



HAL
open science

An IFIP WG5.8 State-of-the-Art View on Methods and Approaches for Interoperable Enterprise Systems

Georg Weichhart, Yves Ducq, Guy Doumeingts

► To cite this version:

Georg Weichhart, Yves Ducq, Guy Doumeingts. An IFIP WG5.8 State-of-the-Art View on Methods and Approaches for Interoperable Enterprise Systems. *Advancing Research in Information and Communication Technology, AICT-600*, pp.222-244, 2021, 10.1007/978-3-030-81701-5_9 . hal-03325982

HAL Id: hal-03325982

<https://inria.hal.science/hal-03325982>

Submitted on 25 Aug 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

An IFIP WG5.8 state-of-the-art view on methods and approaches for Interoperable Enterprise Systems

Georg Weichhart¹[0000-0002-1405-5825], Yves Ducq²[0000-0001-5144-5876], Guy Doumeingts³

¹PROFACTOR GmbH, Austria
georg.weichhart@profactor.at

²Univ. Bordeaux, France
yves.ducq@u-bordeaux.fr

³INTEROP-VLab (B)
guy.doumeingts@interop-vlab.eu
guy.doumeingts@u-bordeaux.fr

Abstract. IFIP's workgroup 5.8 on Enterprise Interoperability was founded in 2008. In this paper we provide a workgroup point of view on the state of the art in Enterprise Modelling, Enterprise Engineering, Enterprise Architectures, Enterprise Integration, Enterprise Interoperability. We present an overview of the state-of-the-art from the WG5.8's research point of view. A brief history of these topics, with special attention to the work developed by IFIP WG5.8 former and current members is given. With respect to application, references to production systems and the manufacturing enterprise will be made. This article closes with a brief look into very recent developments in the domain of Enterprise Interoperability.

Keywords: Enterprise Interoperability, Enterprise Integration, Enterprise Architecture, Enterprise Engineering.

1. Introduction

Over the last decades, enterprises have to improve their performance to compete on a global scale. Improvements are required along multiple dimensions as capabilities of organizational learning, financial short- and long-term perspectives, human resource development, etc. [1]. From a manufacturing enterprise point of view, products and processes and their relations are highly relevant. For this paper the manufacturing domain serves as the prime example, due to the high level of automation, the great potential provided by *Enterprise Interoperability* research can be made visible. In manufacturing there are three dimensions for improvement of relevance. The efficiency dimension requires to take a close look at the used production resources to minimize the amount of resources consumed (and hence reduce also costs). To get the desired quality of the product, the production process needs to be effective to meet the desired quality range. Flexibility, a newer dimension, requires the enterprise to adapt to changing demands and/or other events like supply chain disturbances. This later dimension has evolved in the manufacturing industry in recent years where product

variants have increased, with the vision to have the fully customized product, and where processes are continuously reconfigured [2].

The requirements cannot be considered independently, flexibility of production systems needs to be increased, while the same level of quality and costs has to be maintained [3]. The enterprise and its employees need to continuously learn, improve and the production system has to adapt [4]. Adaptivity is a system's capability to reorganize itself, when coping with changing situations or with disturbances. In addition to this, the production processes and the product type are getting more complex [5]. A system is complex when many hard to understand factors are defining the system. Hard to understand, means that the relationship between factors is nonlinear, hard to observe and/or very dynamic [6]. This complexity is grounded not only in new technologies, but also in the system's ability to evolve and adapt [7].

To support the enterprise system to adapt, handle complexity and still have efficient and effective processes, we describe approaches in the following, which support modelling, engineering, and evolution of enterprise systems.

The following section describes the enterprise as a complex adaptive system. This is followed by an overview on enterprise modelling and enterprise architecting. Enterprise Architectures are described as a special form of enterprise modelling / engineering which supports Enterprise Integration and Enterprise Interoperability. These topics are discussed in later sections.

2. Complex Adaptive Enterprise Systems

For being able to address the enterprise as a system, we take a look at Systems Theory and General Systems Theory (GST) [8]. GST aims at the identification of general principles that are valid across heterogeneous types of systems. GST facilitates the communication between different scientific disciplines through abstraction. The essential concepts of a discipline are described as systems. This abstraction allows scientists of other disciplines to use elements and insights described by systems from other domains [9].

A system is an integrated unit which can be decomposed into a set of tightly connected units. Each unit provides a function which contributes to the overall goal (purpose) of the system. The units and the system do have interfaces that separates them from their environment. A system is independent and autonomous within its environment. In contrast to that, do functional units need the system and the other units.

A system shows a behavior and has a state. It therefore has a history. Sometimes a system contains other systems. In this view, multiple systems form a system-of-systems [10]. The systems in a system-of-systems are independent and hence there is a loose coupling between systems in a system-of-system.

Sometimes systems are described as active units, named agents. Such agents are independent and autonomous - like all systems. Agents convey a notion of intelligence and (pro-) active behavior [11, 12]. Such agent-based systems are called Complex Adaptive Systems, where special attention is placed on the dynamics and time-dependent behavior of the overall system [13, 14].

The interaction of systems in a system-of-systems can be described on multiple levels of granularity. The environment within which the technical systems are found

are the services and functions executed. The environment within which these services and functions are executed is the enterprise. The environment within which the enterprise system exists are legal, social and economic constraints and requirements [15].

Taking a look at enterprises these can be described as information processing systems [16]. The following figure shows different levels in an enterprise system where different human and artificial agents are communicating. Information needs to be conveyed, understood, decisions are influenced by the information and finally actions are triggered by participants of any communication.

The bottom most level addresses the physical transport of information. The next two levels are about having structures that can be read by all participants. Semantics discusses the interpretation of information by different agents. Pragmatics is about actions triggered by the communication. Executed actions need to be compatible within the social world.

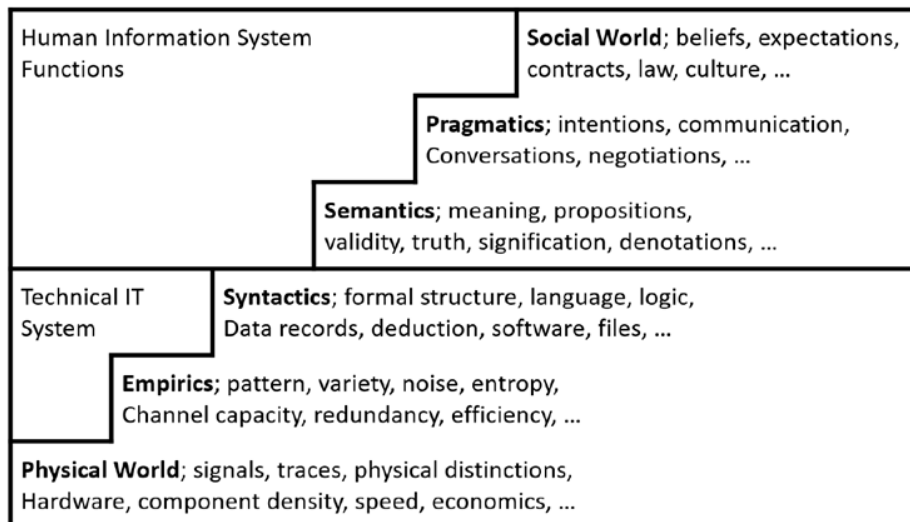


Fig. 1. Organizational Semiotics: Stamper's Semiotic Ladder [16]

Enterprises and Networks of Enterprises can also be described as Complex Adaptive Systems [17, 18]. (Human / Artificial) Agents are interacting with the Enterprise System. There is no organization as such that exercises control, but the agents. However, some agents will have more power than others.

The agents act self-controlled and self-organize as a group establishing the enterprise. Interactions are local between a few agents. In these interactions the agents may be taking a higher system state into account (i.e. an Enterprise-global state). Nevertheless, the overall behavior of the enterprise emerges from these interactions [19].

In any social system, agents' behaviors influence their environment and the environment influences the agents' behaviors [20]. This can be seen when agents learn from each other. Learning and improvement of agents' behaviors are results of self-organization of individual agents and of groups of agents.

To be able to address complex situations, which includes learning, models are helpful. For Enterprises several approaches to Enterprise Modeling and Enterprise Engineering exist, which are discussed in the next section.

