HAL
open science

Collaborative Knowledge Framework for Mediation Information System Engineering

Wenxin Mu, Frederick Benaben, Nicolas Boissel-Dallier, Hervé Pingaud

To cite this version:

Wenxin Mu, Frederick Benaben, Nicolas Boissel-Dallier, Hervé Pingaud. Collaborative Knowledge Framework for Mediation Information System Engineering. Scientific Programming, 2017, 2017 (Article ID 9026387), p.1-18. 10.1155/2017/9026387 . hal-01656514

HAL Id: hal-01656514

https://hal.science/hal-01656514v1

Submitted on 6 Nov 2018

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.

Hindawi
Scientific Programming
Volume 2017, Article ID 9026387, 18 pages
https://doi.org/10.1155/2017/9026387

Research Article

Collaborative Knowledge Framework for Mediation Information System Engineering

Wenxin Mu,[1] Frederick Benaben,[2] Nicolas Boissel-Dallier,[3] and Herve Pingaud[4]

[1] *School of Economics and Management, Beijing Jiaotong University, Shangyuancun 3, Xizhimenwai, Haidian, Beijing 100044, China*
[2] *Toulouse University, Mines D'Albi, Campus Jarlard, Route de Teillet, 81013 Albi, France*
[3] *InteropSys, 23 Bd. Victor Hugo, 31770 Colomiers, France*
[4] *Champollion University, Place Verdun, 81012 Albi, France*

Correspondence should be addressed to Wenxin Mu; wxmu@bjtu.edu.cn

Received 3 March 2017; Revised 7 May 2017; Accepted 30 August 2017; Published 23 October 2017

Academic Editor: Kwok-Yan Lam

Copyright © 2017 Wenxin Mu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the worldwide interenterprise collaboration and interoperability background, automatic collaborative business process deduction is crucial and imperative researching subject. A methodology of deducing collaborative process is designed by collecting collaborative knowledge. Due to the complexity of deduction methodology, a collaborative knowledge framework is defined to organize abstract and concrete collaborative information. The collaborative knowledge framework contains three dimensions: elements, levels, and life cycle. To better define the framework, the relations in each dimension are explained in detail. They are (i) relations among elements, which organize the gathering orders and methods of different collaborative elements, (ii) relations among life cycle, which present modeling processes and agility management, and (iii) relations among levels, which define relationships among different levels of collaborative processes: strategy, operation, and support. This paper aims to explain the collaborative knowledge framework and the relations inside.

1. Introduction

The INTEROP-NOE proposes an enterprise interoperability framework, which defined three interoperability barriers: conceptual, technological, and organizational [1] (INTEROP-VLab, the European Virtual Laboratory for Enterprise Interoperability (I-VLab), is an initiative to develop networked research with critical mass in the Enterprise Interoperability (EI) domain and associated domains (Future Internet and Enterprise Systems Applications). Created on March 2007 as an AISBL (Association Internationale Sans But Lucratif) under Belgian law, I-VLab coordinates now more 50 institutions from 11 countries (including China) and over 200 researchers. http://www.interop-vlab.eu/). Considering that enterprises' information systems are the practical and operational part in enterprise, it is a crucial requirement to break technological barriers among enterprises' information systems. There exists possibility of jumping organizational and conceptual obstacles by crushing technological stumbling block (Figure 1). To search for a method to break the technological barriers among enterprises' information systems, various architectures for the interoperation of information systems are introduced and compared in [2]. They include Peer-to-Peer [3], Standardization (Standardization uses pivot, canonical model, or metamodel to reduce the number of translators (similar to Peer-to-Peer)), Federation (Federation derives from standardization and uses a global, static federated schema), Multibase (Multibase uses a single language for many ISs), Ontology [4], and Mediation [5]. Considering the weak point of adding a new partner (and its IS (Information System)) which requires many translators for each existing partner, Peer-to-Peer and Standardization are eliminated. Considering the difficulty of building common standard and language, Federation and Multibase are removed. Although mediation information system (MIS) *requires the difficult task of constructing automatically collaborative process*, MIS still is a credible and pertinent way of supporting ISs interoperability. The concept of mediation was first presented in this MIS should be able to deal with (Figure 1):

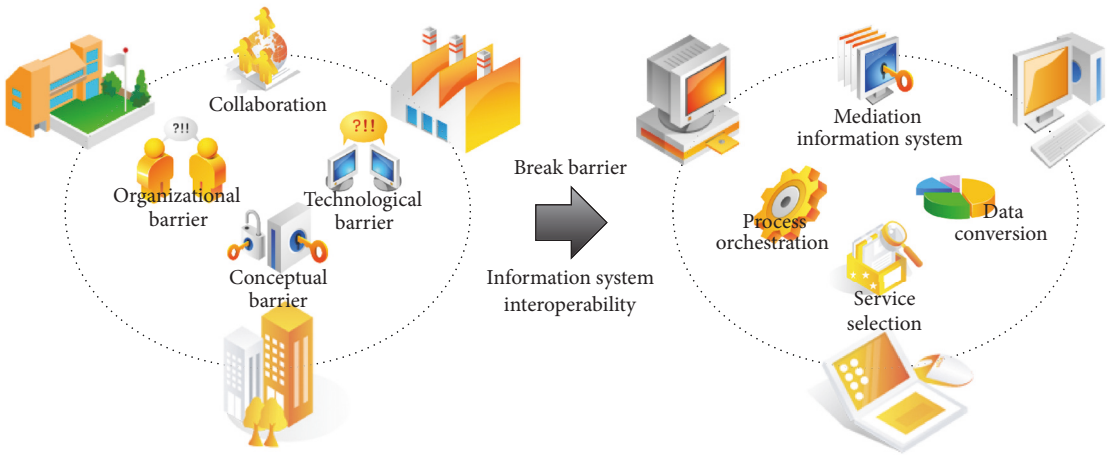


Figure 1: Framework of collaborative situation.

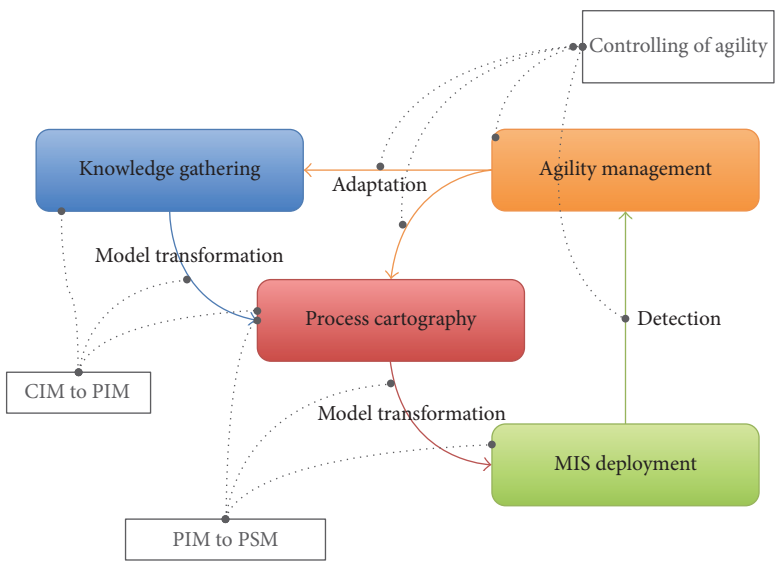


Figure 2: Global picture of MISE 2.0.

(i) Service selection: MIS selects services which are provided by enterprise to achieve the collaborative goal.

(ii) Data conversion: MIS transfers data among different enterprises to share business or technical information.

(iii) Process orchestration: MIS orchestrates the services shared by enterprises in process to transfer shared data and complete collaboration.

Since 2009, MISE 2.0 (Mediation Information System Engineering Version 2.0) project has been launched. The project is Model-Driven Architecture (MDA) [6] and Service Oriented Architecture (SOA) [7] based. MDA is divided into several models. According to [8], the models are Computation Independent Model (CIM), Platform Independent Model (PIM), and Platform Specific Model (PSM). Based on MDA, the global picture of MISE 2.0 is divided into several parts (Figure 2): abstract level, concrete level, and agility management.

Abstract level concerns the transformation from CIM to PIM in the design-time MIS. This part of work is presented in [9]. It gathers basis collaborative knowledge gathering and deduces the cartography of business collaborative process. In this level, the collaborative knowledge (e.g., partner information, collaborative objectives, collaborative network, and partners' functions) is collected and presented by organizational, functional, and informational models. And then this knowledge is transferred into the cartography of collaborative process thanks to collaborative metamodel and model transformation rules.

Concrete level concerns the transformation from PIM to PSM in the design time of MIS. This part of work is presented in [22]. It reuses the cartography of collaborative process and transfers it to the technical collaborative workflow. The workflow is deployed on ESB [23] and developed as MIS.

Agility management concerns the detection of errors in the runtime of MIS and the adaption of CIM, PIM, and PSM to solve the problems. This part of work is presented in [24].

Due to complexity of MISE 2.0 engineering, the collaborative knowledge framework is defined to classify collaborative knowledge in abstract and concrete level. However, abstract level engineering still contains models, metamodel, and model transformation mechanism. They are correlated and interacted with orders and individual processes. In order to clearly explain the engineering approach of MISE 2.0 and to help researchers and enterprises to reuse MISE 2.0 engineering approach, the collaborative knowledge framework is built. In the collaborative knowledge framework, the relations of elements inside the framework are presented. These relations contain (i) relations among elements, presenting model and metamodels used to present or organize collaborative knowledge which is defined in collaborative knowledge framework, (ii) relations among life cycle, defining models' use orders and models' transformation processes, and (iii) relations among levels, providing messages transferred among subcollaborative processes in collaborative business process model.

In this paper, Section 2 first addresses existing enterprise architectures and interoperability frameworks. Secondly, it presents collaborative situation framework. Section 3 provides explanation of abstract level frameworks: relations of collaborative elements, relations of collaborative life cycle, and relations of collaborative levels. Section 4 provides a case study. This case study connects abstract level and concrete level and shows the using of collaborative knowledge framework. Section 5 gives the evaluation of the collaborative knowledge framework. Section 6 draws some concluding remarks, discusses the feasibility of our work, and outlines our future investigations.

2. Collaborative Situation Framework Proposal

2.1. States of Art. In general, framework is a real or conceptual structure intended to serve as a support or guide for the building for something that expands the structure into something useful. A framework is a hypothetical description of a complex entity or process. According to Camarinha-Matos and Afsarmanesh, "in the modeling area, a framework can be seen as an "envelope" that might include a number of (partial) models, collections of templates, procedures and methods, rules, and even tools (e.g., modeling languages)" [25]. Modeling framework provides a set of viewpoints of subject, correlated, organized, and interacted. An enterprise architecture framework is a framework for an enterprise architecture, which defines how to organize the structure and views associated with enterprise architecture.

