Intelligent Environments 2019
A. Muñoz et al. (Eds.)

doi:10.3233/AISE190039

An Approach for Measuring IoT Interoperability Using Causal Modeling

Sebastian KOTSTEIN [a] Christian DECKER [b]
[a] *sebastian.kotstein@reutlingen-university.de*
[b] *christian.decker@reutlingen-university.de*
Hermann Hollerith Zentrum, Reutlingen University,
D-71034 Böblingen, Germany

Abstract. Interoperability is an important topic in the Internet of Things (IoT), because this domain incorporates diverse and heterogeneous objects, communication protocols and data formats. Many models and classification schemes have been proposed to make the degree of interoperability measurable - however only on the basis of a hierarchical scale. In the course of this paper we introduce a novel approach to measure the degree of interoperability using a metric scaled quantity. We consider IoT as a distributed system, where interoperable objects exchange messages with each other. Under this premise, we interpret messages as operation calls and formalize this view as a causal model. The analysis of this model enables us to quantify the interoperable behavior of communicating objects.

Keywords. Internet-of-Things, interoperability, structural causal model, measurement

1. Introduction

The Internet of Things (IoT) transforms ordinary physical objects into so-called smart objects by enriching them with computing capabilities and connecting these objects to a given communication infrastructure, namely the Internet. Functions and services that were previously only available locally (i.e. within the boundaries of the corresponding physical objects) can now be offered to other participants in the IoT environment such that new use cases and applications can be formed out of the composition of multiple services and functions. Furthermore, these smart objects produce tons of data as they are able to sense their physical environment in terms of temperature, humidity, light intensity and so further, and to share the collected data. Using actors, they are even able to manipulate the environment (e.g. a smart light switch allowing to turn on/off the lights). However, interoperability, i.e. the ability to seamlessly communicate, interact and cooperate with each other, is a critical key factor for realizing such novel IoT applications, as the market is coined by many heterogeneous objects, a huge bandwidth of available communication protocols and data formats, and numerous recommendations for possible reference architectures [1][2]. All these technologies are not always compatible to each other leading to an immense manual effort for integrating heterogeneous objects into a uniform IoT environment [1]. Therefore, the interoperability challenge is an important topic addressed by both, the research as well as in the industry [3][4][5].

However, the concept of interoperability is interpreted in various ways due to the complexity and versatility of the topic or sometimes due to a weak understanding [6]. The lack of interoperability in the context of IoT applications is often complained, but there are very few studies that addresses this issue systematically and to the full extent. Existing comprehensive studies (e.g. [3], [6] and [7]) typically address interoperability by attempting to unify as many different aspects of interoperability in one common model as possible. Furthermore, these models try to make the *degree* of interoperability measurable and comparable (i.e. a value indicating to what extent interoperability is given between two or more objects). This degree is typically measured by the fulfilment or non-fulfilment of criteria defined by the respective model: The more criteria fulfilled, the higher the degree of interoperability. Some *higher* (i.e. more advanced) criteria can only be fulfilled if *lower* criteria are fulfilled as well, thus leading to a hierarchy of different criteria and, consequently, a hierarchical scale of measurement. With our approach, we want to make interoperability measurable not only on a hierarchical scale, but also with metric values. With this first approach, the focus here is not on interoperability in its entirely (i.e. with all its characteristics, aspects and criteria), but only on a specific aspect, namely the call of operations by message passing which is ubiquitous in IoT environments. More specifically, we model this specific aspect of message passing as a causal model. Nevertheless, our aspect and thus the metric measurement method can be embedded in a hierarchically oriented model, so that only this aspect is measured metrically, but the remaining aspects are still evaluated using a hierarchical scale.

This paper is organized as follows: Section 2 gives a brief overview of definitions of interoperability to provide a first insight into this complex topic. In section 3, we analyze several approaches for classifying and measuring interoperability aspects based on a hierarchical scale and select one framework in order to embed our measurement method. The causal model is derived in section 4 and analyzed in section 5. Finally, a conclusion is given in section 6.

