Interoperable Process Design
in Production Systems

Georg Weichhart[1,2(✉)] and Christian Stary[2]

[1] PROFACTOR GmbH, Steyr-Gleink, Austria
`Georg.Weichhart@PROFACTOR.at`
[2] Department of Communications Engineering – Business Informatics,
Johannes Kepler University, Linz, Austria
`Christian.Stary@jku.at`

Abstract. In Sensing, Smart, and Sustainable (S^3) Enterprises information is not only gathered from an internet of production things supporting smart decision making, but this information also allows execute processes automatically. These processes need to be interoperable for sustainable operation while enabling dynamic adaptation. In this paper we present a development framework successfully tested in a European project demonstrating key features of subject-orientation for production process support across the IEC 62264 control hierarchy layers. It aligns business planning and logistics with manufacturing operations and production management, utilizing a communication protocol choreography, message passing and data exchange between production-relevant behavior encapsulations. The overall system is modular and adaptive, where interoperability is supported through process models.

Keywords: Enterprise interoperability
Subject-oriented Business Process Management · Internet of Things

1 Introduction

The vision of the Sensing, Smart and Sustainable Enterprise (S^3 Enterprise) builds around a layer of abstraction (the enterprise operating system) that allows to make use of data from heterogeneous sources. Sensing technologies, like the Internet of Things (IoT) are targeting low-power, wireless sensor networks which are connected to the cloud. Collected data is stored context free, information about concrete sources is hidden. The data is analyzed and processed, for supporting decision makers. Sub-symbolic AI algorithms (deep learning) support classification of events using high volume data. To become sustainable, the enterprise system must then adapt to these smart decision. A pragmatic approach to interpretability is needed. This implies changing the way processes are executed reacting to environmental changes. However, planning and executing production processes is not executing simple chains of service calls where semantically unified data is used to trigger one service after the other. Production steps implemented by different human and artificial actors need to be orchestrated. Adaptation in the enterprise implies maintaining interoperability of processes. Advanced process support for actors is required from the global production

process point of view. Interoperability is not sufficient on the technological and data semantics layer but is also required on organizational process and pragmatics layer.

For understanding the needed process interoperability support, a simple process where a human and a robot collaborate, is used to pinpoint the direction of future work.

In Sect. 2 we present interoperability research with a focus on adaptive, modular systems and organizational interoperability. Section 3 presents an initial approach developed in the So-Pc-Pro project for interoperability of processes. This is then summarized in the conclusions at the end of this article.

2 Interoperability

The production *process* is the most important integrating aspect of a production system. Subsystems like machine and human actors need to align their activities with resource availabilities and part flows. When going into details, execution of process parts is dependent on the concrete machines executing these segments.

On a higher level of abstraction, a similar situation exists. The execution of processes along the supply chain is decentralized, where suppliers determine their own production schedule, partially executed concurrently, taking decisions independently. However, at the end of the day, a globally synchronized production plan is required which links machines across multiple supply chain participants.

Therefore, in a production system both, parallel and independent executed sub-systems with their specific behavior, as well as their output and behavior being interoperable with the rest of the system is need.

In the following, we review our view on Enterprise Integration and Interoperability in general, and discuss then process-related approaches to enterprise interoperability.

2.1 Enterprise Integration/Interoperability

Enterprise Integration [1] has its roots in Computer Integrated Manufacturing (CIM) approaches developed in the 80's and 90's. Its aim is to support the alignment of the enterprise's information sub-systems. The enterprise information system includes artificial and human agents that share and exchange information [2].

Integration is an activity, which is necessary to bring together independently created systems in order to realize a larger function that requires all involved systems. Integration activities are researched in the fields of enterprise integration (EI) [1], enterprise interoperability [3], enterprise application integration (EAI) [4]. EAI focuses on the technical aspects of distributed applications and middleware.

It is of importance to understand, that all these research fields are model-driven approaches. The exchanged information and the interfaces between systems is explicit [5]. Enterprise Modelling (EM) and Enterprise Architecture (EA) have strong connections to the problems discussed in enterprise integration and interoperability [6]. Interoperability builds on research on integration, but places emphasis on loosely coupled systems, and stresses the independence and decentralized aspects of the involved systems.