3. Enterprise Modelling & Enterprise Engineering

Modelling is a process where abstract representations of the world are developed. Process of abstraction is guided by the goal for which the model is constructed. Semantic models of the enterprise or parts of the enterprise and its functions help to cope with the complexity of the overall system [21]. Models may be used for knowledge management supporting communication between agents. Collaborative learning is supported by collaborative modelling [22, 23]. Models may also be used for control and decision making. The interactive possibilities for groups to evolve models supports evolution of the common understanding of the group [24].

Enterprise Models, by definition, capture what is seen by the modelling agents as important aspects relevant to reach the goal of the model. There is no widely accepted definition of enterprise modelling [25]. On an informal level enterprise modelling is a set of activities that deal with externalizing knowledge about organizational structures and behaviors of an overall enterprise or parts of it.

Typical applications of enterprise modelling are [25]:

- Documentation of Enterprise Systems: Such documentations support analysis, understanding, and sharing of knowledge related to the enterprise.
- Planning and Management of change: Models are built to document the current situation (as-is) or a vision of a future situation (to-be). This supports the transition and management of change.
- Designing and (Re-)Engineering of Enterprise Systems: Following the model-driven engineering approach [26] detailed specifications of the enterprise can be used also for automation e.g. of business processes.
- Performance Analysis: The as-is or to-be performance of an enterprise can be provided. A very abstract model are performance measurement systems which provide a set of dimensions of performance indicators. Very concrete models are simulation models, which allow to evaluate dynamic aspects of enterprise systems in different situations.
- Enterprise Systems Integration: Enterprise models support the integration of enterprise information systems.

For Enterprise Modelling, the following principles have been identified for the modelling process, capturing a large and complex system-of-systems like the enterprise [21]:

- **Principle #1 – Plural nature of enterprise models:** Complexity and size of enterprise systems imposes the need for multiple models and renders a single large model of the enterprise unfeasible. It is simply not possible to construct a model from a single point of view.
- **Principle #2 – Concept of modelling views:** Every model has to support different points of views. The concept of views reduces the complexity, and

lets users focus on individual aspects of the model per view. In the domain of enterprise modelling essential views are listed below. However, this is not an exclusive list but rather highlights important views that can be found in multiple approaches (cf. [27–31]).

- Function View: functionality provided by processes describing the contributions of agents and the behavior of the enterprise
- Information View: this shows the data, information and knowledge flows and also contains who has access to what.
- Resource View: this view includes the human and artificial agents and passive objects (like materials) that are required for the execution of tasks within the processes
- Organization View: this describes roles and responsibilities and rights to access resources
- **Principle #3 – Coverage of the three fundamental types of flows:** Enterprise models do not only capture structural relationships; it must be possible to include dynamic aspects (i.e. flows). Flows are relevant to enterprises in general and to manufacturing enterprises in particular are:
 - material flow (physical systems flow; this includes power or cooling fluids; any physical resource that is required to perform an activity)
 - information flow (data, information, knowledge and decisions)
 - control flow (workflow, processes).
- **Principle #4 – Concept of modelling levels:** Such mechanism also supports management of the complexity of enterprise models. Examples for different levels of abstraction are: the requirements, design and implementation levels of CIMOSA [27, 32] and ARIS [28], conceptual, organizational and physical levels of GRAI-GIM [31, 33].

Enterprise Engineering is grounded in Enterprise Modelling and in addition to the above, this aims at providing concise models. While there is no common agreed upon definition, Enterprise Engineering can be described as follows:

“Enterprise engineering: define, structure, design and implement enterprise operations as communication networks of business processes, which comprise all their related business knowledge, operational information, resources and organisation relations.” p85 [27].

Three generic goals to Enterprise Engineering have been identified [34]:

- Intellectual manageability: Support the construction and operation of enterprises to work with the above described complexity. Support for managing change and evolution of enterprises is needed.
- Organizational concinnity: Establish a theory and method that supports to address all relevant aspects to be able to operate the enterprise as a coherent and consistent whole.
- Social devotion: Enterprise Engineering aims at the including employees. Humans are important for all aspects of the enterprise, in particular for aspects of dynamics, complexity and uncertainty. Enterprise Engineering aims at empowering humans to gain knowledge.

Enterprise Engineering approaches support analysis, synthesis, decision making, explanation and prediction [34]. The approaches that support the basis for development and design of systems are related and support enterprise architecture engineering. This design aspect of Enterprise Engineering (and Enterprise Modelling) is based on general systems theory (GST) [9] and design science [35]. Selected enterprise architecture frameworks are discussed in the next section.

4. Enterprise Architecture Frameworks

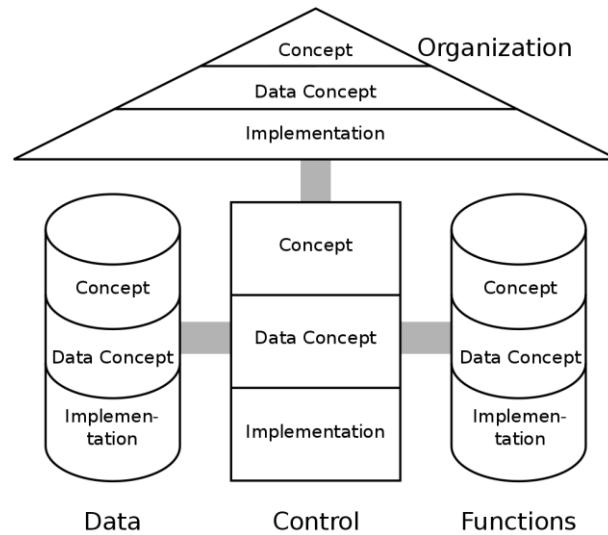
Architectures are models that provide a coarse-grained structure. Like architectures of buildings, these provide rooms and interfaces between rooms (doors). A building's architecture does not prescribe how the rooms are decorated. However, already functionality is assigned to particular rooms like kitchen, bathroom, etc.

Another type of architecture are software architectures: "Software architecture is the principled study of the overall structure of software systems, especially the relations among subsystems and components. From its roots in qualitative descriptions of useful system organizations, software architecture has matured to encompass broad explorations of notations, tools, and analysis techniques. ... it now offers concrete guidance for complex software design and development. Software architecture research overlaps and interacts with work on software families, component-based reuse, software design, specific classes of components (e.g., COM), product line, and program analysis." p257 [36]

Enterprise Architecture applies the above described aspects to enterprise systems. Enterprise Architectures aim at supporting modelling uniform representations and supporting knowledge transfers and integration / interoperability of different domains across the enterprise [30]. With Enterprise Architectures, often an information system centric perspective is taken, but Enterprise Architectures are used to model many different parts of organizations including: organizational structures and processes, information applications, and software and hardware infrastructure [37].

An example of an information system centric architecture modelling standard, is ARIS (Architecture of Integrated Information Systems) [28]. It provides different views to ensure the enterprise information system meets its requirements. These views are shown in the following image.

The Organization view allows to model the structure of a company, including human resources, machines, hardware and their relationships. The Data view provides information about events generating data and business objects and the relationship. The function view shows activities, groupings and the relationship. The main view is the control view where business processes are modelled which integrate objects from all other views. All the views have three levels of abstraction. The finest granular level is the implementation level, using software [28].

Fig. 2. ARIS house¹

ArchiMate® is an enterprise architecture modelling language that also puts emphasis on information systems but follows a service-oriented approach [37]. ArchiMate® is a “visual language with a set of default iconography for describing, analyzing, and communicating many concerns of Enterprise Architectures as they change over time” [38]. ArchiMate® is a registered trademark of The Open Group. The current version of the ArchiMate Core language consists of four layers and three aspects as shown in the following figure.

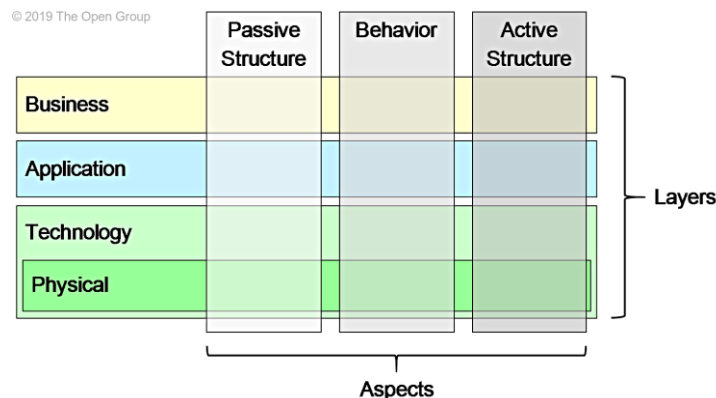


Fig. 3. ArchiMate® Core Framework [38].