2. Definition of Interoperability

The problem of lacking interoperability is neither limited to the area of IoT nor is dedicated to the application field of information systems exclusively. Ide and Pustejovsky, for instance, defines interoperability as *"a measure of degree to which diverse systems, organizations, and/or individuals are able to work together to achieve a common goal"*[8]. According to this definition, interoperability is present whenever there is the necessity that different participants meet and interact with each other to fulfill a common goal, no matter whether these participants are physical machines, IT systems or human actors. Moreover, this definition describes interoperability as a *measurable property* indicating how good or bad the participants can interact with each other, or in other words, to what *degree* they are able to *inter-operate*. In the context of information systems, IEEE defines interoperability as *"the ability of two or more systems or components to exchange information and to use the information that has been exchanged"*[9]. A similar definition is given by Tsilas with *"the ability of heterogeneous IT networks, applications, or components to exchange and use information, that is to 'talk to and understand' each other"*[10]. Both definitions are close to the initial generic definition, but in comparison, the participants are specified as hardware and software components and the interaction takes place by exchanging information. With these criteria in mind, we are automatically in the field of distributed computing as information is exchanged not only locally, i.e. within the boundaries of a component, but also cross-component. Since Internet of Things setups are naturally distributed systems, as the "Things" are components interacting with each other by exchanging information over the "Internet", the cited definitions are sufficient even for the area of IoT.

3. Models and Approaches for Hierarchical Interoperability Measurement

In section 2, interoperability has been identified as a measurable property indicating to what degree interoperability is given between two or more participants. The word *degree* implies that we need some type of scale or a classification approach such that different interoperability scenarios can be compared with each other and ranked. Persisteras and Tarabanis have analyzed twelve proposed classification approaches grouping together different aspects of interoperability [6]: The finding of their analysis is, that all twelve classification approaches use an evolutionary perspective, which means that the degree of interoperability can be measured by answering the question which aspects (also called *features* [6]) of interoperability are fulfilled and which not. The fulfilment of an specific aspect in turn, can be measured by the fulfilment of criteria which are defined in accordance with the aspect. Some aspects are more advanced than other (and therefore their criteria are harder to fulfil) leading to an higher, i.e. a more advanced, degree of interoperability. For instance, the fact that two participants are able to exchange symbols over a wire is a lower fulfilled aspect in comparison to the aspect, that these two participants agree on the same understanding of the meaning of exchanged data structures. These aspects and their criteria are consolidated within so-called levels of interoperability such that a model for classification typically consists of multiple interoperability levels following a strict linearity: For reaching an upper level of interoperability (i.e. a high degree of interoperability) all the aspects of the underlying levels must be fulfilled. Although, the number of interoperability levels vary from approach to approach, the aspects identified by [6] within the twelve analyzed classification approaches are often similar to each other: On the lower levels, the basic aspects of communication are addressed, such as the ability to exchange single symbols up to unstructured and structured data (e.g. in [11] and [12]), whereas the higher levels covers aspects like the semantic representation of single words, data structures up to the ability to integrate the provided services of a component into an workflow for achieving a certain goal (e.g. in [11], [13] and [14]). Although Persisteras and Tarabanis have analyzed these twelve classification approaches in 2006 and in the context of information systems in general, classification approaches and typologies for IoT interoperability that have been developed later use the same evolutionary perspective with different interoperability levels as mentioned above. Recent models and approaches for IoT can be found in [3] and [7].

Due to the scale of fulfilment or non-fulfilment of aspects and their criteria, the measurement within these models is hierarchical and not based on a metric basis. However, a long-term goal should be to measure, evaluate and compare particular interoperability scenarios not only hierarchically, but also based on a metric scale in order to achieve finer granularity. As already mentioned, our contribution is to offer such a metric measurement method. However, this method is limited to only one specific aspect, namely the call of operations by message passing, which means that only this specific aspect can be measured using a metric scale. However, this aspect can be embedded into the existing classification approaches, so that all other aspects of interoperability are still hierarchically measurable, but this one specific aspect offers a complementary metric measurement method. To embed the aspect which is covered by our metric measurement method, we use a classification framework which has been designed by Persisteras and Tarabanis in addition to the twelve approaches which they have analyzed. The Connection, Communication, Consolidation, and Collaboration Interoperability Framework (C^4IF) uses an evolutionary perspective as well. It defines the following levels of interoperability:

- **Connection:** Includes the aspects of being able to exchange signals between two information system components.
- **Communication:** The ability of components to exchange any kind of data which is conform to a predefined data format agreed by both components. This level encompasses two aspect sub levels covering the data format of single exchanged strings ([6] mentions a data format for a string with dd/mm/yyyy as an example) on the first level as well as the data format of whole structures (e.g. an JSON or XML schema) on the second level.
- **Consolidation:** Includes the aspect of being able to understand exchanged data. This requires that both components share and agree on the same meta model describing the semantic meaning of a single string and whole data structures.
- **Collaboration:** The highest level of interoperability includes the ability not only to understand the meaning of exchanged data but also to use the data in such a way that whole processes and workflows can be formed by triggering multiple functions and services. [6] describes this level as "*the ability of systems to act together*". These actions should trigger changes in the physical world.