The to-be integrated, independent systems have interfaces across which it is necessary to exchange information to realize the overarching function. This implies that there is a technical interface, which is used by the other system to exchange information. Such interfaces include human machine interfaces (HMI) when one of the two systems is a human.

These pieces of information must not only have the correct technical data format, but also the semantics of the conveyed information must be clear and synchronized (also in the HMI case). The semantic interoperability barrier (aka conceptual interoperability barrier) refers to the representation of a real-world element and problems with the interpretation of its structure and meaning by different systems.

In addition to syntactical and semantical interoperability, there is an organizational aspect of information exchange. In the case of machine-to-machine communication for example the APIs (application programming interface) must be called in the correct sequence. In the case of considering a production system on supply chain level, (business) processes of the organizational systems need to be aligned. The use of information within a process provides the specific context in which information is used. This aspect incorporates pragmatics and addresses the use of data beyond its context free interpretation.

The European Interoperability Framework discusses three essential levels or dimensions of interoperability [1]. On the bottom, the syntactic level is concerned with technical interoperability between two systems. In the middle, there is the semantic level concerned with the interpretation of data. On top, there is the pragmatic level concerned with the use of information in a concrete context.

Besides the different levels of concern, there is also the degree to which multiple systems are interoperable. This quality is described on a continuum.

To be *fully integrated* implies that there is a common (concrete) model, which is implemented, in full detail, in all involved systems. This is an extreme position on the discussed spectrum, because, as a consequence, the interfaces between the systems get blurred. All systems are dependent, and any change needs to be addressed in all systems.

At the other end of the spectrum are *incompatible* systems. These systems are not able to exchange information or interact (technical aspect), required information is not understood or interpreted in the wrong way (semantic aspect), or the behavior of one system has negative impact on other systems (e.g., incompatible processes: organizational aspect).

In the continuum between *incompatible* and *fully integrated* systems are interoperable systems. The research field of Enterprise Interoperability, in contrast to Enterprise Integration, places emphasis on the loosely coupled (loosely integrated) systems and discusses unified and federated interoperability [7]. It is not implied that the integration end of the continuum provides higher quality. On the contrary, loosely coupled systems are more adaptive and are much harder to engineer.

2.2   Approaches Supporting Interoperability in Production

In a stable system interoperability is less often disturbed once reached. In a system which is dynamic and facing a lot of changes, more sophisticated support for sustainable interoperability is needed.

In the following we review selected approaches supporting interoperability, giving attention to their support for process interoperability and modular, adaptive systems. Due to space constraints, we have to focus on a limited list of approaches.

Technology Approaches

One example for a technological approach is the *Cilia Middleware* service oriented, integration environment. It is based on mediators, supporting the monitoring and dynamical adaption of mediator chains during execution [8]. While the possibility to adapt is a core aspect of the cilia middleware, this adaptation is triggered from the outside. Missing is the possibility to do a process re-planning beyond the exchange of services, if a better candidate is available and requires a changed interaction.

Semantic Approaches

A combination of middleware and ontology is implemented in the Ontology of Enterprise Interoperability extended for Complex Adaptive Systems (OoEI^{CAS}) [9]. It implements middleware aspects using an actor-based system, where the actors encapsulate the external systems to be made interoperable. The realized Domain Specific Language (DSL) implements the systemic aspects of the Ontology of Enterprise Interoperability [10]. This DSL supports the implementation of actors as a core concept for representing systems. This allows to implement interoperability on process level, as the actors may map the behavior of external systems to interoperable behavior.

The Liquid Sensing Enterprise (LSE) approach [11] also assumes that the enterprise is a complex adaptive system. This approach provides the infrastructure that the enterprise evolves over time. Fundamentally, this approach is a model driven engineering approach. Independent agents communicate with others using models. This implies that used models (e.g., for decision making or communication) change and need to be transformed. A model morphism agent is responsible for proposing model mappings. Simulation is used to estimate the impact of change on the existing system. Models may also describe behavior of systems.

