



HAL
open science

Towards a conceptualisation of interoperability requirements

Sihem Mallek, Nicolas Daclin, Vincent Chapurlat

► **To cite this version:**

Sihem Mallek, Nicolas Daclin, Vincent Chapurlat. Towards a conceptualisation of interoperability requirements. international confereneec on Interoperability for Enterprise Software and Applications, IESA'2010, Apr 2010, Coventry, United Kingdom. 10.1007/978-1-84996-257-5_41 . hal-00588822

HAL Id: hal-00588822

<https://hal.science/hal-00588822>

Submitted on 8 Jun 2022

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.

Towards a Conceptualisation of Interoperability Requirements

Sihem Mallek¹, Nicolas Daclin¹ and Vincent Chapurlat¹

¹ Laboratoire de Génie Informatique et d'Ingénierie de Production - LGI2P - site EERIE de l'Ecole des Mines d'Alès, Parc Scientifique Georges Besse, F30035 Nîmes cedex 5, France.

Abstract. Enterprises that aim to work together want, prior to any effective collaboration, to know if they are able to interoperate. On the one hand, this induces to be able to define the particular needs having to be taken into account in order to demonstrate if an enterprise can be or must be interoperable. On the other hand, it requires techniques and approaches allowing to formalise these needs as a set of unambiguous and, as formal as possible, requirements called here interoperability requirements. Finally, verification techniques can be used to detect how and where some of these requirements cannot be satisfied. This allows to highlight interoperability problems. This paper focuses on the two first phases and describes an approach to define and to formalise interoperability needs into interoperability requirements. These requirements are decomposed on three classes named compatibility, interoperation and reversibility requirements.

Keywords: Interoperability requirements, compatibility, interoperation, reversibility, verification, enterprise.

1. Introduction

Collaboration between enterprises, to fulfill a common project, remains difficult and can be considered as efficient if these enterprises are really able to interact. In this context, the development of interoperability has become an important issue, in today's globalized world, to facilitate, optimize and improve these interactions. Basically, interoperability is defined in [1] as "*the ability of two or more systems or components to exchange information and to use the information that has been exchanged*". Therefore, interoperability can be seen as a set of needs that enterprises have to satisfy. However, these needs remain difficult to specify exhaustively considering the enterprise nature, organisation, skills *etc.* Thus, the description of an interoperability need can be ambiguous, not completely coherent and not complete. This is due to their specification based often on complex

expression in natural language. As a consequence, to obtain an unambiguous and clear description of the interoperability needs, a formalisation of these needs is necessary. On the one hand, this formalisation allows to clarify and to structure each need as a set of requirements represented in a model of requirements. These requirements are done by the use of a limited set of concepts and terms coming from natural language. On the other hand, this formalisation allows to become able to prove each need. This proof can be based on several techniques (expertise, behavioural simulation, formal approaches...). The here research work proposes that, formal approaches such as verification techniques will be used to verify if these requirements are satisfied by enterprises. This verification is performed thanks to requirements model applied on collaborative model of enterprises. Moreover, to make this verification possible, formalisation of requirements must be made by (1) using formal languages such as mathematical language, and (2) an enrichment of the modelling languages used to describe the enterprises and their collaborative processes. The here proposed research focuses on interoperability requirements specification and verification. It allows:

- to characterise what are the needs of an enterprise which want to improve its interoperability and its ability to interact. The goal is to determine these needs by using different approaches such as literature analysis and questionnaire to conduct interviews.
- to formalise these needs in the form of a set of unambiguous, and as formal as possible, requirements called “*interoperability requirements*”. The main objective of this formalisation is to provide a requirement model allowing to define and structure interoperability requirements as proposed in different approaches such as [2], [3].
- to use this formalisation in order to verify (*via* formal verification techniques) if the enterprise components respect or not these requirements. This verification allows to highlight interoperability problems. The goal is to adapt existing techniques of verification. This adaptation has to consider (1) the model of interoperability requirements ; (2) existing modelling languages commonly used in enterprises and collaborative process modeling domain [4].

This paper is structured as follows. Section 2 introduces interoperability needs with the presentation of different relevant works on the domain. An interoperability requirements model is presented and explained in section 3. To demonstrate the usefulness of the interoperability requirements model, a case study is briefly presented in section 4. Finally, section 5 presents a conclusion and outlines perspectives.