Enterprise architecture started with the Zachman Framework [26] in 1987. Another early implementation of an enterprise architecture framework was the "Technical Architecture Framework for Information Management" (TAFIM) [27]. During 80s, the most known enterprise architectures are as follows: the Purdue Enterprise-Reference model: PERA (1991), Architecture of Integrated Information Systems: ARIS (1991), the Computer Integrated Manufacturing Open System Architecture: CIMOSA (1993), the Open Group Architecture Framework: TOGAF (1995), the GIM architecture (1996), Enterprise-Reference Architecture and Methodology: GERAM (1997), and Federal Enterprise Architecture Framework: FEAF (1999). In 2003, DODAF (Department of Defense Architecture Framework) [28] was developed by the US Department of Defense. DODAF is an evolutionary upgrade of the C4ISR architecture framework [29]. In 2005, The British Ministry of Defense Architectural Framework (MODAF) v1.0 [30] was developed by MOD from DODAF version 1.0.

The Zachman Framework [31] is used for enterprise engineering and manufacturing. It provides the views from different members (e.g., planner, owner, and designer) and knowledge (data, function, and network). Zachman framework almost covers all the knowledge of enterprise. But the main problem is that there is no related methodology followed to use the framework and potential connections among views are ignored.

The GIM (GRAI Integrated Methodology) architecture [32] is a modeling methodology intended for general description, focused on details in manufacturing control system. This framework has four main parts: functional model, informational model, decision-making model, and physical model. The strong point of this framework is the decision-making model: GRAI [33], which allows user to model all the decision units and activities by time and organization. The weak point is that the framework considers information system as an important part in enterprise without defining detailed connections with business part.

PERA [34] considers that enterprise has three main components: facility, organization, and information system. It manages these three components by different phases in life cycle.

Architecture of Integrated Information Systems: ARIS [35] is architecture for information system integration, which is the most contiguous with the objective of MISE 2.0. ARIS manages integration by views: data, process, function, organization, and control. The connections among views are considered also. It also provides modeling tools and software platform to support model building and process transformation from business to technical. But user must build the business process. The regretful point is that MDA model transformation phase is not shown in the framework.

The Computer Integrated Manufacturing Open System Architecture: CIMOSA [36] is a three-dimension cube (generation of views, instantiation of building blocks, and derivation of models). The clear structure makes the framework easy to understand. Each dimension provides an angle for starting enterprise modeling. CIMOSA provides a process model without software supporting. In CIMOSA, it combines function and process in the function view. With the objective of computer integration, it limits the further development of web services.

The Open Group Architecture Framework: TOGAF [28] is an industry standard architecture framework that may be used freely by any organization wishing to develop information systems architecture for use within that organization. It

has been developed and continuously evolved. In TOGAF v9 (http://www.opengroup.org/togaf/) (2009), TOGAF Architecture Development Method (ADM) is provided. It is used to manage the use process of all the subarchitectures (ADM Guidelines and Technique, TOGAF Architecture Content Framework, Enterprise Continuum, TOGAF Reference Models and TOGAF Capability Framework) in TOGAF. According to TOGAF v9 survey results [37], more than 50% organizations are using TOGAF 8 and 9 to manage their enterprises (TOGAF 8: 30%, TOGAF 9: 21%, Zachman: 24%, FEAF: 7%, DODAF: 7% and MODAF 2%). The scope of the four architecture domains (data architecture (what?), business architecture (how?), technical architecture (where?), and applications architecture (who?)) of TOGAF aligns with the first four rows (what, how, where, and who but without when and why) of the Zachman Framework.

Enterprise-Reference Architecture and Methodology: GERAM [38] provides a generalized framework for describing the components needed in all types of enterprise engineering/enterprise integration processes. The shining point of this framework is life cycle. GERA (generalized enterprise-reference architecture) classified generic concepts as human oriented, process oriented, and technology oriented. The shining point is process oriented which defines enterprise life cycle into enterprise engineering, reengineering, and redesign. And then, the framework even details views (model content, purpose, implementation, and manifestation) and objects (customer service, software, hardware, information, function, machine, human, etc.) on the life cycle. The GERAM framework defines the minimal set of elements, which should be accompanied with, to build enterprise architectures. But these elements are abstract, for example, enterprise engineering methodologies, modeling languages, and modeling methodology. Users have to develop their own specific methodology or choose a developed methodology.

Federal Enterprise Architecture Framework: FEAF [39] provides a common methodology for information technology acquisition, use, and disposal in the Federal government. It provides performance reference model, business reference model, service component reference model, data reference model, and technical reference model.

Except enterprise architectures, in recent years, the enterprise interoperability framework is developed. IDEAS: Interoperability Development for Enterprise Application and Software (2003) [40] manages the enterprise integration from business, knowledge, and application level by solving semantic problems. AIF: ATHENA interoperability framework (2007) [41] improves IDEAS. It defines enterprise interoperability with four levels: business, service, process, and data. The solutions of each level can be ontology, semantics, and model-driven interoperability. EIF differs from IDEAS and AIF. IDEAS and AIF are two-dimension frameworks. EIF: Enterprise Interoperability Framework (2008) [1] is three-dimension framework. It defines enterprise interoperability from three angles: interoperability approaches, interoperability barriers, and interoperability concerns.

The architectures mentioned above have been summarized in Table 1.

2.2. MISE 2.0 Collaborative Situation Framework. The collaborative situation framework should also cover all the collaborative knowledge and direct collaborative situation modeling and helps mediation information system generation. In MISE 2.0, a collaborative framework should define viewpoints by organization, function, information, process, and interconnections among them. Furthermore, the engineering approach of MISE 2.0 goes through all the steps of MDA. So in our framework, two dimensions with viewpoints and MDA are confirmed.

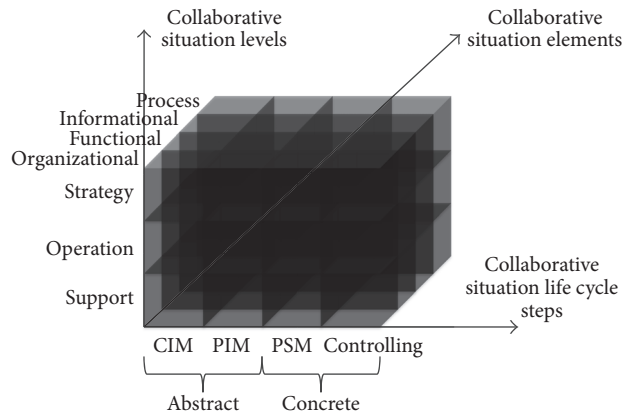


Figure 3: Framework of collaborative situation.

However, almost all the frameworks mentioned in Section 2.1 have a module or a unit for strategy management or decision-making, which is not shown in the main framework. Furthermore, according to ISO 9000 [42, 43], a business process should contain strategy process, operation process, and support process. With our experience on MISE 1.0 deployment, one collaborative process is not good enough to manage collaborative situation. It is very hard to understand for different levels' managers and workers. So we break the two dimensions framework into 3 levels. As shown in Figure 3, it is MISE 2.0 collaborative knowledge framework.

MISE 2.0 collaborative situation framework has three dimensions:

(i) Collaborative situation life cycle steps separate collaboration situation knowledge by mediation information system building steps. The collaboration situation life cycle covers CIM, PIM, PSM, and controlling. The CIM and PIM are at the abstract level. In the abstract level, business information and collaboration requirement have to be gathered. With this information, the business collaborative process may be deduced. Then, at the concrete level, the problem of semantic web service may be additional at PSM and controlling stages. In this part, collaboration process and semantic information are used to build target mediation information system.

(ii) Collaborative situation levels separate collaboration situation knowledge by different collaboration management levels. The dimension provides not only the operation level but also the strategy and support level. The strategy level helps decision-making, collaboration direction choosing and management level communicating. The operation level provides detailed

Table 1: Summary of enterprise architectures.

Framework	Type	Update	Completeness	Practicability	Tool support	Method support	Summary
Zachman (1987)	EA	2011 V3	Middle	Low	No	No	A general framework without specification
GIM (1988)	EA-PM	No	Middle	Middle	No	Yes	Develop a useful decision making method, GRAI
PERA (1991)	EA	No	Middle	Middle	No	No	Consider enterprise as facility, organization, and information system
ARIS (1991)	EIA-IS	2011 V7.2	Middle	High	Yes	Yes	MDA and SOA based; consider enterprise integration views as organization, data, function process, and control
CIMOSA (1993)	EIA	No	Middle	Middle	No	Yes	Clear structured three-dimension framework which are views, models, and levels
TOGAF (1995)	EA-IS	2009 V9	High	High	No	Yes	Architecture of architectures, each part is well detailed
GERA (1997)	EA	No	High	Low	No	Yes	Enterprise engineering life cycle is well detailed
FEA (1999)	EA-IS	2012 V3	High	High	No	Yes	Organize enterprise by levels: business, design, application, and technology; containing as-is and to-be system modeling
DoDAF (2003)	EA-IS	2009 V2.0	High	Middle	No	Yes	Separates enterprise by viewpoints: all, data and information, standards, capability, operational, services, systems, and project
IDEAS (2003)	EF	No	Low	Low	No	No	Defines interoperability levels: business, knowledge, and application. The solution is semantics
AIF (2007)	EF	No	Middle	Middle	No	No	Defines interoperability levels: business, service, process, and data. The solutions are ontology, semantics, and model-driven interoperability
EIF (2008)	EF	No	High	Middle	No	No	Defines interoperability from interoperability concerns, interoperability barriers, and interoperability approaches

EA: enterprise architecture; EIA: enterprise integration architecture; EF: enterprise interoperability framework; IS: information system; PM: product manufacturing.

collaboration solutions and execution results. The support level complements need and functions for operation level and strategy level.

(iii) Collaborative situation elements separate the collaboration situation knowledge by different knowledge viewpoints. It covers the organizational view, the informational view, the process view, and the functional view. The organizational view concerns collaboration network, partners, and collaborative objective. The informational view provides basic business data. Process view provides collaboration process. The functional view provides the capabilities of each partner.

The goal of collaborative situation framework is to transfer organizational, functional, and informational elements of CIM level to process element (which presents the process as strategy, operation, and support process) in PIM level.

3. Abstract Level Architecture Specification

Collaborative situation framework clearly shows the knowledge, which should be gathered in abstract level. But this framework could not answer which knowledge should be gathered first? What are the connections among the knowledge? What kinds of models, tools, or languages could be used to present the knowledge to user?

MISE 2.0 also proposes the following detailed architectures:

(i) Relations among elements provide models which are used to present or organize abstract level knowledge and basis connections.

(ii) Relations among life cycle provide how are these models played, transferred, and built to deduce collaborative process.

(iii) Relations among levels define three kinds of messages, which are used to trigger another process.