The business layer describes roles and business processes that need support by software services. These services and software modules that provide the services are described on the application layer. The software modules need infrastructure services (like

¹ https://en.wikipedia.org/wiki/Architecture_of_Integrated_Information_Systems

database services) that are described on the technology layer (which includes physical aspects like servers). Passive structures are for example data structures (application layer) and business objects (business layer). Behavioral elements are business processes and application functions. Active structures are business roles and application components [38].

TOGAF® is also a registered Trademark of the Open Group. The TOGAF® standard describes an Enterprise Architecture framework that “provides the methods and tools for assisting in the acceptance, production, use, and maintenance of an Enterprise Architecture. It is based on an iterative process model supported by best practices and a re-usable set of existing architecture assets.” [39]. It guides the Enterprise Architect by providing an Architecture Development Method (ADM). The method “includes establishing an architecture framework, developing architecture content, transitioning, and governing the realization of architectures. All of these activities are carried out within an iterative cycle of continuous architecture definition and realization that allows organizations to transform their enterprises in a controlled manner in response to business goals and opportunities.” (section 2.4) [39].

TOGAF® supports the following phases for architecture development and change (see fig. 4).

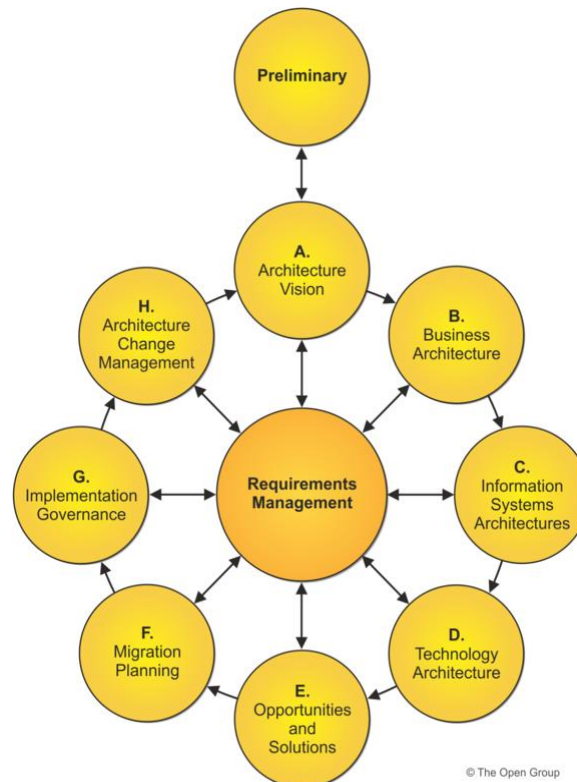


Fig. 4. TOGAF® Phases [39]

It also addresses the business architecture which could be described in the ArchiMate® business layer. Software Applications are supported in the information system

architecture. And the models of the ArchiMate® technology layer can be of use in the technology architecture phase. More phases support the change and evolutions of architectures.

Several Enterprise Architecture Frameworks have been developed for the manufacturing domain. Here, a major driving force for modelling and architecting of large and complex systems has been Computer-Integrated Manufacturing (CIM). The developed frameworks have general applicability to many domains.

The CIMOSA Framework (Computer Integrated Manufacturing Open System Architecture) has been created to support the Enterprise Integration of machines, computers and people [27, 32]. CIMOSA is one of the frameworks (obviously) grounded in the manufacturing domain. The framework provides a process-oriented modelling concept. It allows to model both, the process functionality and the process behavior. Evolutionary Enterprise Modelling and individual enterprise domains (DM) are possible. CIMOSA separates functionality (EA: Enterprise Activity) from the behavior (BRS: Behavioral Rule Set). One might be changed without the other. This allows to break large processes down into smaller processes, ending in networks of EAs which are connected by the BRS [27].

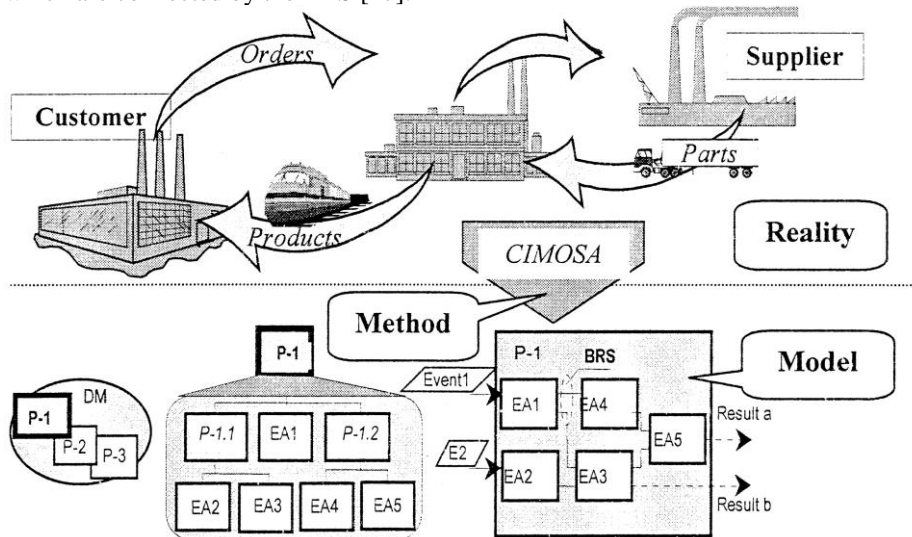


Fig. 5. CIMOSA Modelling Concepts (DM: Domains; EA: Enterprise Activity; BRS: Behavioral Rule Set; P-x: Processes; E: Event) [27]

Another approach routed in Computer Integrated Manufacturing (CIM) is GRAI (Graph with Results and Activities Interrelated) and GIM (GRAI Integrated Methodology [31, 40] (see also fig. 6).

The GRAI Model supports three systems that can be found in any enterprise. The physical system, the information system and the decision system. It provides a reference architecture and modelling approach which addresses three aspects: Modelling Views, Life-Cycles of systems, and multiple levels of abstraction. It also provides the GIM structured approach, which aims at guiding the analysis and the design of (manufacturing) facilities during three phases: the analysis phase, the user-oriented design phase and the technical-oriented design phase [41]. GIM supports also the

analysis of decision systems of flexible manufacturing and CIM systems from a production management perspective [31, 40].

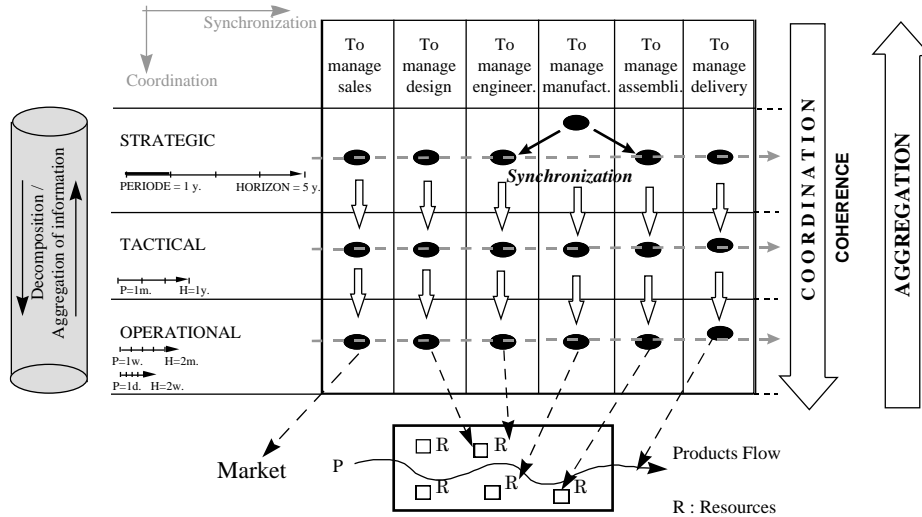


Fig. 6 The GRAI Model [31]

GERAM² (Generalized Enterprise Reference Architecture and Methodology) has been created by a Task Force formed by IFIP WG 5.12 (International Federation for Information Processing - Workgroup - Architectures for Enterprise Integration) and by IFAC TC 5.3 (International Federation of Automatic Control - Technical Committee - (now named) Integration and Interoperability of Enterprise Systems) [29]. The Task Force brought together different Architecture Frameworks taking important aspects from each. The approach evolved into the standard ISO15704, "Requirements for Generalized Enterprise Reference Architectures and Methodologies" [42]. The standard's mandatory section lists requirements for any enterprise reference architecture, and GERAM is a public appendix to the standard, exemplifying how to fulfil them [29].