2. Interoperability Needs and Related Works

Enterprises face many challenges related to the lack of interoperability such as loss of money due to the necessary efforts for an effective collaboration. Thus, the development of interoperability is become a crucial question for enterprises that want to gain competitiveness in a globalized environment. As a consequence,

several works have been developed from the last years in order to reply to the needs of interoperability expressed by enterprises.

For instance, numerous maturity models (LISI, LCIM, OIM...) [5], [6], [7] have been developed. These models allow systems to know its ability to interoperate (at a given moment) with regards to different interoperability levels. Some of these models give recommendations that allow a given system to evolve from its own interoperability level toward full interoperability.

In [8] a methodology is proposed to measure interoperability of enterprises. This measure includes two measures such as compatibility measurement and interoperation performance measurement. To perform the measure of compatibility, a matrix of compatibility is used to inform enterprises on the attendance (or not) of interoperability problems. To perform measure of interoperation performance, some criteria such as cost of interoperation, time of interoperation and quality of interoperation are measured.

Furthermore, in [9], characterisation of interoperability is proposed to evaluate performance of enterprise interoperability. This characterisation identifies requirements as “*rules of interoperability*”. Five rules of interoperability are identified. However, these rules are generic and hardly difficult to exploit (e.g. rule 1 explains that a feedback loop is required between two interoperating systems).

To structure and capitalize these different works, these ones are integrated in an enterprise interoperability framework developed in [10]. This framework considers two fundamental aspects to treat interoperability needs: the “*interoperability barriers*” and “*interoperability concerns*”. The goal is to achieve an efficient collaboration by solving interoperability barriers (conceptual, organisational and technological) with regards to interoperability concerns (data, services, processes and business).

Based on these different works, this research work proposes and provides a model of interoperability requirements, a structure and a method of work (such as REGAL [2]) that enables a user to be guided – with a minimum of omission - in the expression of its needs.

3. Interoperability Requirements Model

As mentioned before, interoperability can be seen as a set of needs (often ambiguous and complex) and furthermore as requirements (more formal, clear and unambiguous) that enterprises must satisfy to be interoperable. A requirement can be defined as “*a statement that specifies a function, ability or a characteristic that a product or a system must satisfy in a given context*” [11].

Generally speaking, interoperability is related to the concept of compatibility. Compatibility means to harmonize enterprises (method, organisation, tool...) together so that heterogeneous information exchanged can be understood and exploited by each ones without interfacing effort. However, compatibility is not sufficient to formulate interoperability requirements. In fact, during the collaboration between two enterprises, many problems, other than compatibility, can occur. For instance, two enterprises use the same language to code their data, but one of them is unable to send its data. In this case, “*two interoperable systems*

are compatible, but two compatible systems are not necessarily interoperable” [12]. This statement leads to consider a second kind of interoperability requirements called “*interoperation requirements*”. These requirements are related to the interaction between enterprises during the collaboration. It means that enterprises can work efficiently together when interacting.

Moreover, after working enterprises wish to retrieve their performance and autonomy while remaining efficient. For instance, after collaboration, one of these enterprises may lose performance that it had before this collaboration. In this case, it is question to be sure that a given enterprise will reach again, its own performance after the end of collaboration. Thus, interoperability considers another concept such as the reversibility [10]. Reversibility means that enterprises can return to their original state and retrieve their autonomy at the end of collaboration with regards to its own performance and including positive and/or negative variation that accepted. As a consequence, requirements related to the reversibility of enterprises are also required.

In summary, interoperability requirements can be decomposed in three categories such as: compatibility requirements, interoperation requirements and reversibility requirements. For instance, these three categories are fully in agreement with the virtual enterprise life cycle (creation, operation and dissolution) where the implementation of interoperability plays a preponderant part [13]. The relationship which consistently relates these three categories with the VE life cycle is highlighted in figure 1. Thus, two enterprises which aim to work together must, firstly create the collaboration, then collaborate efficiently, and finally stop the collaboration. The creation phase takes place before collaboration when a business opportunity is detected. The need is to know if the future collaboration can take place or not in good condition. Compatibility requirement are deployed and verified (e.g. during the collaboration, if a document from a given format is received by a person who dispose of an earlier version, he cannot read this document). Through the runtime phase, a good interoperation is required for an efficient collaboration (e.g. if a product is received later than required in an enterprise, it can retard the continued collaboration). The dissolution phase takes place after the collaboration where reversibility of enterprises is required to return to their original state.