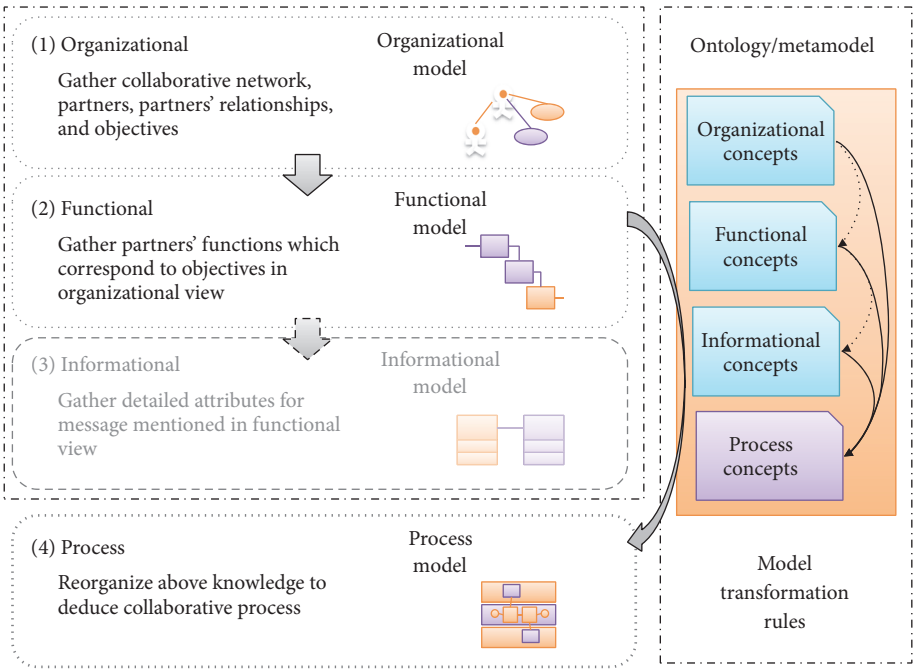


Figure 4: Relations among collaborative situation elements.

3.1. Relations among Elements. Relations of collaborative elements (RCE) are based on Collaborative Knowledge Framework Abstract level. The RCE aims at (i) defining modeling methods or modeling languages to gather or organize the collaborative knowledge at abstract level, which is defined in collaborative situation framework and (ii) providing the gathering orders of collaborative elements at the abstract level. As shown in Figure 4, the RCE has two parts: (i) organizational, functional, informational, and process and (ii) models and metamodel (and ontology). In MISE 2.0, there exists an order, which should be followed when the collaborative elements are gathered.

When the collaboration starts, the first thing to know is the following: what are the objectives? And who are the partners? The *organizational elements* should be gathered first. For these elements, collaborative network, partners, partners' relationships, and objectives of network and partners are gathered. All the knowledge of organizational element is the initial knowledge for a collaborative situation. In order to gather the organizational elements, an organizational model is necessary to gather and present the organizational knowledge.

In organizational elements, the objectives and the partners of the collaboration are provided. Then the next thing to know is the following: if the partners are willing to involve in the collaboration, what are the functions of partners? Which functions could be used to achieve the identified objectives? So, the *functional elements* should be gathered second. For this element, partners' functions have been gathered. In order to fix this requirement, a functional model is required to gather partners' functions.

Even though normally a functional model does not just gather functional information, it also covers input/output messages, which are exchanged among functions. In some case, the input/output messages of functional model do not contain enough informational knowledge. An informational model may be necessary to gather additional informational knowledge to complete the collaborative knowledge. The additional knowledge of informational knowledge may provide the attributes of messages, the relations among messages, and semantic annotation. The third elements to gather are the *informational elements*.

Finally, all the information, which has been gathered by the above three types of elements, is reused, reorganized, and represented to deduce a collaborative process model. This collaborative process model is based on BPMN. This BPMN based collaborative process model is specialized to one mediation pool (containing three collaborative lanes: strategy process, operation process, and support process) and several partners' pools. In order to transfer organizational, functional, and informational elements as process element, the definitions of models cannot accomplish the transformation. The modeling elements of organizational, functional, informational, and process models should be managed and confirmed by metamodel or ontology. Based on ontology and metamodel, transformation rules could be defined to transfer organizational, functional, and informational models to process model. In MISE 2.0, the collaborative ontology and the model transformation rules are defined to complete this mission.

3.2. Relations among Life Cycles. In the collaborative situation framework, the collaborative situation life cycle contains CIM, PIM, PSM, and controlling. In the collaborative situation framework, reader could understand them as follows:

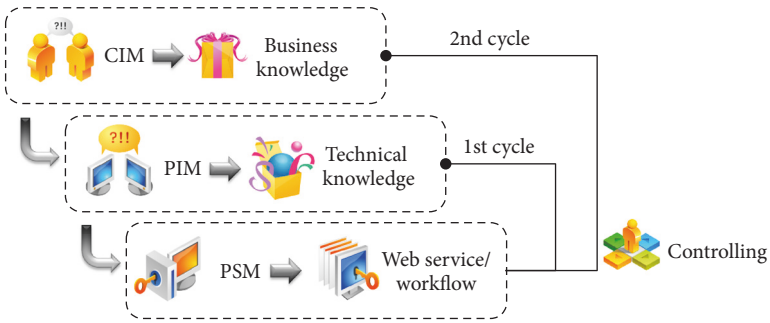


Figure 5: Relations in collaborative situation life cycle.

the collaborative situation life cycle starts with the CIM and moves from the CIM to the PIM and from the PIM to the PSM. The controlling helps to go back to the CIM and to start over a new cycle. But the dimension of collaborative situation life cycle is not that simple. The dimension could be opened and presented in a much more complex way. In order to present the dimensions correctly, the relations of collaborative life cycle (RCC) are defined (Figure 5).

As presented in the collaborative situation framework, the dimension of life cycle is separated as four layers: CIM, PIM, PSM, and controlling. The RCC in Figure 3 also contains these four layers. The CIM present or define the gathered collaborative knowledge. The knowledge of CIM is business knowledge. But the knowledge of PIM is technical knowledge. In order to move from the CIM to the PIM, there is a gap to fix. The gap is to add the technical knowledge and transfer from the CIM to the PIM. After gathering technical knowledge, the life cycle moves from the CIM to the PIM.

The knowledge of the PIM contains technical functions of each partner. But technical functions are not web services. The technical functions have to be implemented or executed by web services. Semantic web service is the next gap to fix. Then the PIM is transferred to the PSM. The life cycle moves from the PIM to the PSM. The PSM is deployed as mediation information system (MIS) at runtime (it is an ESB system to orchestrate BPEL file). Though the MIS is launched to invoke the whole collaborative process, there may be several kinds of failures and errors at runtime. This leads to the last layer of life cycle: the controlling. The controlling is a layer to decide which layers of design-time life cycle should be redone to point against the specific failures or changes at runtime. The RCC defines two kinds of life cycle:

(i) The first life cycle goes back to the PIM layer. It is designed to solve the failures of technical knowledge. For example, if the web service of one technical function is down, the semantic web service has to be redone to select new web services, which could implement the technical function.

(ii) The second life cycle goes back to the CIM layer. It is designed to correct the mistakes of business knowledge. For example, if a new partner entered the collaborative situation or a partner is no longer available for the collaborative situation, the life cycle has to restart all over from the beginning to collect the correct business information.

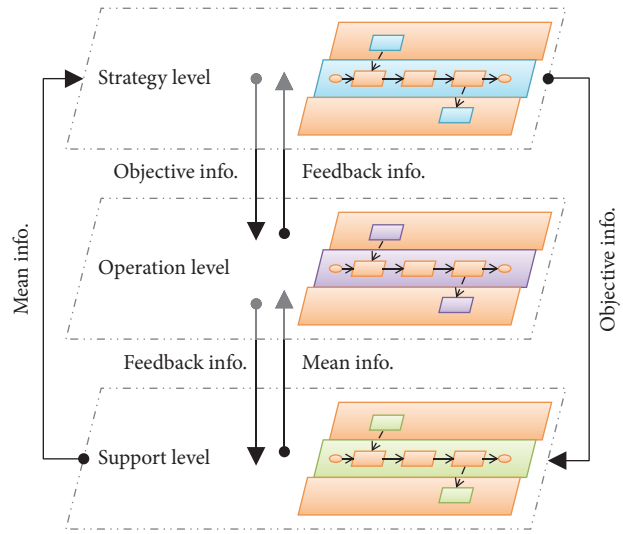


Figure 6: Relations among collaborative situation levels.

3.3. Relations among Levels. We have mentioned in previous section all the models, which have been defined in the RCE, cover strategy, operation, and support level. There are two reasons to define these three levels. First, ISO 9000 [42] has separated business process into three types: strategy, operation, and support. Second, in MISE 1.0 [44, 45], the collaborative process only covers operation level. With the practical experience of research projects, operation process is not enough for a collaborative situation, which involves decision-making and resource supporting. As the results of process deduction architecture, strategy, operation, and support collaborative processes are generated. But we do not know what are the communications among these processes? How could strategy process trigger an operation process? How could a support process complete an operation process? In order to answer these questions, the relations of collaborative levels (RCL) are defined to manage communications among different collaborative processes.

The communications among strategy level, operation level, and support level have been shown in Figure 6. Among

these three levels, three kinds of messages have been involved: objective information, feedback information, and mean.

(i) Objective information: objective is the goal, which is intended to attain. Objective information is a message, which contains the decision result of strategy level. The objective information could be sent to operation and support level. The operation level and the support level invoke homologous process and useful information to attain the goal in objective information.

(ii) Feedback information: feedback information is a message, which contains the operation level result. The feedback information is sent from operation level to strategy level. It is used to report the operational exception, error, result, and so on. Feedback information could also be sent from operation level to support level. This kind of feedback information is used to trigger or direct support process.

(iii) Mean: in the collaboration situation, mean is a message, which could contain any kind of information. It could be an exception, an error, a feedback, or a signal.

4. Case Study

The knowledge framework presented in this paper is used to develop MISE 2.0 project. This project aims to deduce automatically mediation information system to orchestrate the collaborative process among organizations. The methodology to develop MISE 2.0 project has been presented in [46, 47]. In this section, we will present the use of collaborative knowledge framework through a small collaborative case, and each step of the case used the methodology presented in [46, 47], but here we are not focusing on the method but the use result of the framework, so the case only shows the results of CIM, PIM, and PSM. Controlling phase is presented in [48]; it is not detailed in this case.

4.1. CIM: Organizational, Functional, and Informational Model Development [9, 47]. The knowledge in this phase covers the target collaborative situation. In the work of Dr. Rajsiri et al. [44], the initial knowledge is structured according to collaborative network, partners, and common goal. In the work of Dr. Truptil et al. [49], the shared functions of partners are added to the initial knowledge. The above two results are combined together and improved in the methodology. The collaborative network model and function model represent and define the initial collaborative situation. *It covers the CIM, organizational, functional, and informational knowledge involved strategy\operation\support levels*.

The collaborative network model (Figures 7 and 8) does not only collect the collaborative network, partners, and partner relations but also collect subcollaborative network and collaborative objectives. The objective of collaboration is classified into three types: strategy objective, operation objective, and support objective.

The function model (Figure 9) is defined based on IDEF 1. It represents the information concerning shared partner functions and input/output messages.

4.2. PIM: Process Model Transformation [9, 47]. In this phase, the collaborative ontology and transformation rules are defined to transfer the collaboration concepts to the mediation concepts in the collaborative ontology; meanwhile, (i) *transfer CIM to PIM, and* (ii) *transfer organizational, informational, and functional knowledge to process knowledge*. There are five groups of transformation rules: create mediator, create mediator relationship, create generated mediator function, link generated mediator function to mediator, and create intermediator function. Table 2 provides two equations of group 1 and group 2 as examples of transformation rules (equations in total 11). With the transformation rules, the mediation concepts are deduced, but there is not enough knowledge for the extraction of collaborative process, so the next phase comes.