Organizational Approaches

The multi-agent-system based SUddEN environment supports organizational interoperability through the use of performance indicators [12]. A core team of supply chain participants designs the performance measurement system for the planned chain. Supporting the collaborative design activity supports alignment of business goals and due to the need to make indicators explicit, organizational interoperability is indirectly required and measured in the following.

In CrossWork, an agent based systems approach is used to provide a modular infrastructure for supply chain planning [13]. In this approach, process interoperability, requires explicit process models at the external level. An ontology is provided that allows to capture goals, capabilities as well as roles of supply chain partners [14]. Here local workflows (represented by agents) contribute to global workflows.

The Mediation Information System approach (MIS) [15] is a methodological framework for supporting knowledge management and continuous interoperability management for the supply chain. It supports alignment of information systems and collaborative behavior. Adapting MIS needs to be performed manually.

S-BPM (Subject Oriented Business Process Management) can be used to support interoperability [16] and implement production processes [17]. By extending the business process features and directly interfacing machines S-BPM allows to implement a level of abstraction where subjects (roles of agents) exchange information objects, but internal behavior is hidden. Yet the semantics of the information objects has to be defined externally (i.e. full integration is needed). The interface between subjects needs to be determined *ex ante*.

3   Adaptive Processes and Interoperability

In the S^3 Enterprise vision, information is gathered from an internet of production things to enable smart decisions for the enterprise system which are turned into actions to maintain sustainability. Taking a simple scenario from collaborative robotics, in assembly processes show the need for ad-hoc & local re-planning of processes. However, local process changes have impact on the global level as well. The scenario below will serve as an initial case to highlight requirements and needs for an interoperability support infrastructure capable to re-plan processes.

We are considering a collaborative robotic system, where robots work with humans assembling small parts (following [18]). Both sub-systems, the human and the artificial, need to have appropriate tools for their tasks available. It is not possible, that the tools are passed from humans to robots (tools need to be mounted to the robot arm). Not only because of this, for rescheduling a task from a human to a robot, logistic, and preparatory steps have to be taken. Task-relocation happens due to external and also internal events. A GPS sensor is capable of reporting that the delivery of parts is delayed. That information needs to be connected to the right-order schedule, and the overall schedule needs to be redone, where different task processing times of the robot and the human need to be taken into account. Internal rescheduling events will trigger task re-location when a sensor signals physical or cognitive overload of the worker. This will happen on the spot, making ad-hoc changes may be necessary. Task-reallocation to the other direction happens when a sensor identifies a worn-out tool of the robot.

This scenario reveals interoperability issues on all three levels of interoperability, which will be used in the following to structure the discussion.

3.1   General Organizational Interoperability Requirements

On technical level, all components have to be connected and a standard for machine-to-machine communication have to be in place. Assembly is, in principle, a flexible process. It is required, that the hardware components have to be modular in order to re-arrange the overall assembly system over time.

Descriptions of tasks (i.e. models) have to be available in machine and human readable form. The semantics of the tasks depends on the application domain. It captures active resources (artificial and human agents, including their skills), passive resources (e.g. materials), and activities.

Agents are the main elements that execute a certain task. Different agents have different skills allowing to execute different tasks. Agents may be able to execute a task in different qualities. Agents may be organized in teams, projects, and organizational units.

For executing tasks, resources like raw materials, knowledge are needed, and semi-finished parts are produced. Flows of parts and materials (passive resources) between agents are one kind of flows that needed to be addressed. Not only physical flows but also activity flows need to be captured and controlled. Some activities have to be executed in parallel (synchronous, asynchronous), some in sequence. The possibility to use coordination patterns is needed for planning (see the following).

Skills are concepts that allow to do initial planning and select agents capable to certain tasks. Skills are realized by coordinated activities called processes.

All these model elements need to be understood by all agents. The semantics needs to be clear and the syntax needs to be machine and human readable as well.