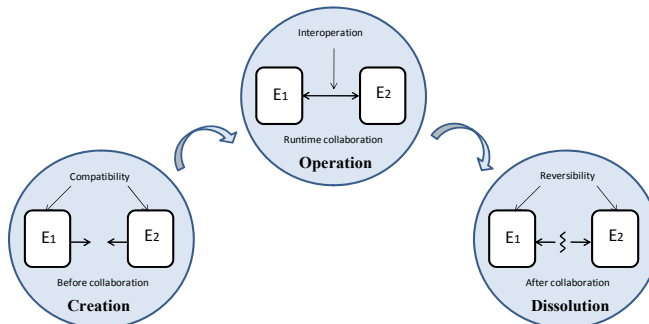


Fig. 1. The three interoperability requirement categories related to the VE life cycle [adapted from 13]

Furthermore, the set of requirements in each category (compatibility, interoperation and reversibility) must be represented clearly and in an unambiguous way. For this reason, the principle of a requirement model as a causal tree is developed. The concept of causal tree allows to represent the refinement of interoperability requirements by a causal relation linking an abstract requirement to more precise requirements with logical function. Thus, the causal tree allows with the refinement to reduce the residual ambiguity of requirements that may exist for each level as shown in figure 2 and presented in [15].

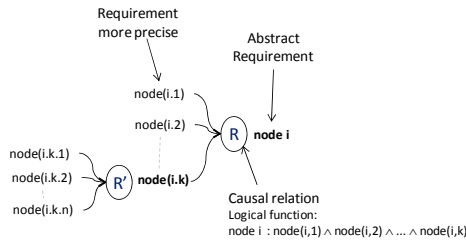


Fig. 2. Requirements refinement structured in a causal tree

This research work proposes a causal tree to represent refinement of interoperability requirements. The first node represents the first interoperability requirement that means enterprises aim to be interoperable. This first requirement can be refined by the three categories of interoperability requirements: compatibility, interoperation and reversibility. Requirements of each category must be refined by concepts related to the INTEROP framework (interoperability concerns and interoperability barriers) [10]. Therefore, each category can be refined by interoperability concerns which are refined by interoperability barriers as shown in figure 3.



Fig. 3. The Interoperability requirements causal tree

The definitions and concepts related to the three categories of interoperability requirements are presented in following sections.

3.1. Compatibility Requirements

A compatibility requirement is defined as “a *statement that specifies a function, ability or a characteristic, independent of time and related to interoperability barriers for each interoperability concerns, that enterprise must satisfy before collaboration effectiveness*”.

Compatibility requirements are derived from the enterprise interoperability framework. They are concerned by the compatibility of interoperability barriers which includes: conceptual (syntax and semantic), technological (platform, communication protocol) and organisational barrier (persons, organisational structure) at each level of enterprise. It is to note that compatibility requirements remain independent of time and have a static aspect.

Thus, a set of compatibility requirements is established taking into consideration these concepts. The following figure represents a set of compatibility requirements structured in the causal tree.

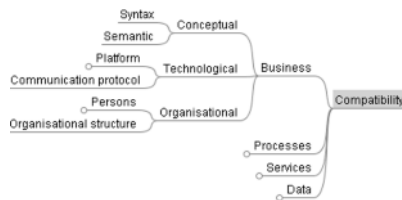


Fig. 4 Compatibility requirements structured into a causal tree

An example of compatibility requirement at the level of services and related to organizational barrier is described as: “*Services must have a clear and defined responsible*”, in order to know who is responsible for doing what.

To help enterprises to see where their compatibility problems are, verification of compatibility requirements can be performed. From this verification, enterprise will be informed of their compatibility problems in agreement with the three barriers of interoperability and the four levels where interoperability take place in enterprises. After, enterprises will have the opportunity to solve their problems.

3.2. Interoperation Requirements

An interoperation requirement is defined as “a *statement that specifies a function, ability or a characteristic, dependent of time and related to the performance of the interaction, that enterprise must satisfy during the collaboration*”.

In the runtime phase of collaboration, enterprises want to know if the interoperation is efficient. The performance of the interoperation can be measured by some performance criteria such as: the cost, the quality and the time of the interoperation as mentioned in [8]. For instance, time criterion can be related to the duration of the interoperation and the quality criterion can be related to the quantity received. It is to note that interoperation requirements are dependent of time and have a dynamic aspect.