The knowledge of this phase presents the matching between objective and functions. In this phase, one methodology is developed: business service selection to choose functions to achieve objectives by linking the functions and objectives to the instances of the collaborative ontology by using "same as" and "nearby" relations. This part of work is detailed in [50]. Figure 10 shows the interface in mediator modeling tool, which defines the "same as" and "nearby" relations.

For the method of process sequence deduction, the linkage of input/output messages and the objective based method are mixed to deduce the sequences and the gateways. First, the linkage of input/output messages among functions is used to get at global picture of the process. Second, the special place (the gateways are needed) of the global picture is taken and redone by using the objective based method. Finally, the linkage of messages checks the results of objective based method to get the best solution.

The knowledge covers the collaborative process extraction and sequence/gateway deduction. In this phase, the deduction rules are defined to extract the collaborative process cartography (Figure 11) and collaborative processes (Figure 19). To complete the sequence and the gateway, the method of sequence deduction is developed.

4.3. PSM: Technical Process Transformation [22, 47]. The first task of PSM is to match web services with business functions. *It covers organizational (web service provider), functional, and informational knowledge in PSM with strategy/operation/support levels*. Whereas a lot of annotation mechanisms exist for web services, the recent BPMN 2.0 is still devoid of a semantic standard. However, in addition to a higher design range (from very high level processes to executable workflows), this second major version brings an XML representation and its extension mechanism. Therefore, we decided to propose our own annotation mechanism, called SA-BPMN 2.0. This extension adds two XML tags: (i) SemanticDetails allows user to describe any activity requirement. It embeds both functional and internal behaviour description.

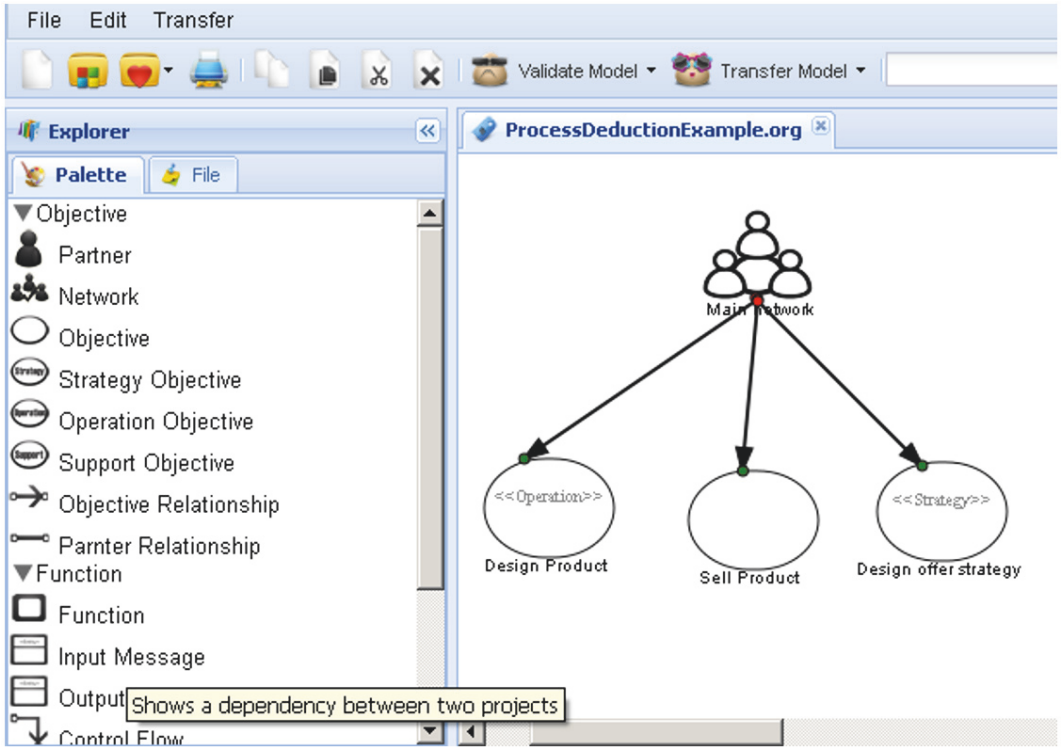


Figure 7: Define organizational model.

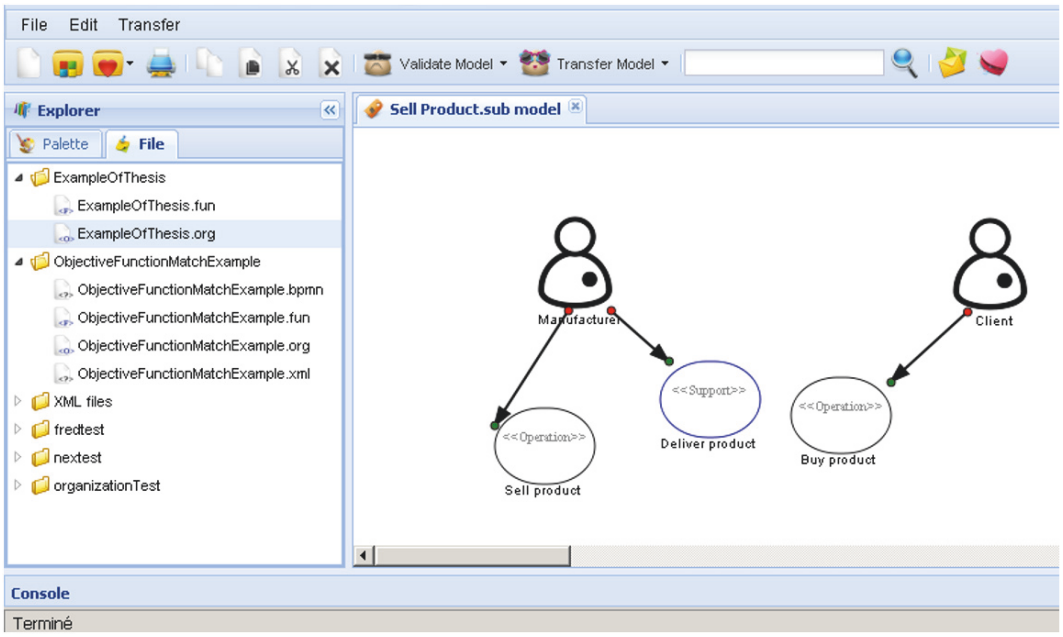


Figure 8: Define organizational model.

Each one contains a name and a list of URIs (corresponding to semantic concepts from any ontology); (ii) SemanticElements aims at describing messages and sequencing flows, attaching a list of expected messages or elements. Each element then contains the syntactic name coupled with a list of concepts.

To simplify semantic annotation, the modeling platform embeds annotation tools to allow users to add or edit semantic concept references directly from the business process view (see Figure 12). Semantic concepts come from partners' business ontologies, developed from scratch or based on MISE's one.

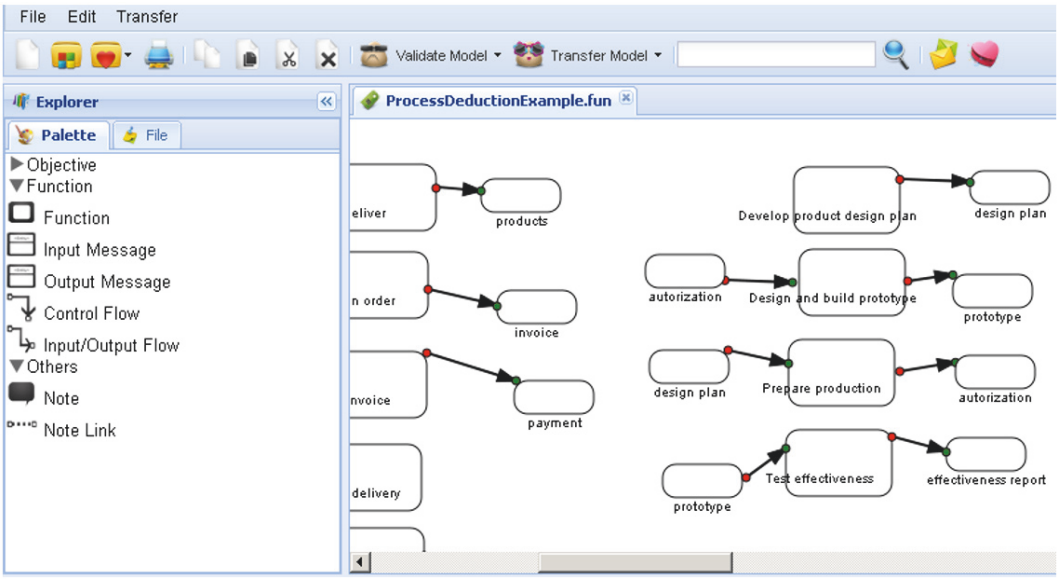


Figure 9: Define functional model.

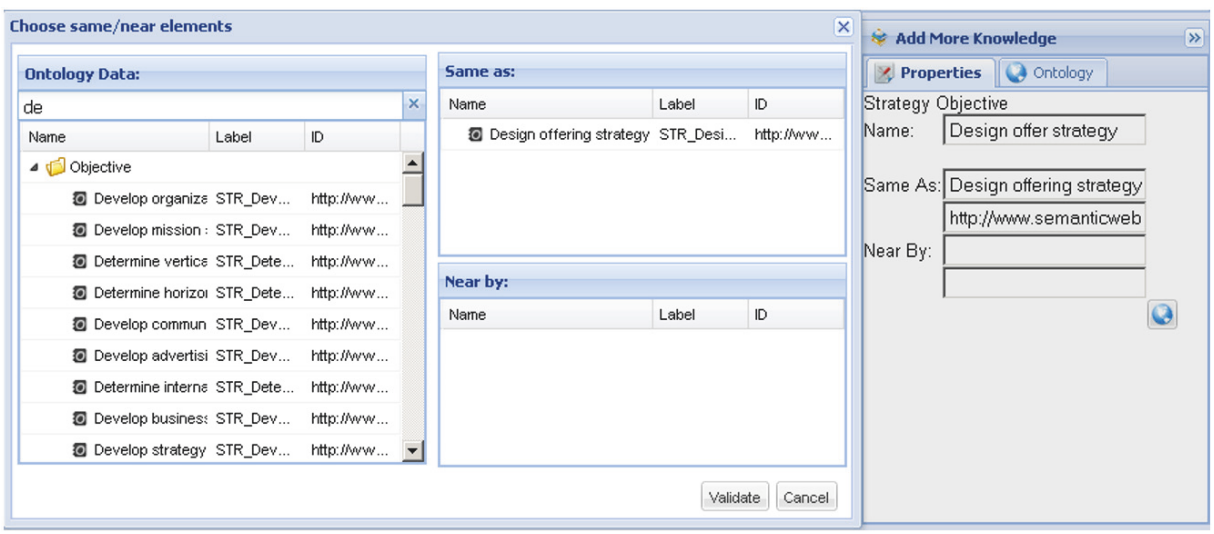


Figure 10: Select "same as" and "nearby" instances.