Given the scenario above, re-planning/redesign takes place in case of events when processes can not be executed as planned/designed.

3.2 Process Design and Execution

In the following, we present a Business & Production Process Management approach, which allows to orchestrate different human and artificial agents on the shop floor. This partially fulfils the above presented requirements.

Subject-oriented Business Process Management (S-BPM) represents a generic approach to modelling, execution and improvement of business processes. It has been applied in production companies, expanding the scope of process management to planning, logistics, and shop floor activities [17]. The different levels of abstraction and granularity of the IEC 62264 control hierarchy have been seamlessly integrated on the process level using the S-BPM approach in the EU-funded project "Subject-Orientation for People-Centred Production"[1]. As shown in Fig. 1, the vertical integration for modularisation is based on using subject-oriented process models as a uniform representation layer. In S-BPM, processes at all levels of the IEC 62264 control hierarchy, including High Level Control (HLC) and Low Level Control (LLC), can be represented. Processes on the LLC levels (i.e. Levels 1 and 2) execute them in real time. Processes on HLC levels (i.e. Levels 3 and 4) operate not on in real time. Data exchange between processes at the different levels can be based on existing automation standards, including OPC UA (IEC 62541) and B2MML (IEC 62264). OPC UA as a communication protocol is implemented in most modern PLC environments. It allows exchanging semantic data models also via web services or binary protocols.

---

[1] www.SO-PC-Pro.eu.
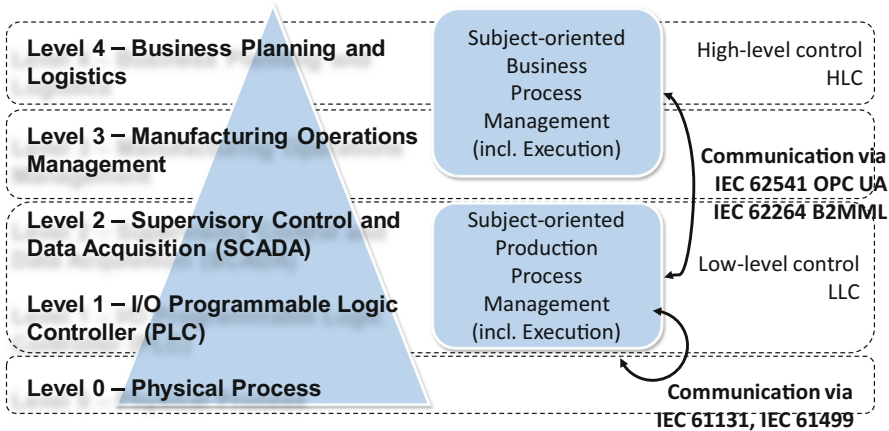
Fig. 1. Vertical integration of processes based on S-BPM and existing data standards including OPC UA extended based on (cf. [19])

Vertical integration allows modularisation of operating processes in mutual dependency with respect to planning, executing and monitoring requirements. The SO-PC-Pro project has led to an S-BPM based process integration framework. Technically SO-PC-Pro is based on recent developments using the Metasonic Suite (www.metasonic.de) with a B2MML interface, an OPC UA interface, an extension for transforming S-BPM behaviours to executable IEC 61131-3 conform PLC code. As a standard for vendor- and platform-independent communication, we have been using OPC UA (cf. IEC 62541) to interface LLC processes executed by Programmable Logic Controllers (PLCs).

The S-BPM approach provides an abstraction that supports process descriptions and designs for human and artificial actors. In the following, we sketch process communication via OPC UA to show how humans and robots are coordinated in a single process. Runtime communication between level 3 and level 2 processes has been enabled including data exchanged (according to the OPC UA standard) [17], p. 33ff.

The OPC UA standard (IEC 2008) supports specification of the interface of the address space of OPC UA servers', to define content that is visible/editable for clients. Clients can monitor and subscribe to attributes and events on the server. Figure 2 shows the schema for the interplay between the behaviour of the "Robot" subject (in the Metasonic Suite) and a PLC addressable via an OPC UA client/server. It enables the configuration of the endpoint of the server and the relevant node (e.g. variable, method, and event). It also allows reading/writing variables from/to information objects carrying data to be exchanged, invoking methods on the server, and subscribing to server events.