As a consequence, a set of interoperation requirements is established according to these concepts and structured in the causal tree as shown in the following figure.



Fig. 5. Interoperation requirements structured into a causal tree

An example of interoperation requirements at the level of data related to technological barrier for quality performance can be described as: “*for each data received, a receipt must be returned*” to be sure of the good reception of data.

To help enterprises to see if the interoperation is efficient, verification of interoperation requirements can be performed. From this verification, enterprise will be informed of their interoperation problems in according to their performance criteria. After, enterprises will have the opportunity to solve their interoperation problems.

3.3. Reversibility Requirements

A reversibility requirement is defined as “*a statement that specify functions, abilities or characteristics related to the capacity of enterprises to retrieve their autonomy and to return to their original state (in terms of their own performance) that they must satisfy after collaboration*”.

During collaboration, enterprises working together are involved in a collaborative public process. This public process can implicate a part of (or whole) activity of an enterprise. On the one hand, enterprises must be able to ensure their own performance related to its private process and, on the other hand the performance related to the public process as shown in Fig 6. Therefore, as mentioned above, usually after collaboration, enterprises want to retrieve their original own performance. This means that, if the implementation of interoperability leads to adaptations or modifications (working methods, tools, organisational structure ...) in enterprises, they must be able to return to their original state at the end of the collaboration. Therefore, enterprises want to know if their performance after the collaboration is in accordance with their performance before the collaboration including variations (*e.g.* enterprises can save time production) which can be admitted by enterprises. In this case, enterprises are reversible and can resume their former activities.

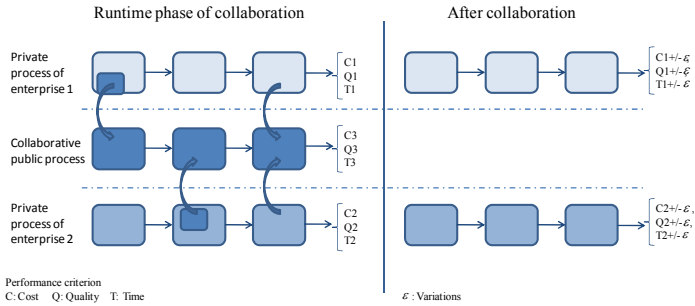


Fig. 6 Reversibility of enterprises

A set of reversibility requirements according to the performance of the enterprise measured by cost, quality and time, are structured in the causal tree as shown in the following figure.



Fig. 7. Reversibility requirements structured into a causal tree

An example of reversibility requirements at the level of service related to organisational barrier for quality criterion can be given as: *“the cost of a given service after the collaboration corresponds to the cost before collaboration including variations”*.

To help enterprises to see if they are able to retrieve their own performance, verification of reversibility requirements can be performed.

In order to illustrate the usefulness of the proposed interoperability requirements model, a case study is presented in the next section.

4. Case Study

The collaborative process issued from [14] represents the collaboration between a retailer and a provider. The sales department of the provider want to collaborate with external and independent retailers in charge of delivering products to customers. The different activities and flows between partners (retailer and provider) are shown in figure 8. The goal is to detect if compatibility, interoperability problems can occur, thanks to the interoperability requirements model.

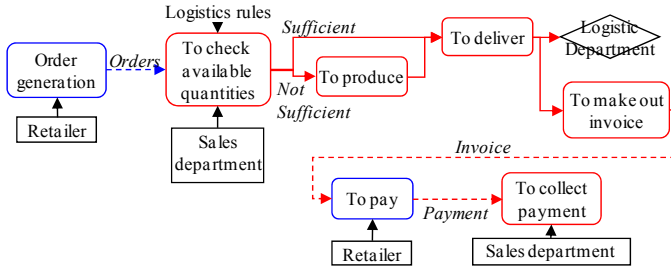


Fig. 8. Collaborative process between provider/retailer

Before the retailer sends its order, several compatibility requirements have to be verified. As far as organisational aspect is concerned, one requirement that must be verified could be: “*A service must have a clear and defined responsible*”. Indeed, if this requirement is not verified, it may dread mistakes during the transmission of orders, that could lead to a deterioration of collaboration performance (loss of time to convey the right order to the right person...).

Then, during the collaboration, it is necessary to verify some interoperation requirements. One requirement considered here can be: “*For each data received, a receipt must be returned*”. For instance, the reception of invoices has to induce, in all cases, the generation of an acknowledgement by the retailer.