Table 2: Examples of transformation rules.

Group 1: create mediator	
Subnetwork → mediator	
∀Sub Network (X) (∀hasPartner (Sub Network (X), Partner (X_1)) ∧ (∀hasPartner (Sub Network (X), Partner ($X2$))∧ ⋯ ∧ (∀hasPartner (Sub Network (X), Partner (Xn))) ⟶ ∃Mediator (X) ∧∃hasMediator (Sub Network (X), Mediator (X))	(1)
Group 2: create mediator relationship	
Strategy and operation objective → main function → business message → order	
If ∀Strategy Objective ($X1$) (∀generates (Strategy Objective ($X1$), Main Function ($X1$))) ∧ ∀Operation Objective ($X2$) (∀generates (Operation Objective ($X2$), Main Function ($X2$))) If ∀Main Function ($X1$) (∀out (Main Function ($X1$), Business Message (m))) ∧ ∀Main Function ($X2$) (∀in (Main Function ($X2$), Business Message (m))) ⟶ ∃ Order (m) (hasMediatorRelationship (Mediator ($X1$), Order (m))) ∧ ∃ Order (m) (hasMediatorRelationship (Mediator ($X2$), Order (m)))	(2)

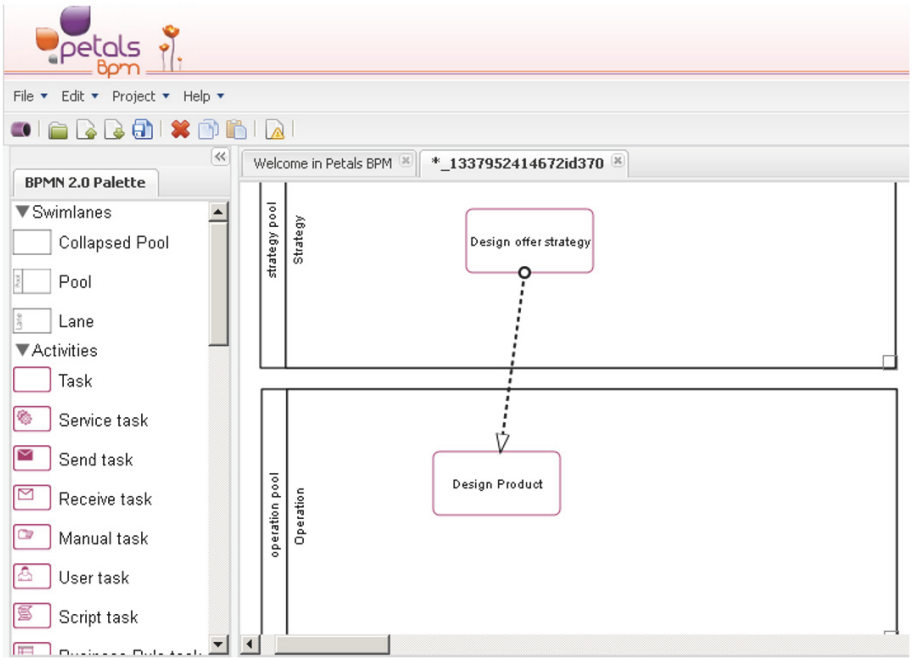


Figure 11: Extract process cartography.

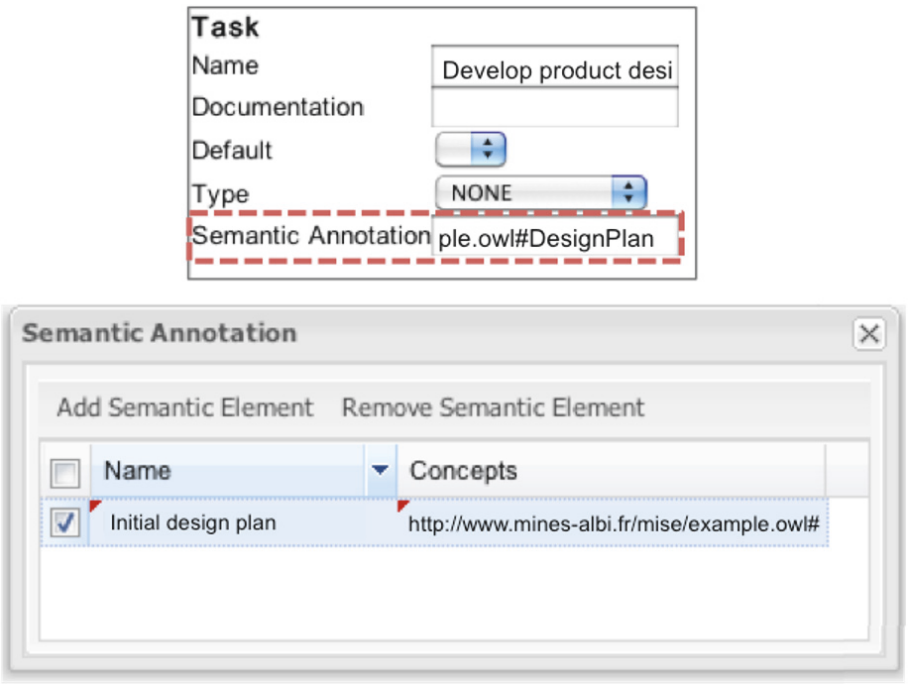


Figure 12: Semantic annotation of business process (SA-BPMN 2.0).

Once business processes annotated, we aim at matching business activity semantic descriptions with technical service ones. The proposed approach is based on a "1-to-1" hybrid matchmaking mechanism and focuses on semantic comparison. Semantic distance between profiles is performed thanks to a logic-based reasoning coupled with a syntactic similarity measurement. These measurements use information from operation (service capability or activity requirement) and I/O. In order to perform this service composition, and despite granularity difference of models to match, we use a semantic profile. This profile (represented in Figure 13) allows us to describe the functional aspects of models. It is filled with semantic annotation from business activities (using our SA-BPMN 2.0 mechanism) or technical services (using

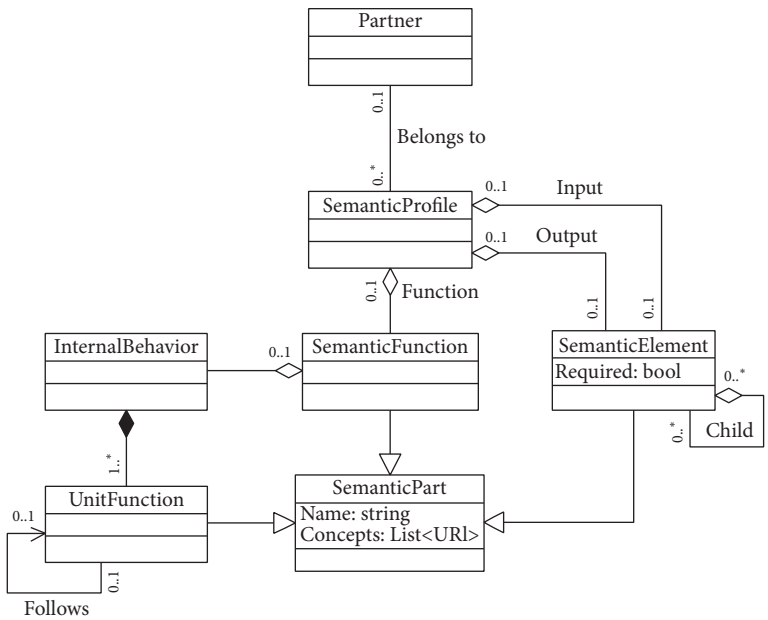


Figure 13: UML model of our semantic profile.

SAWSDL or WSMO-Lite for now). This profile also embeds an internal behaviour description, composed of a sequence of unit activities. Each of these unit activities is represented by a list of semantic concepts, such as functional description. This part enables description of business or technical sets in order to facilitate service composition.

Using syntactic and semantic information from business and technical profiles, our matchmaking mechanism then computes semantic and syntactic distance between models. In this view, we first perform a "1-to-1" service matching, comparing semantic concepts and names from both activities and web services profiles. If no service fits business requirements of the target activity, we then try to respond to the request using a set of services. In order to do so, we select the closest technical service and then deduce a new research profile, containing uncovered business concepts. This new profile, called complementary profile, corresponds to remaining business requirements if we use this first web service. At this time, we perform a new service matching using this profile and then compute the distance between the proposed sets of services and the initial business activity in order to propose "1-to-*n*" matching results to users. This mechanism is performed with several sets of possible services and activities using smart stopping conditions in order to suggest the best results to user while avoiding combinatorial problems.

Finally, service composition results are proposed to user for validation or selection. Figure 20 shows the dedicated interface, which provides rated results (on the left) for each activity or set of activities (on the right).

Once the user has selected technical services, we can focus on real data mapping. The discovery of web services that fit our functional needs is not enough to generate executable processes and ensure good communication between partners IS. We also have to provide interoperability between these services.

Semantic business information is not sufficient for message matchmaking. One business concept such as a date can be expressed in many formats (XML date time, US date format, etc.). This choice belongs to the service developer who can also use classic XML date time, declared as such in the service description, or choose to use an exotic one, declared as a simple string. In order to solve this problem, we propose a technical ontology focused on format concepts and linked to technical databases filled with syntax representation and conversion formulae.

Thanks to semantic and technical data description of involved messages, we generate data transformations using three main steps for each chosen service:

(i) First we search for available outputs using process logic. We have to find out which previous output messages can be used to create our input target message.

(ii) Then, using this available data, we try to compute the whole message transformation using semantic links between tags, format descriptions, and technical information about known transformations.

(iii) If the whole message is not covered by the computed transformation, we first try to find an available transformation service using our service matchmaking mechanism described above. We then submit results to the user for validation or completion.

Once all transformations are validated or completed, our concrete level management mechanism generates the executable workflow (using BPEL or BPMN 2.0 language, depending on targeted execution engine). Links between business activities and composed technical services are stored

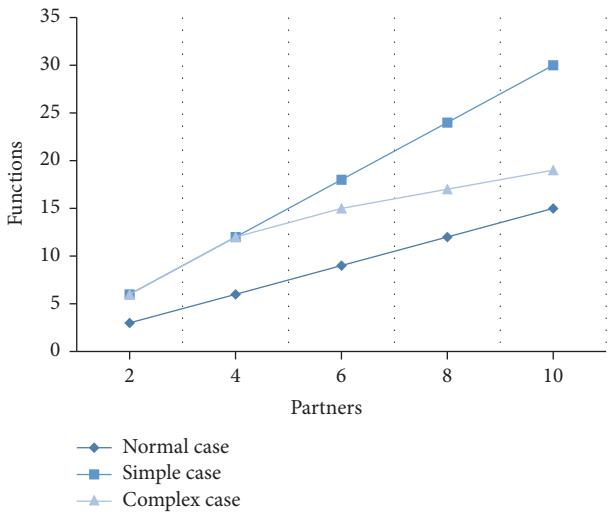


Figure 14: Number of functions in different cases [9].