The OPC UA S-BPM connector enables (i) reading values from a Robot or any PLC and storing them in a business object, and (ii) writing concrete values of an information object to variables of a PLC or Robot. In this way, the concrete OPC UA server endpoint providing desired variables can be implemented. When an action and a relevant information object needs to be selected before mapping variables onto each
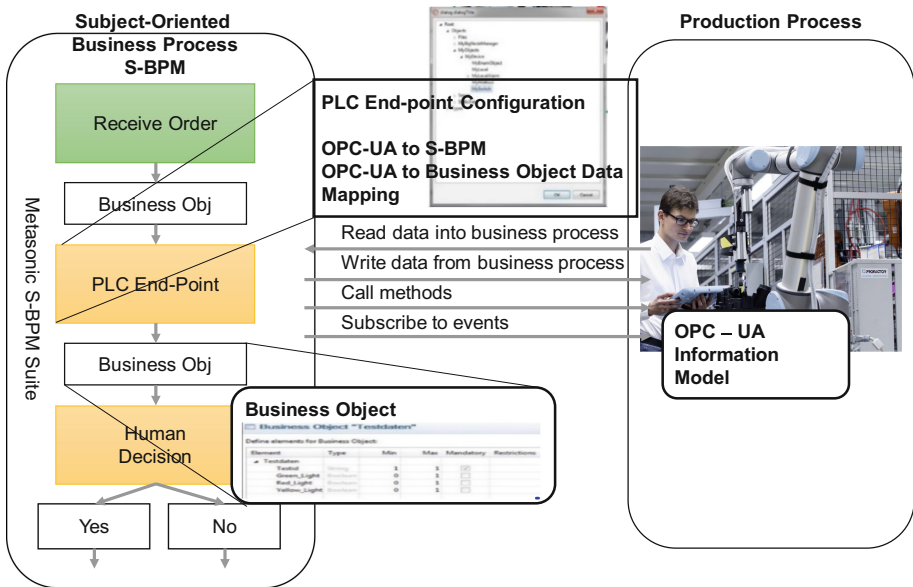
Fig. 2. Schematic interface description (cf. [19])

other, the connection mechanism supports mapping multiple PLC variables to different fields of information objects.

Currently, semantics in the exchanged information objects is not represented, albeit the lack of automated re-planning support.

3.3 Towards Automated Planning

Preparation steps need to be executed by all agents prior to any collaborative activity. Consequently, in order to maintain interoperability in case of events triggering change, either automated re-planning or manual re-design has to be performed [20].

To automate process planning in production, both is needed, a local detailed resource-centric/agent-centric point of view and a collaboration/production-process point of view. Distinguishing between these two layers allows to project the particularities of the concrete hardware and resources to a middleware.

The resource point of view is needed to encapsulate hardware specific aspects. The network point of view is needed to align all process steps/segments with a common (global) product process.

To allow local autonomy and global synchronization in a *process-centric automated control environment* with automated re-planning, there is a need for a framework that (cf. [17, 21]):

– provides a middleware and abstract data-model for adaptability
– provides services supporting automated re-planning (e.g. directory of skills; logging for quality assurance, etc.)

- supports human and artificial production resources on the one hand side and the network layer for connectivity
- allows for decentralized specification and execution of process segments for different production resources
- supports the model concepts and their semantics (tasks, processes, agents, skills, resources, … see above)

4   Conclusions

The initial work in the SO-PC-Pro project shows a running environment for executing processes partly by humans, partially on machines. This work reveals that the abstraction using a paradigm similar to agents supports the coordination of humans and machines in a single process. This is an important aspect towards a middleware that supports automated control of production processes executed by IoT systems, robots, and humans. Still much more research has to be done, to support an abstract production process description to be automatically planned and deployed in such an environment. This does not only involve the organizational interoperability levels, but also the semantic and technical interoperability levels.