According to Persisteras and Tarabanis, the C^4IF has been developed by transferring some well-established concepts from linguistics into the concepts of information system interoperability. More in detail, these four levels of interoperability are mapped to equivalent levels of the human communication. Persisteras and Tarabanis argue, that these concepts of linguistics are considered as common to any kind of communication scenario and can therefore be applied to all kind of interoperability scenario in the information system context [6]. For the fulfillment of our interoperability aspect of issuing an operation by sending a message, we assume that the aspects of the **Connection** as well as of the **Communication** level must be fulfilled so that an operation can be issued by sending a message. Nevertheless, it is required that the receiver of the message is able to understand its meaning in order to choose and perform the appropriate operation. Since the terminology "message passing" can be considered as equal to the terminology of "exchanging data", the aspect is placed in the **Consolidation** level.

4. Metric Interoperability Measurement Model

Our approach to measure the degree of interoperability as a metric quantity is inspired by the observation of communicating processes within the consolidation level of the C^4IF classification scheme. In the following, we will decompose this behavior in order to extract relevant components of the model.

4.1. Observation of Communicating Processes

We consider the following typical situation in the interaction of communicating processes, where a process A, the caller, issues an operation at the process B, the callee. In figure 1, the caller sends a message M to the callee, which processes the message and issues the operation accordingly. Finally, the operation returns a result M'. In this scenario, we assume a working communication and that the callee syntactically understands the messages. This enables the callee to parse the received message and dispatch it to the appropriate operation. Figure 2 displays the link between the message reception and operation via the dispatcher. At first, the dispatcher identifies the operation *op* specified in the message and maps it to the appropriate local operation. The operation starts and afterwards the result $\mathrm{R} = op(\mathrm{M})$ indicates the end of the operation, whether the operation ran successfully or failed. R may be even empty as long as we can figure out that it originates from the callee's operation *op*. In case the message specifies an unknown operation the dispatcher returns some other result U, which can be distinguished from result R. Independent from the content, result R and result U must at least identify the result's originator. Whenever a message is received by the callee, one of R or U is created. A new message M' includes this result and is returned by the callee to inform the the caller. The key observation relevant for interoperability measurement is that the callee receives a message and creates a result from which it can be detected whether an operation was issued or not. Again, the concrete result of the operation's execution as well as the information whether the operation ran successfully or failed is not relevant in this context.

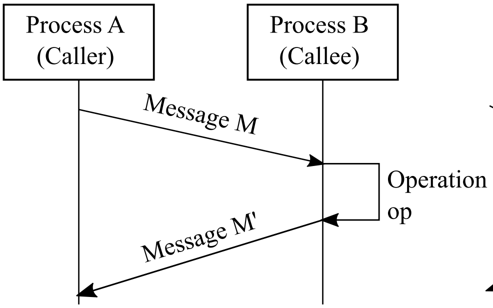

Figure 1. Message based interaction

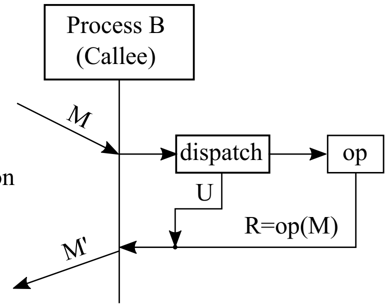

Figure 2. Message dispatch at callee

4.2. Message Passing of Interoperable Processes

We formulate the observation of the message passing behavior from Figure 2 for interoperable processes on the consolidation level of the $\mathrm{C}^4\mathrm{IF}$ scheme as follows:

Observation (Message passing)**.** *If an interoperable process receives a message M, the process's operation op will cause the process to emit a non-arbitrary outgoing message M′.*

Note that we use the term *outgoing message* as an indicator that the operation *op* was issued by the incoming message. It is not required that the outgoing message is sent back to the incoming message's originator. This pinpoints interoperability at the site of the callee process. The observation coins the essence that interoperability on the consolidation level is more than communication because it demands a dependence between message reception and message emission. This dependence is the result of some operation beyond syntactic message parsing. As consequence, it enables us to discriminate between a message-dependent behavior and an arbitrary behavior.