One of GERAM's key contribution is its definition of a concise set of concepts for combining and using these for evaluation of architecture frameworks to understand their completeness with respect to scope and complexity [29]. The following figure shows the framework components that form this standard.

² <http://www.ict.griffith.edu.au/~bernus/taskforce/geram/versions/geram1-6-3/v1.6.3.html>

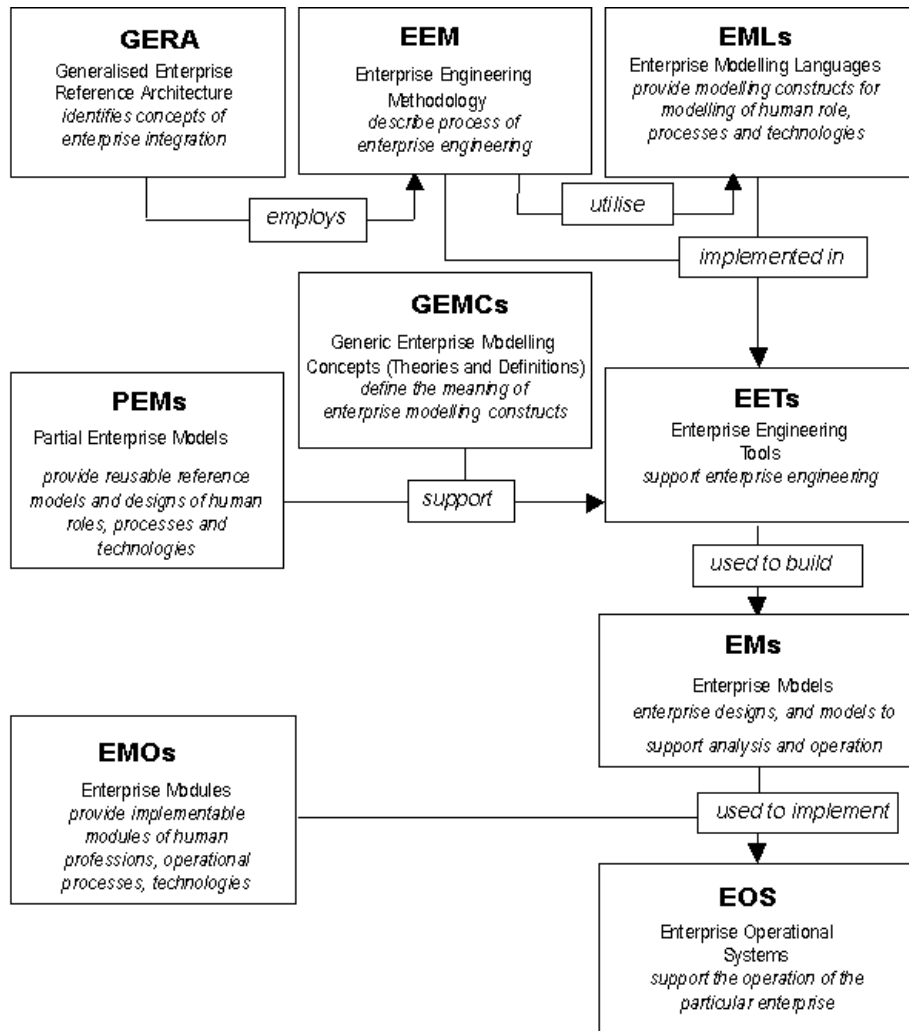


Fig. 7. GERAM Top Level Elements [43]

GERA (Generalized Enterprise Reference Architecture), which is the most important component, holds the basic concepts that are to be used in enterprise engineering and integration. This includes enterprise entities, life-cycles, life histories of enterprise entities, etc. [43].

The Model Driven System Engineering Architecture (MDSEA) follows the Model Driven Architecture (MDA) as defined and adopted by the Object Management Group [44]. MDSEA was developed by EU Project “Manufacturing Service Ecosystem” (MSEE), supporting the transition of enterprises from product manufacturing to product-service production [25, 45]. The MDA approach is adapted to be suitable for a broader modelling domain like product-service supply networks consisting of multiple enterprises.

The following figure shows the abstract architecture applying MDSEA levels to the supply network building of services of virtual enterprises.

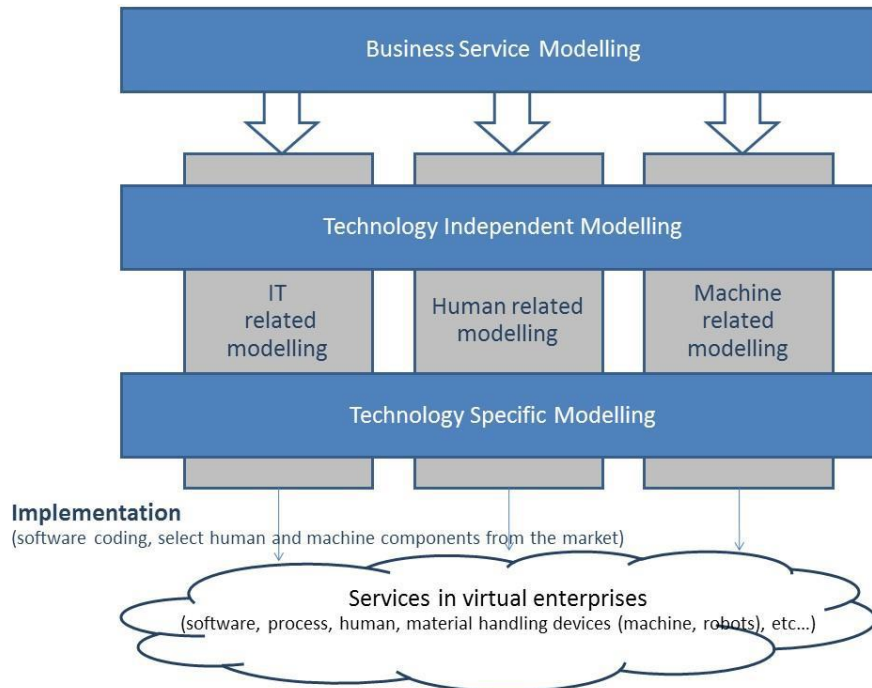


Fig. 8. MDSEA Abstract Architecture [46]

Business Service Modelling provides technology independent high-level models of provided services and links between enterprises. Here models are used to bridge the gap between knowledge of domain experts and the technical knowledge on lower levels. The Technology Independent Model provides detailed specifications with respect to the concepts related to the information systems, organization and human and physical means for a service system. It is still independent from the technology used for implementation. At the Technology Specific Model level (as the name suggests) the model specifies a particular type of technology used for implementation. The level of detail must be high enough to be able to realize or buy a specific solution.

As already mentioned above, enterprise modelling, enterprise engineer, enterprise architecting support the integration of enterprise systems. In the following section we will take a closer look at theories and approaches of Enterprise Integration and Enterprise Interoperability.

5. Integration and Interoperability of Enterprise Systems

Enterprise Integration is an approach for ensuring the interaction of systems to ensure that a common objective is achieved [47, 48]. This interaction can happen on different

levels: Data exchange with integrated technical formats, information models describing a common ontology, interacting business services that contribute to the same business objective, business processes that need to be integrated so the workflows have seamless interfaces.

Full integration implies that the model used for communication is the same as internally used for reasoning and decision making. For more loose coupling Interoperability needs to be discussed. Enterprise Integration can be defined as follows: "Enterprise integration: provide the right information at the right place and at the right time and thereby enable communication between people, machines and computers and their efficient cooperation and coordination" p.85 [27]

Enterprise Interoperability is extending Enterprise Integration towards the requirement to have loose coupled systems, or even systems-of-system. Interoperability is not a single goal that needs to be reached. It also has a process connotation because - due to the independence of systems in a system-of-systems - it is assumed that sub-systems will evolve and interoperability needs to be monitored and re-established as soon as it is lost [33, 49]. In a system-of-systems the systems are not connected in a static manner. In contrast to that, integration is a (final) state where all elements of a system can exchange and understand information. This implies that the model used for communication is also integrated on a semantic and a pragmatic level by the involved elements [18].

The Framework for Enterprise Interoperability (FEI) is an approach developed by the GRAI Research Group (today GR Productique) at University of Bordeaux.

