Integrated design and operation management for enterprise systems

W.J. Zhang, J.W. Wang & Yingzi Lin

To cite this article: W.J. Zhang, J.W. Wang & Yingzi Lin (2019) Integrated design and operation management for enterprise systems, Enterprise Information Systems, 13:4, 424-429, DOI: 10.1080/17517575.2019.1597169

To link to this article: https://doi.org/10.1080/17517575.2019.1597169

Published online: 30 Mar 2019.

Submit your article to this journal

Article views: 449

View related articles

View Crossmark data

Citing articles: 2 View citing articles
Integrated design and operation management for enterprise systems

Enterprise is a mini social-technical-ecological system in that it consists of humans, equipment and machines, and it has a location or site. Its structure follows the substance-infrastructure (S-I) framework (Zhang and Wang 2016; Zhang and van Luttervelt 2011). There are two types of the S-I framework: the substance drives the infrastructure (Type I) and the infrastructure drives the substance (Type II). The enterprise system belongs to Type II. For instance, to a manufacturing system, the substance refers to goods made of materials, and the infrastructure refers to humans and machines, which produce and deliver goods to customers in response to their demands. To a service system (Wang et al. 2014), the substance refers to data (knowledge and information) (Zhang 1994) or signals or humans, and the infrastructure refers to humans and machines, which generate data, produce signals, or offer services to customers in accordance with their demands.

Enterprise is a dynamic system, and it changes in its state and/or structure with respect to time, location, and/or event, and both the substance and infrastructure may change. A change on the part of the structure and/or state, say A, always has a reason or reasons, and this change is further associated with the change of another part of the structure and/or state of the system, say B; B is an independent variable and A is a dependent variable in this case (Zhang et al. 2005). The knowledge that governs the relation of A and B is called principle (Zhang et al. 2005; Zhang and Wang 2016). For instance, B is the force \( F \) applied on a block and A is the acceleration \( a \) of the block system, and the knowledge that governs the relation of A and B, in this case, is the Newton’s second law, that is, \( F = ma \), where \( m \) is the mass of the block system. A care must be taken that the principle (knowledge) may be hidden or unfolded in data or big data but a correspondence relation of A and B can be built using various machine learning methods, e.g., Artificial Neural Network (ANN) (Zhao and Zhang 2017), various deep learning methods (Zhang et al. 2018), etc. The independent variable is a function of time, location, and/or event, so is the dependable variable, and thus the whole system changes with respect to time, location, and/or event.

Design of an enterprise system means to determine its structure (infrastructure and substance) in response to a need or demand in a context (Zhang and Wang 2016). For instance, in response to the need of charging to electric vehicles, a new enterprise idea, the electric charge station enterprise, emerges. To this new enterprise, one needs to determine the charging equipment, number of workers, and so on, which makes sense to the design of an enterprise (Zhang and Wang 2016). Construction of an enterprise follows its design. Design and construction are processes, so it makes sense to say about their management. A good practice of the management of design and construction thus results in a good structure.
After an enterprise is designed and constructed (i.e., its structure is available), the operation of the enterprise takes shape. The operation includes: planning, scheduling, and executing. It makes sense to say about their management called operations management. A good practice of the management of these operations, together with the structure of the enterprise system, eventually results in a cost-effective performance (i.e., supply meeting demand) along its behavioural properties (i.e., stability, reliability, robustness, resilience, and sustainability) of the enterprise system. It is to be noted that to a robotic system, the operation activities (i.e., planning, scheduling and executing) are crashed into a compact one called ‘actuation’, and management changes to control accordingly.

Figure 1 shows the relationship among design, operation, structure (including customer and technical specification), and management. By generalisation, management has a target to be managed, such as design and operation activities. Management is to control the running of the activities to achieve a certain goal under a certain constraint (Figure 1). The goal of design management is to generate a specific structure of a system along with customer voice of needs and technical specification (Figure 1), and the goal of operation management is to make the structure to a specific performance along with a set of specific properties (e.g., stability) that the structure is expected to exhibit (Figure 1). Due to the presence of imprecise structure and data (Cai et al. 2017), the feedback approach to management is widely employed (Figure 1; Loop 1–Loop 4). The feedback management may also take place on Loop 5 (Figure 1), which involves design activities and operation activities and their managers. In feedback management, management is a decision-making process, so it is a system called management system or manager.