4.3. Causal Modeling

Our observation of the message passing behavior of interoperable processes from section 4.2 establishes a cause-effect relation. We formulate this behavior as a structural causal model (SCM) which describes in a functional form how the components interact with each other. As a result, we have clear rules for analyzing interoperable processes. A structural causal model defined by Pearl[15] is an ordered triple $< U, V, F >$ with

- U, a set of exogenous variables which are not determined by the model
- V, a set of endogenous variables which are determined by the model
- E, set of structural equations to express the values of variables in V

We define the endogenous and exogenous variables in table 1 for the SCM of interoperable processes. For each variable in table 1 we need to define its value using the structural

Component	Endogenous Variables V	Exogenous Variables U
Incoming message	M	U_M
Dispatcher	P	U_P
Outgoing message	M'	$U_{M'}$
Discriminator	D	U_D

Table 1. Causal model variable definition

equations $f_M, f_P, f_{M'}, f_D \in E$. These equations relate the variables with each other and conclude the model.

Message M. The incoming message M consists of an operation specification and corresponding arguments. The operation op is taken from set F^M of operations. We specify M using EBNF.

$$M := op,\ \{Arg\}$$

$$op := op_1 \mid op_2 \mid \ldots \mid op_k$$

, where $Arg = U_M$ are the arguments of the operation op unknown to the model and $op \in F^M$ with $|F^M| = k$.

Dispatcher P. The dispatcher parses M and maps the operation specified within to a local operation from the set F^P which are accessible by the dispatcher on the callee process. If the dispatcher cannot find the operation specified in the incoming message, it returns an unknown result.

$$P := f_P(M, U_P)$$

$$:= \begin{cases} op & \text{, if } op \in F^P \cap M \\ U_P & \text{, otherwise} \end{cases}$$

, where $U_P \notin F^P \cup F^M$. It may seem odd that the dispatcher returns an unknown result U_P. In causal modeling, we express with an unknown variable U_P that we do not explain how it is caused.

Outgoing message M'. Previously, the dispatcher P has selected the corresponding local operation according to the operation specified within the message M. The outgoing message M' contains the result of the execution of the operation op, which we define as the following equation.

$$\begin{aligned} M' &:= f_{M'}(P, M, U_{M'}) \\ &:= \begin{cases} R & \text{, if } P = op \wedge op \in M \\ U_{M'} & \text{, otherwise} \end{cases} \end{aligned}$$

, where $R = op(M)$ is the result of the operation op taking the arguments from the incoming message M. If op was not known to the dispatcher, the result is the unknown variable $U_{M'}$ which says that we do not determine its value from the causal model.

Discriminator D. Finally, the discriminator identifies whether the outgoing message M' is the operation's result or an unknown dispatcher's result.

$$\begin{aligned} D &:= f_D(M', U_D) \\ &:= \begin{cases} 1 & \text{, if } M' = R \\ 0 & \text{, otherwise} \end{cases} \end{aligned}$$

The discriminator returns 1, if M' results from the operation execution. Therewith, the discriminator establishes the dependence between incoming and outgoing message. Note that we do not need to know the concrete result of the operation execution. In this model, the operation result R only identifies its origin.

Graphical model. Finally, every SCM is associated with a graphical causal model, or simply a graph. The graph's nodes represent the variables V and U from table 1 and the edges represent the structural equation in E. The figure 3 depicts the graphical causal model of the message passing behavior of interoperable processes.

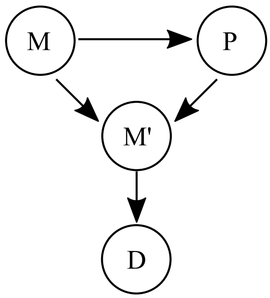

Figure 3. Graphical causal model of the message passing behavior of interoperable processes

5. Analyzing the Interoperability Measurement Model

In section 4.3 we have derived a causal model for the message passing behavior of interoperable processes. We now investigate the model behavior in order to decide on the degree of the callee's interoperability behavior. We discuss the model's behavior using probabilities, e.g. there is a probability that an incoming message M contains an operation which the dispatcher can map to a local one. The probability of the complement means that M contains an unknown operation to the dispatcher.