FEI aims at structuring basic Enterprise Interoperability concepts and issues. The framework has three basic dimensions: (i) Interoperability concerns that define the content of interoperation that may take place at various levels of the enterprise (data, service, process, business) (ii) Interoperability barriers that identify various obstacles to interoperability in three categories (conceptual, technological, organizational), (iii) Interoperability approaches that represent the different ways in which barriers can be removed (integrated, unified, and federated). The intersection of an interoperability barrier, an interoperability concern and an interoperability approach is the set of solutions to the breakdown of that interoperability barrier existing for the particular concern and using the selected approach.

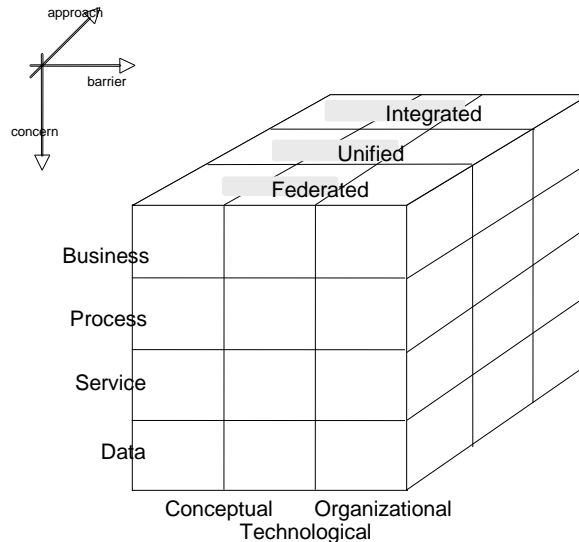


Fig.9. Framework for Enterprise Interoperability [33,50]

In the ATHENA project different levels (already indicated above) have been identified [33, 51, 52]. To all levels, all barriers identified above are relevant. Similar to above, selected examples are given.

- **Data:** Data models and programming interfaces are the finest granular level of detail. Exemplary organizational barriers are access rights, and privacy issues.
- **Services:** This level addresses business services providing functions. Barriers include incompatibilities in the granularity of a service offered or management rights for services that lead to interoperability issues.
- **Processes:** Process models define the behavior of organizations and are subject to interoperability barriers like different coverages of overlapping processes; technological issues of workflow systems; incompatible processes of collaborating organizational units.
- **Business:** conceptual barriers concern the strategy of different collaborating enterprises; or the degree of computerization of units; Legal aspects of organizations can also be an issue.

The above dimensions “barriers” and “concerns” represent the problem space. Solutions are discussed along a third dimension. The tightest level of interoperability is integration. Here the same model is shared between different systems. Hence modifications require all systems to be modified. To increase the loose coupling, interacting systems can agree on a unifying layer. That layer defines the interaction and data on a higher level of abstraction. That layer is known by all systems but not used system-internally.

The degree of increased “looseness” is a federated approach to interoperability where interoperability is negotiated at runtime. Here infrastructure is needed to support

that negotiation. For example, management systems to monitor and manage the discovery of services.

Going beyond a federated approach to interoperability means compatibility. However, this marks systems that are not collaborating, but are only not-disturbing each other.

A different framework to interoperability is the European Interoperability Framework [53]. This approach has roots in European, Governmental, Organizational Interoperability. As such it has a legal level, which can be interpreted similar to the “social” level of Stamper’s semiotic ladder [16]. The legal level represents (explicit) social norms (laws) to which any enterprise operation must be interoperable. In democratic societies, laws represent explicit social demands and requirements of the society.

In contrast to the framework above there is no conceptual barrier but rather a semantic level, which basically addresses the same need.

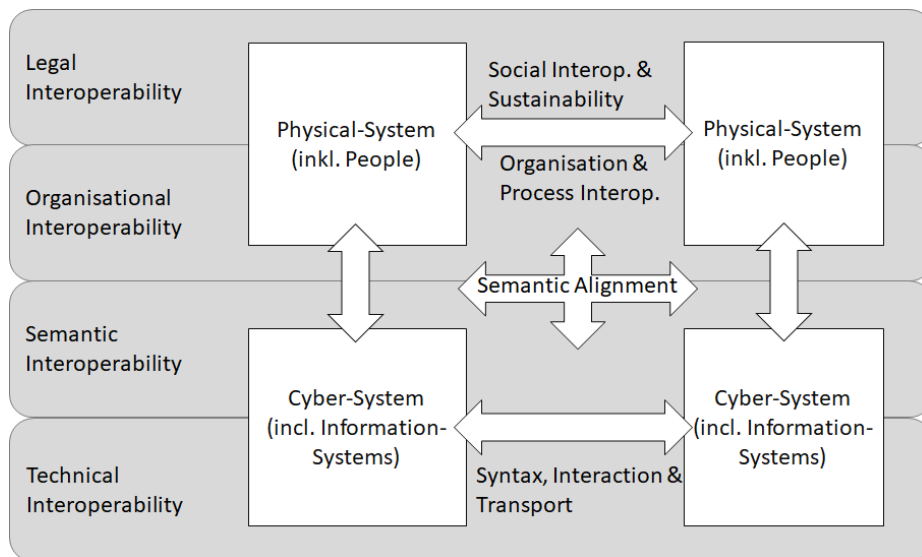


Fig. 8. European Interoperability Framework [54]

Several approaches exist on the different layers. On technical level several standards exist supporting connectivity. For example, in manufacturing, OPC-UA is an approach that supports access to remote machine data and the execution of programs [55]. OPC-UA also supports interoperability with the information model [56]. When combined with the Automation ML Standard [57] it supports heterogeneity and variability [58, 59].

Approaches that target the semantic aspects are often based on ontologies. The Ontology of Enterprise Interoperability (OoEI) [49] is such an approach. It formalizes the enterprise as a system and provides a number of concepts for modelling interoperability of enterprise systems. Core concepts of the OoEI are built on top of General Systems Theory [9]. It allows to describe relationships between systems.

In order to address the enterprise as a complex adaptive system (CAS), the OoEI has been extended towards an Ontology of Enterprise Interoperability extended for CAS (OoEI^{CAS}) [60]. This approach also makes a move away from the ontological approach towards a Domain Specific Language (DSL). It captures the systemic core of the OoEI and represents it as data structure in OoEI^{CAS}. In this approach the actor-model is used to provide a means to implement agents [61, 62]. Using this approach allows to capture active organizational systems and their interaction. The Domain Specific Language (DSL) is implemented in the functional programming language SCALA [63].

An approach that is focusing on the organizational level is MISE (V1 and V2), which provides methodological and tool support for emerging collaborative situations in production systems [64, 65]. It supports dynamic collaborations of networked organizations which are seen as complex adaptive systems. MISE has a business level where the collaborative situation of multiple enterprises is modelled. The technical level provides a workflow model. MISE has services that support the deployment, orchestration and runtime. An agility infrastructure supports adaptation and evolution of models at different levels. It is based on the view of the enterprise as a complex adaptive system [64].

The Liquid Sensing Enterprise (LSE) [66] and the S³Enterprise (Sensing, Smart and Sustainable Enterprise) [46] also have as a basic assumption that the enterprise system is a complex adaptive system. From an implementation perspective the LSE builds on the High-Level Architecture (HLA) simulation standard for supporting dynamics and interaction. The S³Enterprise Approach sees a strong need for an enterprise operating system (EOS) that supports a unified information flow between the sensors and the smart algorithms for decision making and the human decision makers as well.

The Enterprise Operating Systems (EOS) [67, 68] addresses the need for support of integration of heterogeneous enterprise software applications, which is a complex, costly and time-consuming task, especially for SMEs. An “EOS can be defined as a set of IT services supporting execution, control and monitoring of enterprise operations by means of message exchange among enterprise resources, model enactment and interoperability capabilities” p. 2716 [68].

The basic conceptual building blocks are shown in the next graphic. As can be seen the EOS combines several existing functionalities in an overarching system, similar to operating systems of computers.

A current implementation of the EOS exists based on the High-Level Architecture (HLA) [67]. Nevertheless, the conceptual approach is technology independent.

The EOS does provide key steps towards manufacturing systems of the future, enabling real-time enterprise operations monitoring and control. The EOS approach does include and is in line with IIoT (Industrial Internet of Things) and CPPS (Cyber-Physical Production System) principles.