When the collaboration is over, it is necessary to verify that the collaboration has not deeply impacted behaviour, organisation, and practices of partners. Thus, verification of reversibility requirements is required. A reversibility requirement considered here, could be: “*Internal functioning rules (e.g. logistics rules) of a partner P_1 must remain the same all along the collaboration and not be modified suddenly and under the control of another partner P_n , excepted if this modification are proposed and validated by P_1 itself*”. The verification of this requirement allows to detect if new (or deeply modified) logistics rules used by sales department (P_1) during the activity named ‘*to check available quantities*’ emerge independently of the sales department.

5. Conclusion

The current economic context forces enterprises to work in collaboration to improve their performances. Therefore, formalisation, structuration and verification of interoperability requirements to help enterprises to find their interoperability problems can be a solution to improve this collaboration. This paper focused on the conceptualisation of interoperability requirements. Three categories of interoperability requirements were identified and defined: compatibility, interoperation and reversibility requirements.

To help enterprises to find their interoperability problems, analysis of interoperability requirements can be performed. The provable requirements which can have a static or a dynamic aspect will be analysed by formal verification

techniques. There are two techniques related to the types of the provable interoperability requirements (static or dynamic). The first technique allows to verify static requirements as shown in [15] based on Conceptual Graphs. The second technique allows to verify dynamic requirements using model checker (Uppaal, SMV...) [16]. The requirements that are not provable will be analysed by experts. Future works are related to the formalisation of interoperability requirements into properties with a formal language to verify them later with formal verification techniques.

6. References

- [1] IEEE, (1990) A compilation of IEEE standard computer glossaries. Standard computer dictionary, New York
- [2] REGAL: Requirements Engineering Guide for All, Available on line at: <http://www.incose.org/REGAL>
- [3] Van Lamsweerde A., Dardenne A., Delcourt B., Dubisy F., (1991) The KAOS Project: Knowledge Acquisition in Automated Specification of Software. Proceedings AAAI Spring Symposium Series, Stanford University, American Association for Artificial Intelligence, pp. 59-62, March
- [4] Petit M., (2002) UEML Thematic Network - Contract n°: IST – 2001– 34229 – Work Package 1 Report - Enterprise Modelling State of the Art, October
- [5] C4ISR Architecture Working Group, (1998) «Levels of Information Systems Interoperability (LISI)». United States of America Department of Defence, Washington DC, USA, 30 March
- [6] Tolk A., Muguira J.A., (2003) « The Levels of Conceptual Interoperability Model ». Proceedings of Fall Simulation Interoperability Workshop (SIW), Orlando, USA, 2003
- [7] Clark T., Jones R., (1999) « Organisational Interoperability Maturity Model for C2 ». Proc of Command and Control Research & Techn. Symposium, Newport, USA
- [8] Daclin N., Chen D., Vallespir B., (2008) Methodology for enterprise interoperability. 17th IFAC World Congress (IFAC'08), Seoul, Korea
- [9] Blanc S., Ducq Y., Vallespir B., (2007) A graph based approach for interoperability evaluation. Interoperability for Enterprise Software and Applications, IESA 2007, Madeira, Portugal
- [10] INTEROP, (2007) Enterprise Interoperability-Framework and knowledge corpus - Final report, INTEROP NoE, FP6 – Contract n° 508011, Deliverable DI.3, May 21st
- [11] Scucanec S. J., Van Gaasbeek J. R., (2008) A day in the life of a verification requirement. U.S Air Force T&E Days, Los Angeles, California, February
- [12] Kasunic M., Anderson W., (2004) Measuring systems interoperability: challenges and opportunities. Software engineering measurement and analysis initiative, Technical note CMU/SUE-2004-TN-003
- [13] Camarinha-Matos L. M., Afsarmanesh H., Rabelo R. J., (2003) Infrastructure developments for agile virtual enterprises. Journal of Computer Integrated Manufacturing, ISSN 0951-192X, Vol. 16, N. 4-5, Jun-August
- [14] Singular Enterprise, « Implementation methodology – Blue Print – Demo TelCo », 2003
- [15] Chapurlat V., Roque M., (2009) Interoperability constraints and requirements formal modelling and checking framework. Advances in Production Management Systems, APMS 2009, Bordeaux, France
- [16] Behrmann G., David A., Larsen K. G., (2004) A tutorial on Uppaal. Department of Computer Science, Aalborg University, Denmark