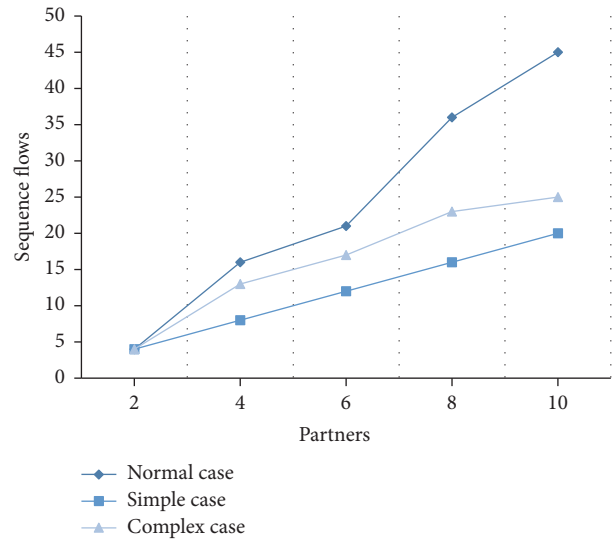


Figure 15: Number of sequences in different cases [9].

in order to enable business monitoring during the runtime phase (see Figure 21). *It covers process knowledge in PSM with strategy/operation/support levels.*

5. Evaluation

5.1. Part One: Evaluation of Collaborative Framework. Three cases were built to calculate the number of functions (CIM) [9], sequence flows (PIM) [9], and web services (PSM). The performance of collaborative knowledge framework in each life cycle can be evaluated. "Normal case" means that the collaborative process goes from one partner to another partner with MISE. The "simple case" means that there is a mediator but only mathematic calculation. For example, one partner function is invoked by one mediator function, and the number of partner functions and mediator functions should be equal. If we consider only the mathematics, the number of total functions is simply doubled. The "complex case" is the real result of MISE with collaborative framework presented in this paper.

As shown in Figure 14, for the "complex case," by the increase in partners, the number of functions can be decreased and is infinitely close to the "no mediator" case. In Figure 15, the "mediator" combines the same functions of partners into one invoking function. For the simple case, more partners lead to more sequence flows. For the complex case, with the invoking function, more partners lead to more sequence flows being saved. With the merging and invoking functions, the complexity of the collaborative process decreases. Figure 16 shows the numbers of web services in three cases. As the complexity is increasing, the number of web services in MISE is much less than the one in simple case. *The MISE methodology with collaborative knowledge framework shows the strong advantage of addressing a complex collaborative situation.*

Except the evaluation of each step, we also did an evaluation between MISE 1.0 and MSIE 2.0. During the research of MSIE 1.0, the collaborative knowledge network

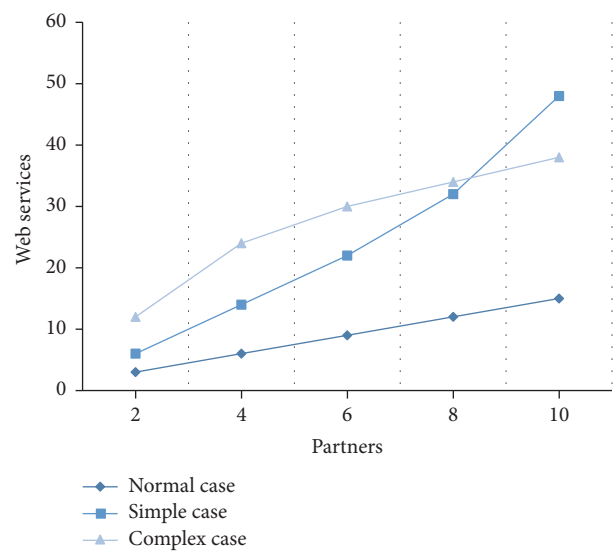


Figure 16: Number of web services in different cases.

was not yet developed. All the research work is depending on the MDA and SOA theory. With the complement of MISE 1.0, many problems appear. So in the research of MISE 2.0, the collaborative knowledge framework is developed first to avoid the mistakes in MISE 1.0 and to conclude new considerations, agility, and automation. Figure 22 shows the evaluation results:

(i) Cause of the addition of controlling: MISE 2.0 has strong agility.

(ii) Cause of the addition of knowledge gathering process and knowledge classification: MSIE 2.0 gathered more complete knowledge in organization, function, process, and data.

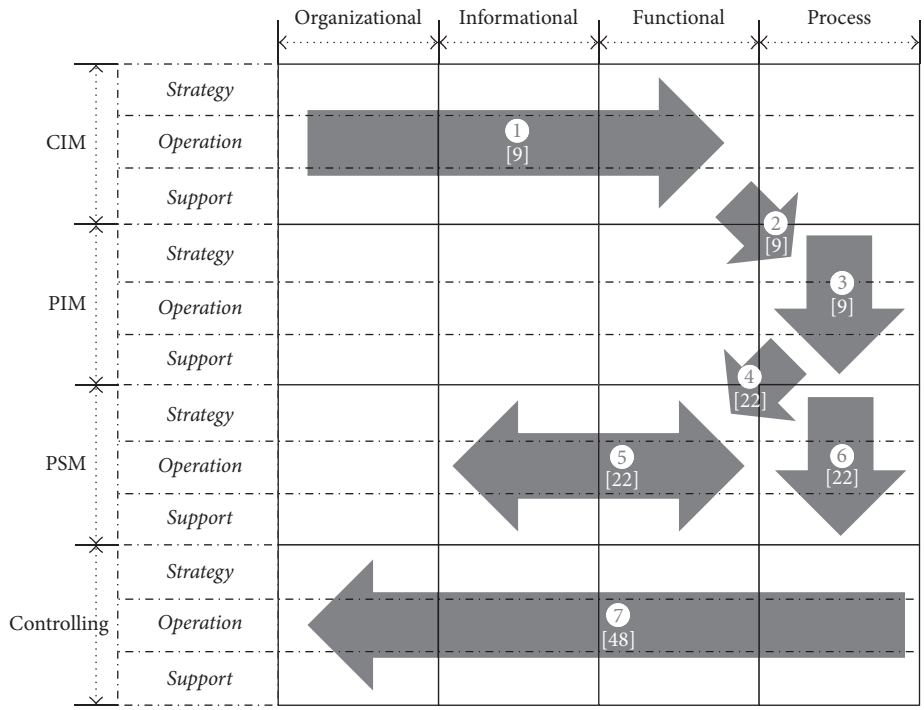


Figure 17: The developing path of MISE 2.0 methodology.

(iii) Cause of the collaborative levels: even though both have the same level of interoperability, MISE 2.0 has clearer process levels and process cartography.

5.2. Part Two: Comparisons of Related Works. The collaborative knowledge framework defines the knowledge that should be gathered or covered during the collaboration. Section 6 has introduced the MISE 2.0 developing methodology; this methodology is based on the framework and follows the knowledge gathering steps which are defined in the framework. In order to evaluate the framework, the MISE 2.0 case is located in Figure 17.

Several problems have been found:

(i) Once strategy/operation/support objectives and functions have been collected in CIM, the strategy/operation/support knowledge in PIM and PSM can be skipped.

(ii) Once organization/functional/informational knowledge has been collected in CIM, the process knowledge of CIM can be skipped. The transformation directly to the process knowledge in PIM is more useful.

(iii) In PSM, the informational and functional knowledge is more important for the transformation of process in PSM, and the organizational information can be skipped.

In order to evaluate the collaborative knowledge framework, we searched papers published from 2015 to 2017 in Web of Science, using key words "collaborative knowledge" and "framework." 338 papers have been found. After manual selection, we got 13 papers which are strongly related to this paper. Those 12 papers are summarized in Table 3 and Figure 18. [18, 20] are review papers. The paper [18] reviewed all the enterprise architectures from life cycle and modeling views. Compared with the collaborative knowledge framework, the modeling views are similar with regard to organizational, informational, functional, and process elements (collaborative situation elements). But the life cycles are very different depending on the purpose of framework. All those 13 papers can be located into strategy/operation/support levels (Figure 5). For controlling, there are no papers located. But another word for controlling could be agility; [48] gives a careful review according to agility. So this step is skipped in this paper.

We conclude the following:

(i) The collaborative knowledge framework did give a guide to gather knowledge and deduce automatically the collaborative process and workflow.

(ii) The final purpose of collaborative knowledge framework is to develop a MIS based on MDA; the life cycle is different from others.

(iii) The same knowledge gathering has been repeated in the framework; our suggestion is to gather organizational/information/functional in CIM, deduce process in PIM, carefully gather informational/functional in PSM, and deduce workflow in PMS.

(iv) For controlling, depending on different event, the knowledge should be adapted back to difference levels.

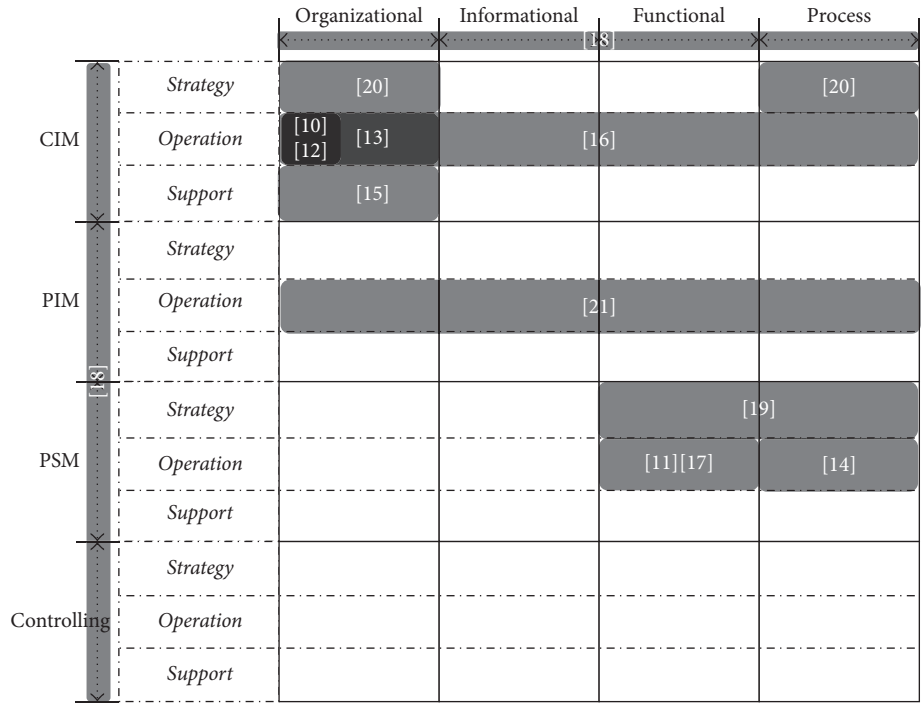


Figure 18: The positions of related papers.