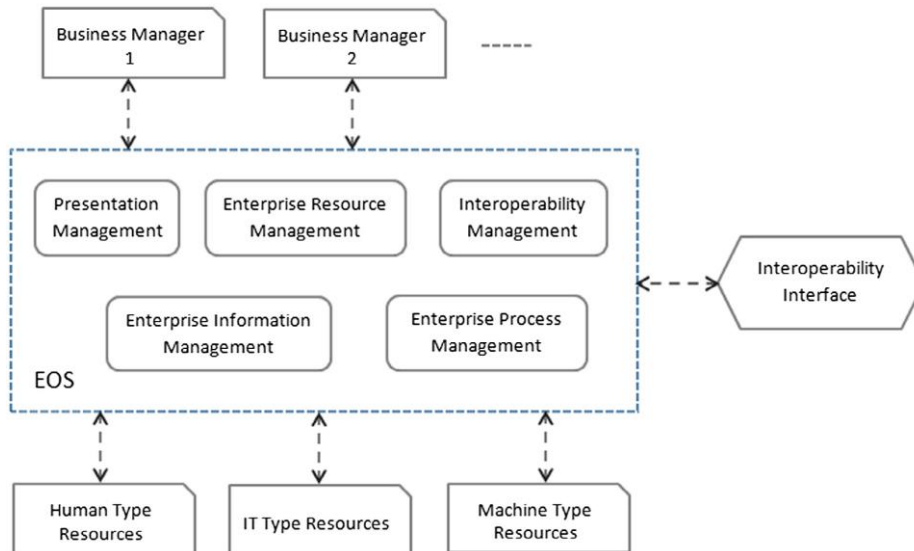


Fig. 9. Enterprise Operating System Building Blocks [68]

6. Towards Interoperability of Cyber-Physical Production Systems (CPPS)

In the most recent years of IFIP WG5.8 “Enterprise Interoperability” members have been starting towards extending the research to include the CPS (Cyber-Physical Systems), CPPS (Cyber-Physical Production Systems) and IIoT (Industrial Internet of Things) [26, 46, 69].

A Cyber-Physical System is a system, which includes a physical and a cyber part. Only both combined bring the required functionality. Such systems have sensors and actors that provide this interaction between both worlds. In addition to this, the software often supports the networking of such systems. Multiple such systems form in the manufacturing domain Cyber-Physical Production Systems [70, 71, 72].

In manufacturing one of the main challenges of a modular and agile CPPS is, to disintegrate the current architecture called Automation Pyramid and modularize manufacturing systems. This leads to a situation where modules provide their functionality to many different systems. However, the original pyramid architecture is in place due to different timing constraints and planning horizons that the systems have on the different levels. Unfortunately, that hierarchical organization has been shown to be very rigid. The new CP(P)S type of organization addresses the missing adaptability.

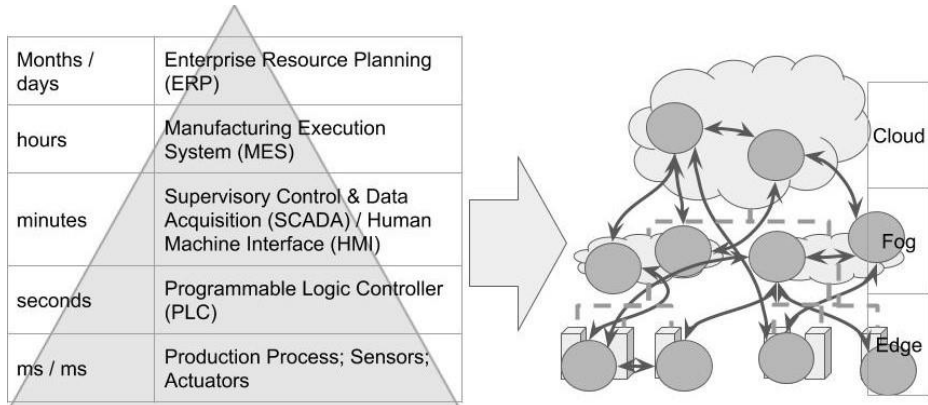


Fig. 10. Automation Pyramid Architecture evolving into a CPPS Architecture (based on [73])

The CPPS perspective provides new challenges which require addressing the enterprise from the physical parts over sensors and data streams to an adaptive and modular network of systems. One of the challenges is the need for new planning and engineering approaches that address Systems-of-Systems, Agent-based Systems and/or Cyber-Physical Production Systems [74,75].

The Model Driven System Engineering Architecture (MDSEA) and its extension to interoperability concerns called MDISE (Model Driven Interoperability System Engineering) framework are currently extended towards CPS as well [26].

A second fundamental challenge that needs to be solved to enable CP(P)S to provide its full potential is interoperability of such systems [54]. A proper interoperability environment needs to be researched to enable the modular and adaptive nature of a CPPS. Otherwise, the CPSs will be hardwired and the modularization will be more overhead than advantage. This challenge is addressed in the vision of the EOS [46, 68].

7. Conclusions

This article gives an overview of essential developments that have been driven by members of the IFIP WG5.8, even before the founding of this group in 2008. The authors are founding members of IFIP WG5.8. Guy Doumeingts has been the first chair of the workgroup. Yves Ducq served on the board since the beginning of the workgroup and is currently treasurer. Georg Weichhart is WG chair since 2019.

In this article we have reported on several approaches and future challenges in the research field of Enterprise Interoperability and its neighboring fields.

The work developed in WG 5.8 originated in the work of INTEROP Network of Excellence and ATHENA Integrated Project. The work on Enterprise Interoperability is also supported by INTEROP-VLab that was created after the INTEROP NoE ended.

The main methodological principles are to use Enterprise Modelling, Enterprise Architecture and Ontologies to solve problems of interoperability for enterprise applications.

Some of the work mentioned in this article is coming from national and European projects that were carried out from the starting of WG 5.8.

For instance, the first interoperability framework that was developed in the frame of the IDEAS European project was improved in INTEROP-NoE (Network of Excellence) to provide Framework for Enterprise Interoperability (FEI). Another example, concerning Enterprise Modelling for Enterprise Engineering, is the development of a Model Driven Service Engineering Architecture (MDSEA) within the frame of MSEE (Manufacturing Service Ecosystem), a European Project supported by the European Commission

So, the philosophy of the WG 5.8 is to continue to contribute to the development of interoperability topics with sharing the work done by all the partners through their own and common project and to share their experience and application case studies. Of course, some of the WG5.8 members being also members of INTEROP-VLab, which plays a main role to drive the development of this topic at the European and international levels.

In a most recent development, IFIP WG5.8 “Enterprise Interoperability” (see also <http://www.ifip-ei.org>) has teamed up with IFAC TC5.3 “Integration and Interoperability of Enterprise Systems” (see also <http://www.ifac-tc53.org>), forming a joint task force to work on new methods and approaches of modelling and architecting for interoperable Cyber-Physical Systems [76]. This will support to drive the research on both CPS and Enterprise Interoperability.

8. Acknowledgement

This work has been supported by Pro²Future (FFG under contract No. 854184). Pro²Future is funded within the Austrian COMET Program (under the auspices of the Federal Ministry for Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK), the Federal Ministry for Digital and Economic Affairs (BMDW)) and of the Provinces of Upper Austria and Styria. COMET is managed by the Austrian Research Promotion Agency FFG. It has also received support by the European Union and the State of Upper Austria within the strategic program Innovative Upper Austria 2020, projects: “Smart Factory Lab” and “DigiManu”.