5.1. Variables Dependencies and d-separation

The observation of message passing of interoperable processes in section 4.2 establishes a dependency between the incoming and outgoing message through the causal relationships between the variables. We analyze the paths between the nodes in the graphical causal model of figure 3 utilizing the process of d-separation, where one can conclude the variable dependencies from. We quote the definition from Pearl's book [15], page 46.

Definition (d-separation)**.** *A path p is blocked by a set of nodes Z if and only if*

1. *p contains a chain of nodes $A \rightarrow B \rightarrow C$ or a fork $A \leftarrow B \rightarrow C$ such that the middle node B is in Z (i.e. B is conditioned on), or*
2. *p contains a collider $A \rightarrow B \leftarrow C$ such that the collision node B is not in Z, and no descendant of B is in Z.*

If Z blocks every path between two nodes X and Y are d-separated, conditional on Z, and thus are independent conditional on Z.

If two graph's nodes are d-separated, the variables they represent are independent. In contrast, if two nodes are d-connected, a path exists between them, i.e. the variables are most likely dependent.

5.2. d-separation Analysis

The analysis of d-separation in the causal graph of figure 3 let us identify the conditions for message dependencies. Concretely, the discriminator D shall determine the message M' origin as a result of the incoming message M. As a consequence, we formulate:

Problem. *Find sets of nodes in the causal graph under which M and D are d-connected or d-separated.*

Discussion of the cases, if M, D are d-connected. Using an empty conditioning set Z, then, according to the definition above, every path between M and D forms a chain with no blocking node in between. So, M and D are d-connected and therefore dependent. In this case, the incoming message M affects the probability of D. In the context of interoperability, it is understood as follows: For messages M containing operations known or unknown to the dispatcher the discriminator D yields $D = 1$ or $D = 0$ corresponding to the probability the incoming message M contains an operation known or unknown to the dispatcher. A single occurrence of message M containing an operation known to the dispatcher will yield $D = 1$ according to the defined causal relationships in section 4.3.

More formal, let M_{op} an incoming message specifying an operation the dispatcher can map to a local one, i.e. $op \in F^M \cap F^P$. Then $P(M = M_{op})$ is the probability that M_{op} is received by the callee, and $P(M = \overline{M_{op}})$ is the probability of an incoming message, which cannot be mapped by the dispatcher. Due to d-separation the discriminator always depends on the incoming message according to the causal relationships, i.e.

$$P(D = 1 \mid M = M_{op}) = P(D = 0 \mid M = \overline{M_{op}}) = 1 \tag{1}$$

$$P(D = 0 \mid M = M_{op}) = P(D = 1 \mid M = \overline{M_{op}}) = 0 \tag{2}$$

As a result, it is derived

$$P(D = 1) = \sum_{M \in \{M_{op};\overline{M_{op}}\}} P(D = 1 \mid M)P(M) = P(M = M_{op}) \tag{3}$$

$$P(D = 0) = \sum_{M \in \{M_{op};\overline{M_{op}}\}} P(D = 0 \mid M)P(M) = P(M = \overline{M_{op}}) \tag{4}$$

Discussion of the cases, if M, D are d-separated. Using $Z = \{M'\}$ will block the path between M and D, rendering D independent from M, i.e. $P(D \mid M', M) = P(D \mid M')$. Given we filter for $M' = R$ we may always observe interoperability. For $M = M_{op}$ it is the result of the causal relations . For $M = \overline{M_{op}}$, contrary to equation 4, there is an exogenous factor which changes M' to $M' = R$. A technical example could be a fail-safe operation, which catches the dispatcher's U_P result and creates a non-arbitrary response $M' = R$.

$$P(D \mid M' = R, M) = P(D \mid M' = R) = \begin{cases} 1 & \text{, for } D = 1 \\ 0 & \text{, for } D = 0 \end{cases} \tag{5}$$

For the complementary event, $P(D \mid M' \neq R)$ inverts the results from equation 5.

5.3. Degree of Interoperability

Table 2 summarizes the d-separation analysis for M and D and relates it to the degree of interoperability. Using the equations 3, 4 and 5, we can finally probabilistically measure the degree of the interoperable behavior of the callee proccess. The probability $P(D = 1)$ is the metric quantity which states how often the the callee shows an interoperable behavior under arbitrary incoming messages M.