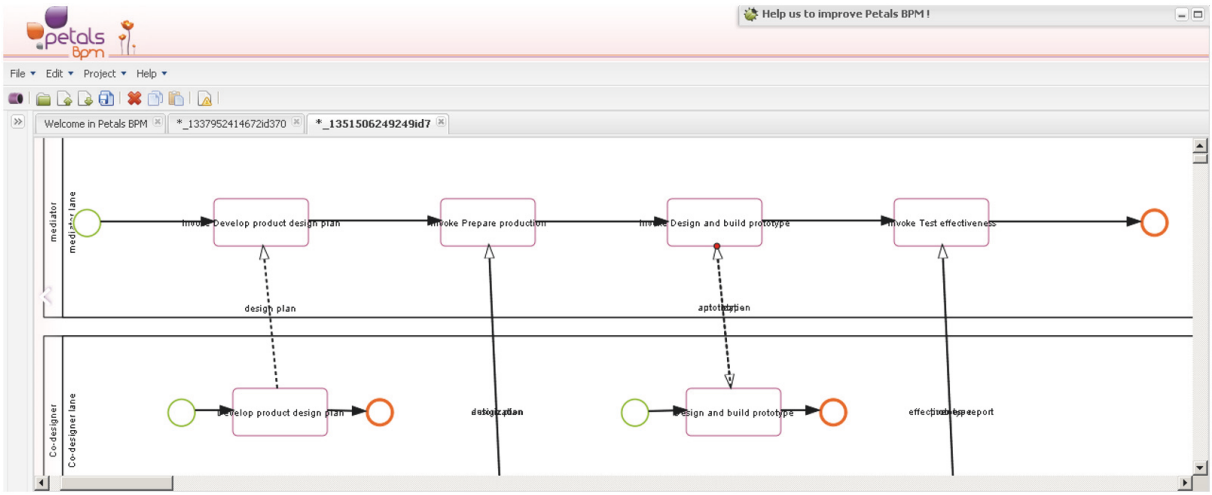


Figure 19: Extract collaborative processes.

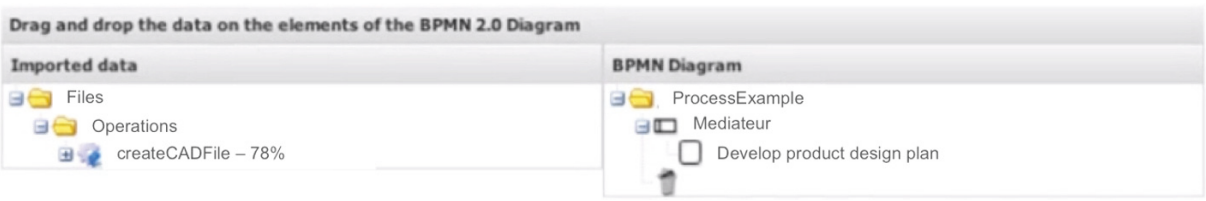


Figure 20: Service matchmaking validation by user.

Table 3: Summary of collaborative knowledge works.

Ref.	Published year	Framework/ knowledge modeled	Method
[10]	2016	Collaborative network on operation	Questionnaire
[11]	2017	Web services	Semantic annotation
[12]	2017	Collaborative network on operation	Matrix for collaborative innovation
[13]	2017	Collaborative network and objectives on operation	Interview and case study
[14]	2016	Process in PSM	Knowledge based software developing
[15]	2016	Collaborative network on resources	ABC framework and case study
[16]	2016	Collaborative agencies and process on operation (material, time, info)	Knowledge framework for collaborative simulation
[17]	2016	Web services	Service ontology
[18]	2016	Life cycle and modeling views	Review
[19]	2016	Web services and process on strategy	Ontology learning
[20]	2016	Actor and behavior on CIM	Review
[21]	2015	Partners and process on operation	Case study

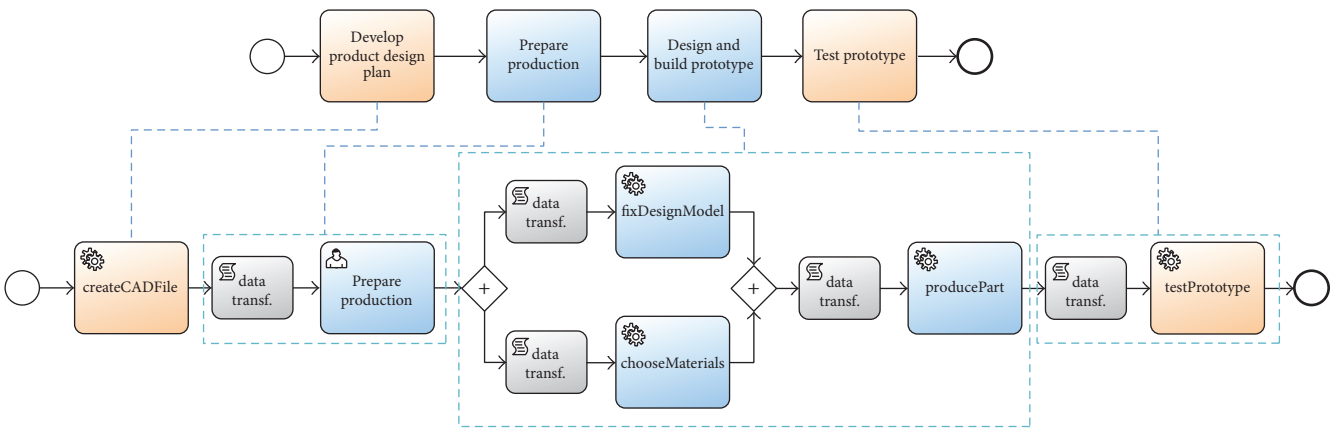


Figure 21: Comparison between business process and generated workflow.

6. Conclusion

MISE 2.0 aims to develop a mediation information system, which manages process orchestration, data conversion, and service selection in enterprises' information systems. To do so, the first problem is to define or deduce a business collaborative process. This paper presents abstract framework for deducing business collaborative process model. In relations of elements, organizational model, collaborative network model, IDEF0 based functional model, IDEF1 informational model, and BPMN based collaborative process model are used to present collaborative knowledge. Metamodel is defined to confirm each model. Relations of life cycle define the agility management in the MISE 2.0. In relations among levels, we defined types of messages transferred among strategy, operation, and support process.

With the accomplishment of models, metamodel, and transformation rules, software tool is going to develop to support models' building and transformation rules' implementation. The MISE 2.0 abstract level software tool should implement the following three main functions: (i) creation of organizational model, functional model, informational model, and process model (use GWT and Java 2D graphical design); (ii) transformation from organizational model, functional model, and informational model to process model (use JDOM, Java, or ATL); (iii) extraction of the BPMN collaborative process cartography (use JDOM and Java). The detailed explanation of deduction of collaborative process cartography is presented in [51].

The whole BPMN collaborative process cartography is provided to MISE 2.0 concrete level. Concrete level concerns MIS deployment. Firstly, with provided process cartography in abstract level, web services are selected automatically by semantic annotation and semantic ontology. And then business process cartography is transferred into executable technical process. The BPMN based collaborative process

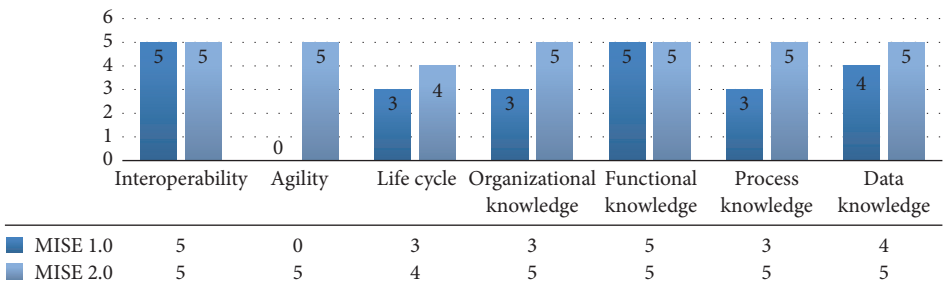


Figure 22: The developing path of MISE 2.0 methodology.

cartography is transferred to BPEL [52] file and deployed in ESB (Enterprise Service Bus). The concrete level work is presented in [53].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by ARMINES (Acteur de l'Innovation par la Recherché Partenariale) in France and National Higher-Education Institution General Research and Development Funding of China (B15JB00340).

References

[1] D. Chen, G. Doumeingts, and F. Vernadat, "Architectures for enterprise integration and interoperability: past, present and future," *Computers in Industry*, vol. 59, no. 7, pp. 647–659, 2008.

[2] C. Nicolle, J. C. Simon, and K. Yetongnon, "Interoperability of information systems," *Database*, vol. 1651, p. 1650, 2005.

[3] D. S. Milojicic, V. Kalogeraki, R. Lukose et al., "Peer-to-peer computing," *Citeseer*, 2002.

[4] N. Guarino and P. Giaretta, "Ontologies and knowledge bases—towards a terminological clarification," in *Towards Very Large Knowledge Bases*, pp. 25–32, IOS Press, Amsterdam, The Netherlands, 1995.

[5] G. Wiederhold, "Mediators in the architecture of future information systems," *Computer*, vol. 25, no. 3, pp. 38–49, 1992.

[6] J. Bézivin, S. Gérard, P. A. Muller, and L. Rioux, *MDA Components: Challenges and Opportunities*, 2003.

[7] N. Josuttis, *SOA in Practice*, O'reilly, 2007.

[8] J. Miller and J. Mukerji, "MDA Guide Version 1.0. 1," *Object Management Group*, vol. 234, p. 51, 2003.

[9] W. Mu, F. Bénaben, and H. Pingaud, "A methodology proposal for collaborative business process elaboration using a model-driven approach," *Enterprise Information Systems*, vol. 9, no. 4, pp. 349–383, 2015.

[10] S. Oppl, "Supporting the Collaborative Construction of a Shared Understanding About Work with a Guided Conceptual Modeling Technique," *Group Decision and Negotiation*, vol. 26, no. 2, pp. 247–283, 2017.

[11] G. Guerrero-Contreras, J. L. Navarro-Galindo, J. Samos, and J. L. Garrido, "A collaborative semantic annotation system in health: towards a SOA Design for knowledge sharing in ambient intelligence," *Mobile Information Systems*, vol. 2017, Article ID 4759572, 10 pages, 2017.

[12] B. Knoke, M. Missikoff, and K.-D. Thoben, "Collaborative open innovation management in virtual manufacturing enterprises," *International Journal of Computer Integrated Manufacturing*, vol. 30, no. 1, pp. 158–166, 2017.

[13] W. Medema, J. Adamowski, C. Orr, A. Furber, A. Wals, and N. Milot, "Building a foundation for knowledge co-creation in collaborative water governance: Dimensions of stakeholder networks facilitated through bridging organizations," *Water*, vol. 9, no. 1, article no. 60, 2017.

[14] M. Milosevic, D. Lukic, A. Antic, B. Lalic, M. Ficko, and G. Simunovic, "e-CAPP: A distributed collaborative system for internet-based process planning," *Journal of Manufacturing Systems*, vol. 42, pp. 210–223, 2017.

[15] E. Yeboah-Assiamah, K. Muller, and K. A. Domfeh, "Rising to the challenge: a framework for optimising value in collaborative natural resource governance," *Forest Policy and Economics*, vol. 67, pp. 20–29, 2016.

[16] Q. Long, "A multi-methodological collaborative simulation for inter-organizational supply chain networks," *Knowledge-Based Systems*, vol. 96, pp. 84–95, 2016.