References

1. Weichhart, G., Feiner, T., Sary, C.: Implementing organisational interoperability – The SUddEN approach. *Comput. Ind.* 61, 152–160 (2010).
<https://doi.org/10.1016/j.compind.2009.10.011>.
2. Napoleone, A., Pozzetti, A., Macchi, M.: A framework to manage reconfigurability in manufacturing. *Int. J. Prod. Res.* 56, 3815–3837 (2018).
<https://doi.org/10.1080/00207543.2018.1437286>.
3. Kusiak, A.: Fundamentals of smart manufacturing: A multi-thread perspective. *Annu. Rev. Control.* 47, 214–220 (2019).
<https://doi.org/10.1016/j.arcontrol.2019.02.001>.
4. Mehandjiev, N., Grefen, P. eds: *Dynamic Business Process Formation for Instant*

- Virtual Enterprises. Springer-Verlag London (2010). <https://doi.org/10.1007/978-1-84882-691-5>.
5. Napoleone, A., Macchi, M., Pozzetti, A.: A review on the characteristics of cyber-physical systems for the future smart factories. *J. Manuf. Syst.* 54, 305–335 (2020). <https://doi.org/10.1016/j.jmsy.2020.01.007>.
 6. Norman, G.: Chaos, complexity and complicatedness: lessons from rocket science. *Med. Educ.* 45, 549–559 (2011). <https://doi.org/10.1111/j.1365-2923.2011.03945.x>.
 7. Holland, J.H.: *Hidden Order: How Adaptation Builds Complexity*. Basic Books, New York (1996).
 8. Bertalanffy, L. von: An Outline of General System Theory. *Br. J. Philos. Sci.* 1, 134–165 (1950).
 9. Bertalanffy, L. von: *General System Theory - Foundations, Development, Applications*. George Braziller, New York (1969).
 10. Boardman, J., Sauser, B.: System of Systems - the meaning of of. In: *System of Systems Engineering, 2006 IEEE/SMC International Conference on*. 1–6 (2006). <https://doi.org/10.1109/SYSOSE.2006.1652284>.
 11. Leitão, P., Vrba, P.: *Recent Developments and Future Trends of Industrial Agents*. Springer Berlin Heidelberg (2011).
 12. Wooldridge, M.J., Jennings, N.R.: *Intelligent Agents: Theory and Practice*. *Knowl. Eng. Rev.* 10, 115–152 (1995).
 13. Mitchell, M.: *Complexity: A guided tour*. Oxford University Press (2009).
 14. Waldrop, M.M.: *Complexity: the emerging science at the edge of order and chaos*. Simon & Schuster, New York (1992).
 15. European Commission: *New European Interoperability Framework*. (2017).
 16. Stamper, R., Liu, K., Hafkamp, M., Ades, Y.: Understanding the roles of signs and norms in organizations - a semiotic approach to information systems design. *Behav. Inf. Technol.* 19, 15–27 (2000). <https://doi.org/10.1080/014492900118768>.
 17. Anderson, P.: Complexity Theory and Organization Science. *Organ. Sci.* 10, 216–232 (1999).
 18. Weichhart, G.: Requirements for Supporting Enterprise Interoperability in Dynamic Environments. In: Mertins, K., Bénaben, F., Poler, R., and Bourrières, J.-P. (eds.) *Enterprise Interoperability VI*. 479–488 (2014). https://doi.org/10.1007/978-3-319-04948-9_40.
 19. Weichhart, G.: Supporting Interoperability for Chaotic and Complex Adaptive Enterprise Systems. In: Demey, Y.T. and Panetto, H. (eds.) *On the Move to Meaningful Internet Systems: OTM 2013 Workshops*. 86–92 (2013). https://doi.org/10.1007/978-3-642-41033-8_14.
 20. Bandura, A.: Social cognitive theory of self-regulation. *Organ. Behav. Hum. Decis. Process.* 50, 248–287 (1991).
 21. Weichhart, G., Stary, C., Vernadat, F.: Enterprise Modelling for Interoperable and Knowledge-based Enterprises. *Int. J. Prod. Res.* 56, 2818–2840 (2018). <https://doi.org/10.1080/00207543.2017.1406673>.
 22. Oppl, S., Stary, C.: Facilitating shared understanding of work situations using a tangible tabletop interface. *Behav. Inf. Technol.* 33, 619–635 (2014). <https://doi.org/10.1080/0144929X.2013.833293>.

23. Oppl, S.: Supporting the Collaborative Construction of a Shared Understanding About Work with a Guided Conceptual Modeling Technique. *Group Decis. Negot.* 26, 247–283 (2017). <https://doi.org/10.1007/s10726-016-9485-7>.
24. Weichhart, G.: Supporting the evolution and interoperability of organisational models with e-learning technologies. *Annu. Rev. Control.* 39, 118–127 (2015). <https://doi.org/10.1016/j.arcontrol.2015.03.011>.
25. Vernadat, F.: Enterprise modelling: Research review and outlook. *Comput. Ind.* 122, 103265 (2020). <https://doi.org/10.1016/j.compind.2020.103265>.
26. Zacharewicz, G., Daclin, N., Doumeingts, G., Haidar, H.: Model Driven Interoperability for System Engineering. *Modelling.* 1, 94–121 (2020). <https://doi.org/10.3390/modelling1020007>.
27. Kosanke, K., Vernadat, F., Zelm, M.: CIMOSA: enterprise engineering and integration. *Comput. Ind.* 40, 83–97 (1999). [https://doi.org/10.1016/S0166-3615\(99\)00016-0](https://doi.org/10.1016/S0166-3615(99)00016-0).
28. IDS Scheer AG: *Handbuch zur ARIS-Methode - Version 5*. Saarbrücken (2000).
29. Bernus, P., Noran, O., Molina, A.: Enterprise architecture: Twenty years of the GERAM framework. *Annu. Rev. Control.* 39, 83–93 (2015). <https://doi.org/10.1016/j.arcontrol.2015.03.008>.
30. Jonkers, H., Lankhorst, M.M., Quartel, D.A.C., Proper, E., Iacob, M.-E.: ArchiMate(R) for Integrated Modelling Throughout the Architecture Development and Implementation Cycle. In: *Commerce and Enterprise Computing (CEC), 2011 IEEE 13th Conference on*. 294–301 (2011). <https://doi.org/10.1109/CEC.2011.52>.
31. Vallespir, B., Merle, C., Doumeingts, G.: The GRAI Integrated Method: a Technico- Economical Methodology to Design Manufacturing Systems. *IFAC Proc. Vol.* 25, 73–78 (1992). [https://doi.org/10.1016/S1474-6670\(17\)52231-4](https://doi.org/10.1016/S1474-6670(17)52231-4).
32. Kosanke, K.: CIMOSA — Overview and status. *Comput. Ind.* 27, 101–109 (1995). [https://doi.org/10.1016/0166-3615\(95\)00016-9](https://doi.org/10.1016/0166-3615(95)00016-9).
33. Ducq, Y., Chen, D., Doumeingts, G.: A contribution of system theory to sustainable enterprise interoperability science base. *Comput. Ind.* 63, 844–857 (2012). <https://doi.org/10.1016/j.compind.2012.08.005>.
34. Dietz, J.L.G., Hoogervorst, J.A.P., Albani, A., Aveiro, D., Babkin, E., Barjis, J., Caetano, A., Huysmans, P., Iijima, J., van Kervel, S., Mulder, H., Op 't Land, M., Proper, H.A., Sanz, J., Terlouw, L., Tribolet, J., Verelst, J., Winter, R.: The discipline of enterprise engineering. *Int. J. Organ. Des. Eng.* 3, 86–114 (2013). <https://doi.org/10.1504/IJODE.2013.053669>.
35. Simon, H.A.: *The Sciences of the Artificial*. MIT Press, Cambridge, Massachusetts (1981).
36. Shaw, M.: The coming-of-age of software architecture research. In: *Proceedings of the 23rd International Conference on Software Engineering. ICSE 2001*. 656–664 (2001).
37. Iacob, M.E., Jonkers, H., Lankhorst, M., Proper, H., Quartel, D.: *ArchiMate 2.0 Specification*. The Open Group (2012).
38. The Open Group: *ArchiMate Specification*. (2019).
39. The Open Group: *TOGAF Standard*. (2018).
40. Chen, D., Vallespir, B., Doumeingts, G.: GRAI integrated methodology and its mapping onto generic enterprise reference architecture and methodology.