References

1. Vernadat, F.B.: Technical, semantic and organizational issues of enterprise interoperability and networking. Ann. Rev. Control **34**, 139–144 (2010)
2. Romero, D., Vernadat, F.B.: Enterprise information systems state of the art: past, present and future trends. Comput. Ind. **80**, 3–13 (2016)
3. Ducq, Y., Chen, D., Doumeingts, G.: A contribution of system theory to sustainable enterprise interoperability science base. Comput. Ind. **63**, 844–857 (2012)
4. He, W., Xu, L.D.: Integration of distributed enterprise applications: a survey. IEEE Trans. Ind. Inform. **10**, 35–42 (2014)
5. Zacharewicz, G., Diallo, S.Y., Ducq, Y., Agostinho, C., Jardim-Goncalves, R., Bazoun, H., Wang, Z., Doumeingts, G.: Model-based approaches for interoperability of next generation enterprise information systems: state of the art and future challenges. IseB **15**, 1–28 (2016)
6. Chen, D., Doumeingts, G., Vernadat, F.B.: Architectures for enterprise integration and interoperability: past, present and future. Comput. Ind. **59**, 647–659 (2008)
7. Tu, Z., Zacharewicz, G., Chen, D.: A federated approach to develop enterprise interoperability. J. Intell. Manufact. **27**, 11–31 (2014)
8. Lalanda, P., Morand, D., Chollet, S.: Autonomic mediation middleware for smart manufacturing. IEEE Internet Comput. **21**, 32–39 (2017)

9. Weichhart, G., Guédria, W., Naudet, Y.: Supporting interoperability in complex adaptive enterprise systems: a domain specific language approach. Data Knowl. Eng. **105**, 90–100 (2016)
10. Naudet, Y., Latour, T., Guédria, W., Chen, D.: Towards a systemic formalisation of interoperability. Comput. Ind. **61**, 176–185 (2010)
11. Agostinho, C., Jardim-Goncalves, R.: Sustaining interoperability of networked liquid-sensing enterprises: a complex systems perspective. Ann. Rev. Control **39**, 128–143 (2015)
12. Weichhart, G., Feiner, T., Stary, C.: Implementing organisational interoperability – the SUddEN approach. Comput. Ind. **61**, 152–160 (2010)
13. Mehandjiev, N., Grefen, P. (eds.): Dynamic Business Process Formation for Instant Virtual Enterprises. Springer, London (2010)
14. 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)
15. Bénaben, F., Mu, W., Boissel-Dallier, N., Barthe-Delanoe, A.-M., Zribi, S., Pingaud, H.: Supporting interoperability of collaborative networks through engineering of a service-based Mediation Information System (MISE 2.0). Enterp. Inf. Syst. **9**, 556–582 (2015)
16. Weichhart, G., Wachholder, D.: On the interoperability contributions of S-BPM. In: Nanopoulos, A., Schmidt, W. (eds.) S-BPM ONE 2014. LNBIP, vol. 170, pp. 3–19. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-06065-1_1
17. Neubauer, M., Krenn, F., Majoe, D., Stary, C.: Subject-orientation as design language for integration across organisational control layers. Int. J. Prod. Res. **55**(13), 3644–3656 (2017)
18. Fast-Berglund, Å., Palmkvist, F., Nyqvist, P., Ekered, S., Åkerman, M.: Evaluating Cobots for Final Assembly Procedia. CIRP **44**, 175–180 (2016)
19. Neubauer, M., Stary, C. (eds.): S-BPM in the Production Industry. Springer International Publishing, Heidelberg (2017)
20. Åkerman, M., Fast-Berglund, A., Ekered, S.: Interoperability for a dynamic assembly system procedia. CIRP **44**, 407–411 (2016)
21. Dar, K., Taherkordi, A., Baraki, H., Eliassen, F., Geihs, K.A.: A resource oriented integration architecture for the Internet of Things: a business process perspective. Pervasive Mobile Comput. **20**, 145–159 (2015)