[17] S. Y. Xu and B. Raahemi, "A semantic-based service discovery framework for collaborative environments," *International Journal of Simulation Modelling*, vol. 15, no. 1, pp. 83–96, 2016.

[18] A. Vargas, L. Cuenca, A. Boza, I. Sacala, and M. Moisescu, "Towards the development of the framework for inter sensing enterprise architecture," *Journal of Intelligent Manufacturing*, vol. 27, no. 1, pp. 55–72, 2016.

[19] R. Costa, C. Lima, J. Sarraipa, and R. Jardim-Gonçalves, "Facilitating knowledge sharing and reuse in building and construction domain: an ontology-based approach," *Journal of Intelligent Manufacturing*, vol. 27, no. 1, pp. 263–282, 2016.

[20] C. Durugbo, "Collaborative networks: A systematic review and multi-level framework," *International Journal of Production Research*, vol. 54, no. 12, pp. 3749–3776, 2016.

[21] K. Shahriari, A. G. Hessami, A. Jadidi, and N. Lehoux, "An approach toward a conceptual collaborative framework based on a case study in a wood supply chain," *IEEE Systems Journal*, vol. 9, no. 4, pp. 1163–1172, 2015.

[22] N. Boissel-Dallier, F. Benaben, J.-P. Lorré, and H. Pingaud, "Mediation information system engineering based on hybrid service composition mechanism," *Journal of Systems and Software*, vol. 108, pp. 39–59, 2015.

[23] D. A. Chappell, *Enterprise Service Bus*, O'Reilly Media, 2004.

[24] A.-M. Barthe-Delanoë, S. Carbonnel, F. Bénaben, and H. Pingaud, "Event-driven agility of crisis management collaborative

processes," in *Proceedings of the 9th International Conference on Information Systems for Crisis Response and Management (ISCRAM '12)*, Vancouver, Canada, April 2012.

[25] L. M. Camarinha-Matos and H. Afsarmanesh, "On reference models for collaborative networked organizations," *International Journal of Production Research*, vol. 46, no. 9, pp. 2453–2469, 2008.

[26] J. A. Zachman, "A framework for information systems architecture," *IBM Systems Journal*, vol. 26, no. 3, pp. 276–292, 1987.

[27] DTIC, *Technical Architecture Framework for Information Management. Volumes 1–8. Version 3.0 (Computer Diskette)*, 1996.

[28] N. Umheh, A. Miller, and C. Dagli, *TOGAF vs. DoDAF: Architecting Frameworks for Net-Centric Systems*, 2007.

[29] C4ISR, *C4ISR Architecture Framework Version 2.0*, AWG—US Department of Defence, 1997.

[30] B. Biggs, "Ministry of Defence Architectural Framework (MODAF)," *IEE Seminar on UML Systems Engineering*, vol. 2005, no. 10814, pp. 43–82, 2005.

[31] J. Zachman, *The Zachman Framework for Enterprise Architecture*, Zachman International, 2002.

[32] D. Chen, B. Vallespir, and G. Doumeingts, "GRAI integrated methodology and its mapping onto generic enterprise reference architecture and methodology," *Computers in Industry*, vol. 33, no. 2-3, pp. 387–394, 1997.

[33] G. Doumeingts, B. Vallespir, and D. Chen, "GRAI GridDecisional modelling," in *Handbook on Architectures of Information Systems*, pp. 321–346, 2006.

[34] T. J. Williams and H. Li, "PERA and GERAM—enterprise reference architectures in enterprise integration," in *Information Infrastructure Systems for Manufacturing II*, pp. 3–30, 1999.

[35] A. W. Scheer, *ARIS-Business Process Modeling*, Springer, 2000.

[36] K. Kosanke, "CIMOSA - Overview and status," *Computers in Industry*, vol. 27, no. 2, pp. 101–109, 1995.

[37] J. Vamus and N. Panaich, *TOGAF 9 Survey Results Presentation*, 2009.

[38] I. I. Force, "GERAM: Generalised Enterprise Reference Architecture and Methodology," IFIP-IFAC Task Force Archit. Enterp. Integr. Tech Rep, 1999.

[39] C. I. O. Council, *Federal Enterprise Architecture Framework (FEAF)—Version 1.1*, 1999.

[40] IDEAS, *A Gap Analysis—Required Activities in Research, Technology and Standardisation to Close The RTS Gap—Roadmaps And Recommendations on RTS Activites*, 2003.

[41] A. Berre et al., "The ATHENA interoperability framework," in *Enterprise Interoperability II*, pp. 569–580, 2007.

[42] ISO 9000, "ISO 9000 Quality management," September 2005, http://www.iso.org/iso/home/store/publications_and_e-products/publication_item.htm?pid=PUB100224.

[43] ISO 9000 X50-130, "NF EN ISO 9000 X50-130 Systèmes de management de la qualité—Principes essentiels et vocabulaire," October 2005, http://catdoc.mines-albi.fr:8080/Record.htm?idlist=6&record=19143202124919614849.

[44] V. Rajsiri, J.-P. Lorré, F. Bénaben, and H. Pingaud, "Knowledge-based system for collaborative process specification," *Computers in Industry*, vol. 61, no. 2, pp. 161–175, 2010.

[45] J. Touzi, F. Benaben, H. Pingaud, and J. P. Lorré, "A model-driven approach for collaborative service-oriented architecture design," *International Journal of Production Economics*, vol. 121, no. 1, pp. 5–20, 2009.

[46] F. Benaben and F. B. Vernadat, "Information System agility to support collaborative organisations," *Enterprise Information Systems*, vol. 11, no. 4, pp. 470–473, 2017.

[47] F. Benaben, W. Mu, N. Boissel-Dallier, A.-M. Barthe-Delanoe, S. Zribi, and H. Pingaud, "Supporting interoperability of collaborative networks through engineering of a service-based Mediation Information System (MISE 2.0)," *Enterprise Information Systems*, vol. 9, pp. 556–582, 2015.

[48] A.-M. Barthe-Delanoë, S. Truptil, F. Bénaben, and H. Pingaud, "Event-driven agility of interoperability during the Run-time of collaborative processes," *Decision Support Systems*, vol. 59, no. 1, pp. 171–179, 2014.

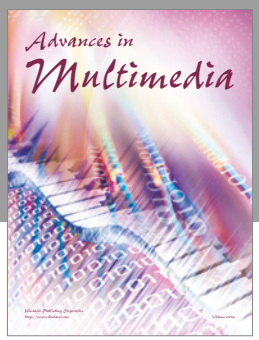
[49] S. Truptil, F. Benaben, H. Pingaud, and C. Hanachi, "Une architecture de système d'information collaboratif pour la gestion de crise," in *Proceedings of the INFORSID09*, 2009.

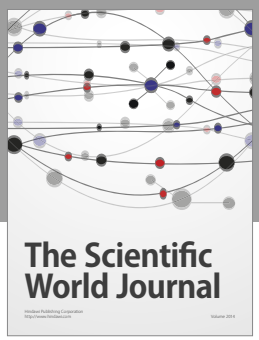
[50] W. Mu, F. Bénaben, and H. Pingaud, "An ontology based collaborative business service selection—contributing to automatic building of collaborative business process," in *Risks and Resilience of Collaborative Networks*, L. M. CamarinhaMatos, F. Benaben, and W. Picard, Eds., vol. 463, pp. 639–651, Springer, Berlin, Germany, 2015.

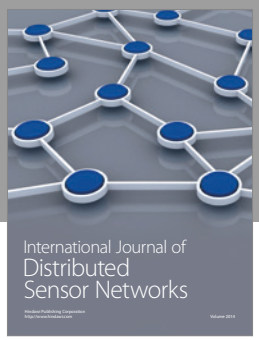
[51] W. Mu, *Caractérisation et logique d'une situation collaborative*, INPT, Toulouse, France, 2012.

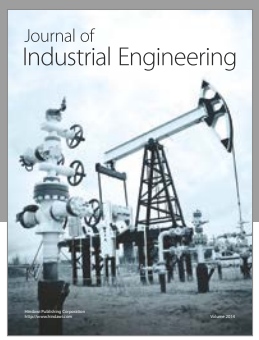
[52] S. Lee, T.-Y. Kim, D. Kang, K. Kim, and J. Y. Lee, "Composition of executable business process models by combining business rules and process flows," *Expert Systems with Applications*, vol. 33, no. 1, pp. 221–229, 2007.

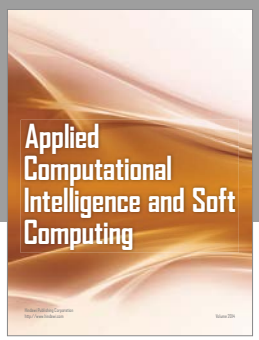
[53] F. Bénaben, N. Boissel-Dallier, J.-P. Lorré, and H. Pingaud, "Semantic reconciliation in interoperability management through model-driven approach," *IFIP Advances in Information and Communication Technology*, vol. 336, pp. 705–712, 2010.

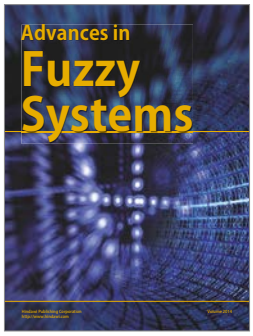


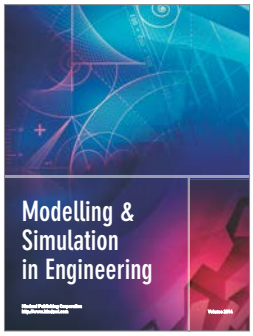


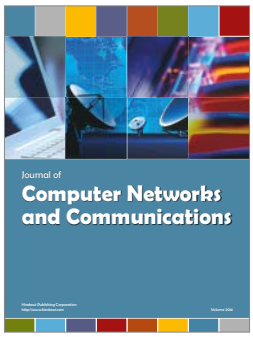


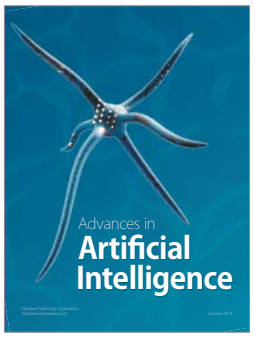
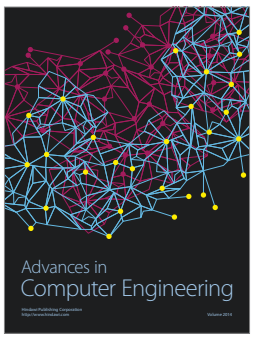
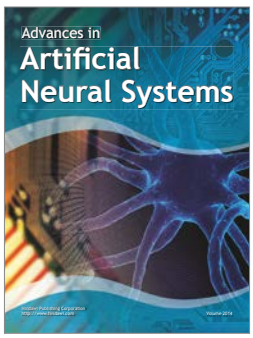
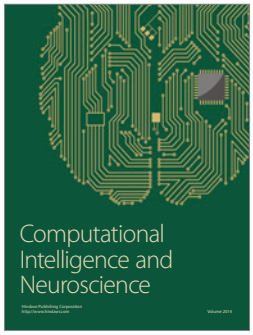










Hindawi

Submit your manuscripts at
https://www.hindawi.com