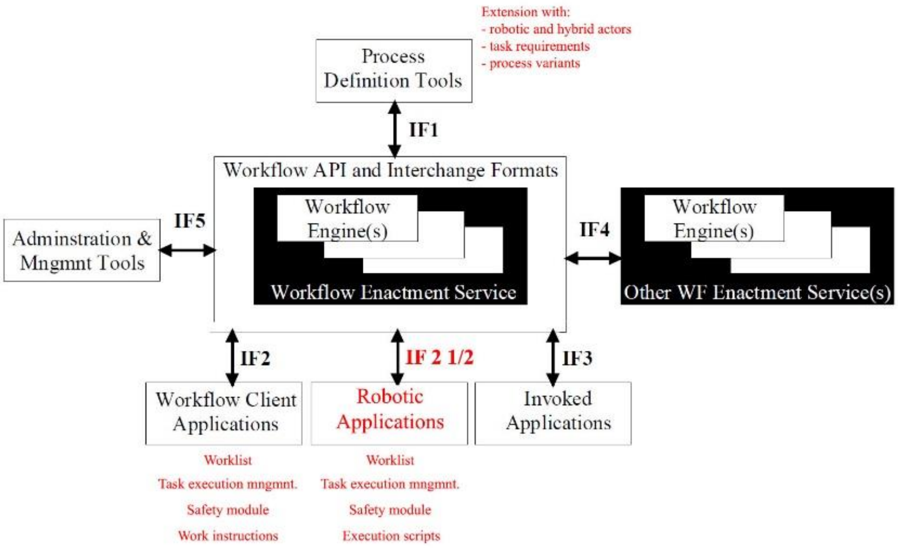

Fig. 3. The WfMC reference model including the proposed extension.

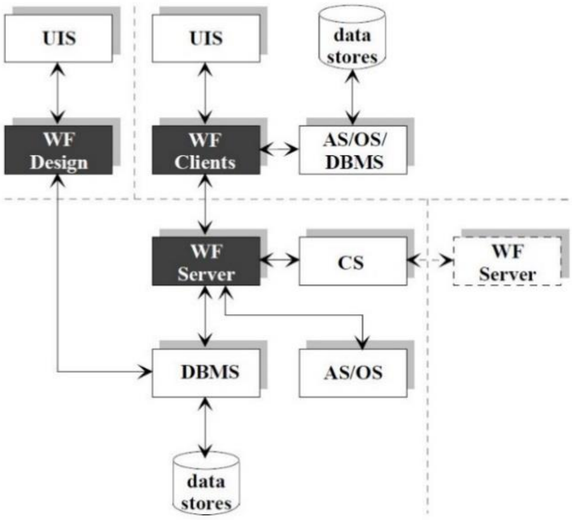

Fig. 4. The Mercurius global reference architecture

5.2 Mercurius reference architecture

A second reference architecture for BPM technology is the Mercurius reference architecture [9]. It is also designed to provide blueprints for the design of workflow management systems. Mercurius describes three levels of decomposition and as such is a more detailed architecture than the WfMC reference model. Below we discuss the extensions that have to be made to the Mercurius architecture in order to support discrete manufacturing processes with mixed actors on the second level of abstraction. **Fig. 4** depicts the global level of the Mercurius architecture. It defines three modules: the Design Module, the Server Module, and the Client Module, and is comparable to the WfMC model in abstraction level. Below we described the extensions to be made to the three more detailed levels of the architecture in order to support the requirements given in Section 2.

Extension of the Design Module architecture

The architecture of the design module defines three sub modules, two interfaces and a number of data sets. As is shown in **Fig. 5**, the extensions to be made to this architecture in order to support the additional functionality needed for modeling complex manufacturing processes may be included as extension modules. In the "*hybrid actor design*" module the further specification of the organizational data, i.e. actor classes, actors and their capabilities, and other attributes, is covered. In the "*task requirement design*" module, the additional tasks requirements on actor capabilities, safety, etc. and other allocation constraints are modeled. And in the "*process variant design*" module the design time changes to the standardized production process are specified for each individualized product.

By introducing these three extension modules requirement 1, 2 and 3 are covered respectively.

Extension of the Enactment server architecture

The enactment server architecture, shown in **Fig. 6**, also has extension modules. Here we propose to add the required functionality through adding the following extension modules.