The manager is a cognitive system rather than a physical system. As such, the structure of a management system is essentially a symbolic expression, e.g., \( y = f(x) \), where \( f(.) \) is a mapping, and \( x \) and \( y \) represent objects in the domain of concepts which are structure, state, principle, behaviour, context, and function according to the FCBPSS architecture (Zhang et al. 2005; Zhang and Wang 2016). A different \( f \) means a different management system or manager. A manager could be a human being or computer (or machine). To a human manager, \( f \) is hardwired in the human brain, and to a computer manager, \( f \) is coded. In a particular management process, \( f(.) \) may not be unchanged. A manager (design manager, operation manager) that its \( f(.) \) may change during the management
process is called adaptive manager, and the change is taken by meta manager (meta-
design manager, meta operation manager), see Figure 1. The meta-design manager (meta
operation manager) is mapping as well, denoted by $g(.)$, and $\delta f(.) = g(z)$, where
$z$ represents the structure (performance), and $\delta f(.)$ represents the modification $f(.)$.

It is noted that when the design manager ($f_d$) changes, the structure of the system
(e.g., enterprise) under design and construction changes. If a structure has already been
built, the change of the structure requires that the structure is designed as a changeable
or adaptable structure. Further, the implementation of the change may be performed by
the structure itself (self-change) or by the external entity.

Design management and operation management (design and operation in short) in
the above must be integrated because both refer to the same entity (the general reason
for integration). This integration theory for systems may be called IDOM (Integrated
Design and Operation Management), which is an extension of the integrated design and
control for robotic systems (Li, Zhang, and Chen 2001, Cheng et al. 2012). In the
following, several axioms for IDOM are proposed by generalising the existing results
from integrated design and control of robots.

First, a robot may be modularised, and a module may include several parameters
which further correspond to multiple performance indexes. As such, by selection of one
module, all the parameters of this module are selected, which means that all the
performances related to these parameters must be considered together (Bi and Zhang
2001), which gives a sense of integration. By a generalisation of this scenario, an
integration axiom (IDOM Axiom I) can be concluded as follows:

**IDOM Axiom I:** When a module or a set contains a number of parameters ($p_i$, $i = 1,$
2, $\ldots$, $n$), which correspond to a number of performances or properties ($pp_j$, $j = 1,$
2, $\ldots$, $m$), the simultaneous determination of $p_i$ with the consideration of all the $pp_j$ under
the constraint that the modules are available, say $m_k$ ($k = 1, 2, \ldots, w$), and each module is
characterized by $p_i$ is of necessity. This integration problem can be mathematically repre-
sented as a constrained multi-objective optimization problem with discrete variables.

Second, the simultaneous determination of the parameters, particularly one group
that describes the structure and the other group that describes the operation, may lead
to a better performance than the sequential determination of the parameters in robotics
(Zhang and Chen 2000; Cheng et al. 2012). By a generalisation of this scenario, one can
lead to another integration axiom (IDOM Axiom II) as follows:

**IDOM Axiom II:** If a set of parameters across two or more phases (e.g., the parameter
set A in phase I, and the parameter set B in phase II) are coupled to one or more
performances, the simultaneous determination of these parameters for the optimal perfor-
mances may lead to improved performances as opposed to the sequential determination of
the parameters based on the precedence of the phases (i.e., the phase I precedes the phase
II; A in the phase I is determined first, followed by B in the phase II).

Finally, in robotics, a well-known approach to the integration of design and control is
to design the structure of a robot such that some properties of the structure can be
obtained, e.g., cancellation of gravitational force of the robot, which reduces the
‘burden’ of the controller and subsequently improves the overall performance of the
robot further. This approach in robotics is called DFC (Design For Control) (Zhang, Li,
and Guo 1999). The philosophy behind DFC can be explored. One can view control as an
intervention to a system. To each burden, a particular control is designed. Suppose that
the system has several burdens say A, B, C. The control for A may have bad side effects on the control for B and C (respectively), and the control for B may have bad side effects on the control for A and C. So if the structure can be designed to remove some burdens to the control, which will increase the chance of minimising bad side effects across the different controls and thus improve the overall performance of the robot. By a generalisation of this scenario, one can lead to another axiom for IDOM (IDOM Axiom III) as follows:

**IDOM Axiom III**: Suppose the parameter set A is in the phase I and the parameter set B is in the phase II and the phase I precedes the phase II. Identify the potential burdens to the phase II, and perform the activities in the phase I such that some of the burdens can be removed.