- Comput. Ind. 33, 387–394 (1997).
41. Chen, D., Doumeingts, G., Vallespir, B.: GIM (GRAI Integrated Methodology) and its Mapping onto GERAM. *IFAC Proc. Vol. 29*, 648–653 (1996). [https://doi.org/10.1016/S1474-6670\(17\)57734-4](https://doi.org/10.1016/S1474-6670(17)57734-4).
 42. ISO/TC 184/SC 5: ISO 15704:2019 Enterprise modelling and architecture - Requirements for enterprise-referencing architectures and methodologies. International Organization for Standardization (ISO) TC 184 / SC 5 Interoperability, integration, and architectures for enterprise systems and automation applications (2019).
 43. IFIP-IFAC Task Force: GERAM: Generalised Enterprise Reference Architecture and Methodology (GERAM EA Framework v.1.6.3), <http://www.ict.griffith.edu.au/~bernus/taskforce/index.html>, last accessed 2019/11/05.
 44. Object Management Group (OMG): Model Driven Architecture (MDA) Guide rev. 2.0. Object Management Group (2014).
 45. Bazoun, H., Zacharewicz, G., Ducq, Y., Boyé, H.: SLMTToolBox: An Implementation of MDSEA for Servitisation and Enterprise Interoperability. In: Mertins, K., Bénaben, F., Poler, R., and Bourrières, J.-P. (eds.) *Enterprise Interoperability VI*. 101–111 (2014).
 46. Weichhart, G., Molina, A., Chen, D., Whitman, L., Vernadat, F.: Challenges and Current Developments for Sensing, Smart and Sustainable Enterprise Systems. *Comput. Ind. 79*, 34–46 (2016). <https://doi.org/10.1016/j.compind.2015.07.002>.
 47. Morel, G., Panetto, H., Mayer, F., Auzelle, J.-P.: System of Enterprise-Systems Integration Issues: an Engineering Perspective. In: IFAC (ed.) *IFAC Conference on Cost Effective Automation in Networked Product Development and Manufacturing, IFAC-CEA'07*. Elsevier, Monterrey, Mexique (2007).
 48. Panetto, H., Jardim-Goncalves, R., Molina, A.: Enterprise Integration and Networking: Theory and practice. *Annu. Rev. Control. 36*, 284–290 (2012). <https://doi.org/10.1016/j.arcontrol.2012.09.009>.
 49. Naudet, Y., Latour, T., Guédria, W., Chen, D.: Towards a systemic formalisation of interoperability. *Comput. Ind. 61*, 176–185 (2010). <https://doi.org/10.1016/j.compind.2009.10.014>.
 50. Chen, D.: Framework for enterprise interoperability. In: *Congrès International de Génie Industriel (CIGI2009)* (2009).
 51. Berre, A.-J., Elvesæter, B., Figay, N., Guglielmina, C., Johnsen, S.G., Karlsen, D., Knothe, T., Lippe, S.: The ATHENA interoperability framework. In: *Enterprise interoperability II*. 569–580 (2007).
 52. Chen, D., Doumeingts, G., Vernadat, F.B.: Architectures for enterprise integration and interoperability: Past, present and future. *Comput. Ind. 59*, 647–659 (2008). <https://doi.org/10.1016/j.compind.2007.12.016>.
 53. Kouroubali, A., Katehakis, D.G.: The new European interoperability framework as a facilitator of digital transformation for citizen empowerment. *J. Biomed. Inform. 94*, 103166 (2019). <https://doi.org/10.1016/j.jbi.2019.103166>.
 54. Panetto, H., Iung, B., Ivanov, D., Weichhart, G., Wang, X.: Challenges for the Cyber-Physical Manufacturing Enterprises of the Future. *Annu. Rev. Control. 47*, 200–213 (2019). <https://doi.org/10.1016/j.arcontrol.2019.02.002>.
 55. Mahnke, W., Leitner, S.-H., Damm, M.: *OPC Unified Architecture*. Springer

- (2009). https://doi.org/10.1007/978-3-540-68899-0_1.
56. Pauker, F., Frühwirth, T., Kittl, B., Kastner, W.: A Systematic Approach to OPC UA Information Model Design. *Procedia CIRP*. 57, 321–326 (2016). <https://doi.org/10.1016/j.procir.2016.11.056>.
 57. Lüder, A., Hundt, L., Keibel, A.: Description of manufacturing processes using AutomationML. In: 2010 IEEE 15th Conference on Emerging Technologies & Factory Automation (ETF A 2010). 1–8 (2010).
 58. Schleipen, M., Gilani, S.-S., Bischoff, T., Pfrommer, J.: OPC UA & Industrie 4.0 - Enabling Technology with High Diversity and Variability. *Procedia CIRP*. 57, 315–320 (2016). <https://doi.org/10.1016/j.procir.2016.11.055>.
 59. Henssen, R., Schleipen, M.: Interoperability between OPC UA and AutomationML. *Procedia CIRP*. 25, 297–304 (2014).
 60. Weichhart, G., Guédria, W., Naudet, Y.: Supporting Interoperability in Complex Adaptive Enterprise Systems: A Domain Specific Language Approach. *Data Knowl. Eng.* 105, 90–106 (2016). <https://doi.org/10.1016/j.datak.2016.04.001>.
 61. Hewitt, C.: Viewing control structures as patterns of passing messages. *Artif. Intell.* 8, 323–364 (1977). [https://doi.org/10.1016/0004-3702\(77\)90033-9](https://doi.org/10.1016/0004-3702(77)90033-9).
 62. Agha, G.A.: Actors: A Model of Concurrent Computation in Distributed Systems, <http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA157917>, (June).
 63. Bonér, J., Klang, V., Kuhn, R., Nordwal, P., Antonsson, B., Varga, E.: Akka Scala Documentation. Typesafe Inc. (2014).
 64. Bénaben, F., Mu, W., Boissel-Dallier, N., Barthe-Delanoë, 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). <https://doi.org/10.1080/17517575.2014.928949>.
 65. Bénaben, F., Boissel-Dallier, N., Pingaud, H., Lorre, J.-P.: Semantic issues in model-driven management of information system interoperability. *Int. J. Comput. Integr. Manuf.* 13 (2012). <https://doi.org/10.1080/0951192X.2012.684712>.
 66. Agostinho, C., Jardim-Goncalves, R.: Sustaining interoperability of networked liquid-sensing enterprises: A complex systems perspective. *Annu. Rev. Control.* 39, 128–143 (2015). <https://doi.org/10.1016/j.arcontrol.2015.03.012>.
 67. Youssef, J., Zacharewicz, G.: Enterprise Operating System (EOS) in Action: Distributed Simulation of Enterprise Activities and Operations. In: 2019 Winter Simulation Conference (WSC). 90–104 (2019). <https://doi.org/10.1109/WSC40007.2019.9004958>.
 68. Youssef, J.R., Zacharewicz, G., Chen, D., Vernadat, F.: EOS: enterprise operating systems. *Int. J. Prod. Res.* 56, 2714–2732 (2017). <https://doi.org/10.1080/00207543.2017.1378957>.
 69. Ferreira, J., Lopes, F., Doumeings, G., Sousa, J., Mendonça, J.P., Agostinho, C., Jardim-Goncalves, R.: Empowering SMEs with Cyber-Physical Production Systems: From Modelling a Polishing Process of Cutlery Production to CPPS Experimentation. In: Jardim-Goncalves, R., Sgurev, V., Jotsov, V., and Kacprzyk, J. (eds.) *Intelligent Systems: Theory, Research and Innovation in*

- Applications. 139–177 (2020). https://doi.org/10.1007/978-3-030-38704-4_7.
70. Monostori, L.: Cyber-physical production systems: Roots, expectations and R&D challenges. *Procedia CIRP*. 17, 9–13 (2014). <https://doi.org/10.1016/j.procir.2014.03.115>.
 71. Monostori, L., Kádár, B., Bauernhansl, T., Kondoh, S., Kumara, S., Reinhart, G., Sauer, O., Schuh, G., Sihn, W., Ueda, K.: Cyber-physical systems in manufacturing. *CIRP Ann.* 65, 621–641 (2016). <https://doi.org/10.1016/j.cirp.2016.06.005>.
 72. Weichhart, G., Panetto, H., Molina, A.: Interoperability in the Cyber-Physical Manufacturing Enterprise. *Annual Reviews in Control.* (2021). <https://doi.org/10.1016/j.arcontrol.2021.03.006>.
 73. VDI/VDE-Gesellschaft Mess- und Automatisierungstechnik: Cyber-Physical Systems: Chancen und Nutzen aus Sicht der Automation. VDI/VDE-Gesellschaft Mess- und Automatisierungstechnik (2013).
 74. Weichhart, G., Fast-Berglund, Å., Romero, D., Pichler, A.: An Agent- and Role-based Planning Approach for Flexible Automation of Advanced Production Systems. In: 2018 International Conference on Intelligent Systems (IS), 391–399 (2018). <https://doi.org/10.1109/IS.2018.8710546>
 75. Weichhart, G., Reiser, M., & Sary, C.: Task-Based Design of Cyber-Physical Systems—Meeting Representational Requirements with S-BPM. In International Conference on Subject-Oriented Business Process Management. 63-73 (2020).
 76. Weichhart, G., Panetto, H., Guédria, W., Bhullard, G. and Moalla, N.: Pathways to CP (P) S modelling & architecting. In 10th International Conference on Interoperability for Enterprise Systems and Applications (I-ESA 2020). (2020)