The *safety management module* is contacted by the process engine in case a safety warning is received from one of the actors or sensors in the process. The safety management module processes this warning and decides what measures have to be taken to solve the safety issue. It may decide to pause the whole production process or parts of it and resume when the issue is resolved. It may decide that the task has to be re-assigned to another actor that is then selected by the task re-allocation module. The communication between these different modules is done via the enterprise service bus and coordinated by the process engine.

The *task (re)allocation module* calculates the most appropriate actor for a certain task, given the constraints specified in the process model and resource model. It is contacted by the process engine when a task has to be assigned or re-assigned to an actor. It may contain complex decision algorithms that also take into account overall

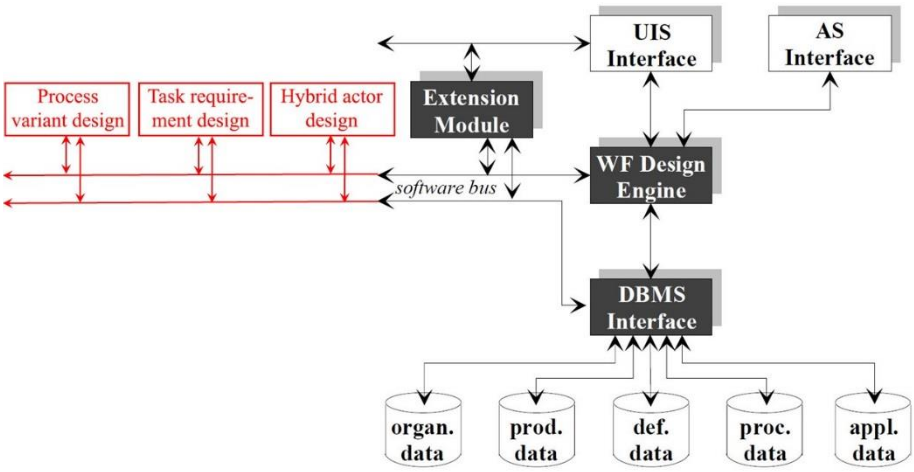

Fig. 5. The Mercurius design module architecture, including the proposed extension

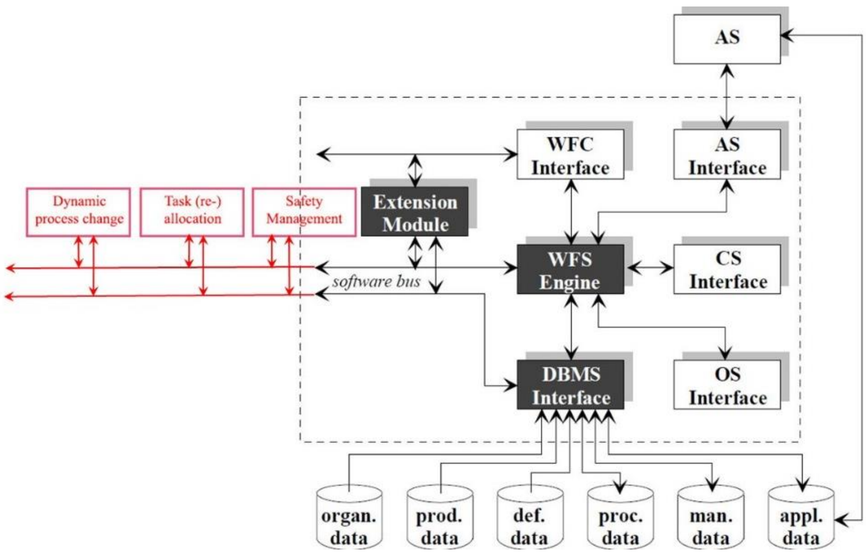

Fig. 6. Mercurius enactment server architecture including the proposed extension

process efficiency or other overall objectives for successful management of the process (such as low idle time of certain expensive actors, or reasonable workload for human actors).

The *dynamic change module* is responsible for changing the process of running process instances. These changes may be invoked by e.g. the safety management module.

The addition of these three extension modules covers the required functionality of requirement 5, 6 and 7.

Extension of the Enactment client architecture

The architecture for the enactment client is depicted in **Fig. 7**. The client interface (UIS interface) to the process engine provides functionality for the handling of tasks through e.g. worklist. We propose to extend this interface with the functionalities listed below.

For safety management, functionality is included to signal (manually or automatically) a safety issue by the actor, or to receive a safety warning and required action from the process engine.