Several remarks can be made regarding the above IDOM approach. **Remark 1**: Any integration scheme that goes beyond the above axioms may actually get some bad results, as the integration can inherently create unnecessary constraints, which may lose some promising solutions called bad integration. **Remark 2**: In case that the phase I is design and the phase II is operation, IDOM can only be performed based on the simulation unless the structure can be changeable or adaptable, simulation of the activities on the phase II in particular. **Remark 3**: The three axioms of IDOM are applicable to activities in one phase (e.g., the planning and scheduling activities in the operation phase) or to activities across different phases (e.g., the structure in the design phase and the plan and schedule in the operation phase). **Remark 4**: An enterprise could be a single organisation or a group of organisations, especially a chain of organisations (Figure 2). In the latter case, the IDOM approach is applicable to the group with (1) integration on activities in one phase (Integration 3 in Figure 2), (2) integration on activities across different phases (Integration 2 in Figure 2), and (3) integration on activities across different organisations, e.g., Structure 1 and Structure 2 (Integration 1 in Figure 2). **Remark 5**: The integration problem can be formulated as an optimisation problem model, and it is a non-trivial issue to solve such a model effectively and efficiently. **Remark 6**: Integration also makes sense to integrated design of substance (products) and of infrastructure (manufacturing systems), a specialisation of which is the so-called design for manufacturing approach, as well as integrated design of operation and of operator or manager.

This special issue collected seven papers on the theme of integration in various ways yet under the framework as illustrated in Figures 1 and 2. **Paper 1** entitled ‘A novel resilient scheduling paradigm integrating operation and design for manufacturing systems with uncertainties’ proposes an approach to integrated system design and operation scheduling for the task-specific production performance along with the resilience property. The

**Figure 2**: Integration schemes on two organisations. D: Design, C: Construct, O: Operation, P: performance or property, CTS: Customer-Technical specification-Structure.
integration scheme of the paper hits Integration 2 in Figure 2. The problem is further solved by a method that incorporates big data analytics into a multi-stage optimisation method. **Paper 2** entitled ‘An integrated approach for dynamic customer requirement capturing for product development’ proposes an integrated approach to capture the customer’s voice of needs and technical specification, which are two activities in the design phase for products or enterprises that make the products. The integration scheme of the paper hits Integration 4 in Figure 2. **Paper 3** entitled ‘Aspect-oriented challenges in system integration with microservices, SOA and IoT’ discusses tools with which integration can be enabled, AOP (Aspect-Oriented Programming) in particular. A promising integration can only be realised upon an accurate understanding of each system and their interactions, and not all integrations are good according to Remark 1 in the above discussion. The contribution of this paper is the provision of information about AOP and its role in integration. **Paper 4** entitled ‘Integrated scheduling of production and distribution operations in a global MTO supply chain’ discusses the problem of the integration of production scheduling of products and their distribution in a global MTO supply chain with two objectives: cost and delivery time. The integration scheme of the paper hits Integration 3 in Figure 2. The contribution of the paper is a new formulation of the problem and a new solution to it. **Paper 5** entitled ‘Patient assignment scheduling in a cloud healthcare system based on Petri Net and greedy-based heuristic algorithm’ proposes an approach to integrated resource planning and patient assignment scheduling for a group of hospitals, which share the same resource. The integration scheme of the paper hits Integration 5 in Figure 2. **Paper 6** entitled ‘The modelling and operations for the digital twin in the context of manufacturing’ develops a platform for digital modelling and simulation of products, manufacturing or construction processes, operations or actions. The platform is a tool for integrated design and construction or building or manufacturing of products, i.e., Integration 4 in Figure 2. **Paper 7** entitled ‘Integrated scheduling for a distributed manufacturing system: a stochastic multi-objective model’ develops a model for integrated scheduling among a group of firms with consideration of uncertainties. The integration of the paper hits Integration 5 in Figure 2.

The IDOM is still under development, and its potential to enterprise system design and operation management is still being explored. The main issues for future IDOM are: (1) how to efficiently solve the IDOM problem model, as it is usually a multi-objective optimisation model and the computational challenge is high, (2) how to identify any unnecessary integration, as integration of two unrelated activities is creating unnecessary constraints that degrade their performance, (3) how to optimise the integration scheme for complex network and distributed systems such as holistic supply chain network systems to improve their task performance as well as their robustness and resilience (Wang et al. 2016, 2017; Said, Bouloiz, and Gallab 2019).

---

**References**