6. Conclusion and Outlook

In this paper, we investigated the call of operations by message passing as a special aspect of interoperability. We designed a causal model formalizing this behavior. It enables us to quantify the degree of interoperability, that is how successful the dispatcher calls a process's operation according to an incoming message. It is assumed that there is an operational communication between processes. Our approach contributes to previous work on C^4IF by Persisteras and Tarabanis; it can be embedded into the Consolidation interoperability level of their framework.

Conditioning set	Statement about M, D	Degree of Interoperability, $P(D)$
$Z = \emptyset$	d-connected, i.e. M, D are dependent, see equations 3, 4	$P(D=1) = P(M = M_{op})$
$Z = \{P\}$	d-connected, i.e. conditioning on the dispatcher keeps M, D dependent, see equations 3, 4	$P(D=1) = P(M = M_{op})$
$Z = \{M'\}$	d-separated, M, D are independent, M' determines interoperability, see equation 5	$P(D=1 \mid M' = R, M) = 1$
$Z = \{P, M'\}$	d-separated, i.e. conditioning on the dispatcher has no effect, if $M' \in Z$. M and D remain independent, see equation 5	$P(D=1 \mid M' = R, M) = 1$

Table 2. Summary of d-separation analysis to measure the degree of interoperability

As a next step we plan to extend the relations our causal model spans across various message processing components. A promising tool to apply is the $do()$-calculus. It enables queries on causal models and allows us to quantify interoperable behavior in more detail.

References

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Communications Surveys and Tutorials*, vol. 17, pp. 2347–2376, June 2015.

[2] D. Miorandi, S. Sicari, F. D. Pellegrini, and I. Chlamtac, "Internet of things: Vision, applications and research challenges," *Ad Hoc Networks*, vol. 10, pp. 1497–1516, April 2012.

[3] M. Noura, M. Atiquzzaman, and M. Gaedke, "Interoperability in Internet of Things: Taxonomies and Open Challenges," *Mobile Networks and Applications*, July 2018.

[4] A. Gluhak, S. Krco, M. Nati, D. Pfisterer, N. Mitton, and T. Razafindralambo, "A Survey on Facilities for Experimental Internet of Things Research," *IEEE Communications Magazine*, vol. 49, pp. 58–67, November 2011.

[5] B. Ahlgren, M. Hidell, and E. C.-H. Ngai, "Internet of Things for Smart Cities: Interoperability and Open Data," *IEEE Internet Computing*, vol. 20, pp. 52–56, December 2016.

[6] V. Peristeras and K. A. Tarabanis, "The Connection, Communication, Consolidation, Collaboration Interoperability Framework (C4IF) For Information Systems Interoperability," *IBIS*, vol. 1, pp. 61–72, 2006.

[7] G. Hatzivasilis, I. G. Askoxylakis, G. Alexandris, D. Anicic, A. Bröring, V. Kulkarni, K. Fysarakis, and G. Spanoudakis, "The Interoperability of Things: Interoperable solutions as an enabler for IoT and Web 3.0," in *23rd IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks, CAMAD 2018, Barcelona, Spain*, pp. 1–7, September 2018.

[8] N. Ide and J. Pustejovsky, "What Does Interoperability Mean , Anyway ? Toward an Operational Definition of Interoperability for Language Technology," 2010.

[9] "IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries," *IEEE Std 610*, pp. 1–217, Jan 1991.

[10] N. L. Tsilas, "Open Innovation and Interoperability," in *Opening standards: The global politics of interoperability* (L. DeNardis, ed.), pp. 97–117, Cambridge, MA, USA: MIT Press, 2011.

[11] NATO, "NATO C3 Technical Architecture NC3TA)," tech. rep., 2003.

[12] D. P. Brutzman and A. Tolk, "JSB composability and web services interoperability via extensible modeling & simulation framework (XMSF) and model driven architecture (MDA)," *Proceedings of SPIE - The International Society for Optical Engineering*, 08 2004.

[13] IDABC, "European Interoperability Framework," tech. rep., 2004.

[14] B. Medjahed, *Semantic Web Enabled Composition of Web Services*. PhD thesis, Virginia Polytechnic Institute, Falls Church, 2004.

[15] J. Pearl, M. Glymour, and N. Jewell, *Causal Inference in Statistics: A Primer*. Wiley, 2016.