The task execution management module is used for the interaction between the process engine and the actor on e.g. pausing, resuming, quiting, handing over, or declining a task.

Through this interface the actor may also receive specific work instructions (for human or hybrid actors) or execution scripts (for robotic actors) for the task to be executed.

The functionality added by these extensions covers requirement 4 and 7.

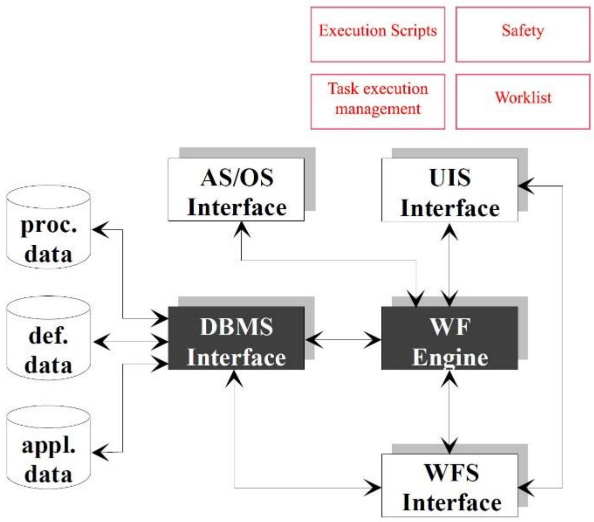

Fig. 7. Mercurius enactment client architecture including the proposed extension

6 Conclusion

In this paper we have introduced the idea to apply BPM concepts and techniques in a new domain: the high tech manufacturing domain. This domain has several specific characteristics that pose research challenges to the further development of existing BPM technology. First of all, we find mixed actor types in these manufacturing processes. Humans, robots or hybrid actors collaborate to produce the products. This collaboration poses challenges to role resolution, safety management, and dynamic (re)assignment of tasks in case of unavailabilty in the process. Secondly, integration with real-world events (e.g. from sensors) has to take place to support situation-aware processes, Finally, high tech products are often customizable leading to the specification of many process variants. Therefore, more detailed process definitions, mixed actor role resolution mechanisms and process execution mechanisms, such as real time process control, and structured physical exception handling, are to be developed. We have formulated a number of conceptual solutions as extensions to well-known reference architectures describing BPM systems and – as a validation by construction – we have argued how the derived requirements for supporting discrete manufacturing processes are satisfied by our conceptual design.

The proposed extensions to the architectural models described above will be elaborated and concretely implemented in our future work in the context of the European Horizon 2020 project called HORSE.

7 References

1. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A: Fundamentals of Business Process Management. Springer Berlin Heidelberg, Berlin, Heidelberg (2013).
2. Dumas, M., Van der Aalst, W.M.P., Ter Hofstede, A.H.: Process-Aware Information Systems: Bridging People and Software Through Process Technology. John Wiley & Sons, Inc., Hoboken, New Jersey (2005).
3. Lenz, R., Reichert, M.: IT support for healthcare processes – premises, challenges, perspectives. Data Knowl. Eng. 61, 39–58 (2007).
4. Grefen, P., Mehandjiev, N., Kouvas, G., Weichhart, G., Eshuis, R.: Dynamic business network process management in instant virtual enterprises. Comput. Ind. 60, 86–103 (2009).
5. Schulte, S., Schuller, D., Steinmetz, R., Abels, S.: Plug and play virtual factories. IEEE Internet Comp. 16, 78–82 (2012).
6. Grefen, P., Ludwig, H., Dan, A., Angelov, S.: An analysis of web services support for dynamic business process outsourcing. Inf. Softw. Technol. 48, 1115–1134 (2006).
7. Reichert, M., Dadam, P.: ADEPTflex - Supporting dynamic changes of workflows without losing control. Journal of Intelligent Information Systems 10, 93–129 (1998).
8. Workflow Management Coalition: Glossary - A workflow Management Coalition Specification.

9. Grefen, P., de Vries, R.R.: A reference architecture for workflow management systems. Data Knowl. Eng. 27, 31–57 (1998).
10. News item China Daily, retrieved from http://usa.chinadaily.com.cn/weekly/2012-10/05/content_15797110.htm.
11. News item Just Auto, retrieved from http://www.just-auto.com/news/audi-adds-human-robot-cooperation-in-manufacturing_id155721.aspx
12. News item Sky News, retrieved from http://news.sky.com/story/1512029/production-line-robot-kills-car-factory-worker