On Semantic Interoperability and the Flow of Information

Marco Schorlemmer[1] Yannis Kalfoglou[2]

[1]School of Informatics, The University of Edinburgh.
[1]Escola Universitària de Tecnologies d'Informació i Comunicació, Universitat Internacional de Catalunya.
[2]School of Electronics and Computer Science, University of Southampton.
marco@cir.unica.edu; y.kalfoglou@ecs.soton.ac.uk

Abstract. We discuss current approaches that, for the sake of automation, provide formal treatments to the problem of semantic interoperability and integration, and we reflect upon the suitability of the Barwise-Seligman theory of information flow as a candidate for a theoretical framework that favours the analysis and implementation of semantic interoperability scenarios.

1 Introduction

In a large-scale, distributed, and often deregulated environment such as the World Wide Web, systems integration is seen as the viable solution in order to cross organisational and market boundaries and hence enable applications deployment in a wide variety of domains, ranging from e-commerce to e-Science Grid projects. Although systems integration has been studied and applied for years in closed and controlled environments within organisational boundaries and vertical market segments, the situation is quite different in the emergent Semantic Web [17].

One of the ambitious goals of the Semantic Web is for systems to be able to exchange information and services with one another in semantically rich and sound ways [4]. The semantics, being a key aspect of the Semantic Web, should therefore be exposed, interpreted, and used to enable services and to support distributed applications. This means that semantics should be understood, verified against an agreed standard, and used to endorse and validate reliable information exchange. These high-level goals were similar to those pursued within the context of database schema and information integration, where the problem of semantic heterogeneity among different data sources had to be tackled [21, 9]. If these goals were achieved, two systems could be interoperable, moreover, semantically interoperable.

Semantic Interoperability and Integration

Semantic interoperability and semantic integration are much contested and fuzzy concepts which have been used over the past decade in a variety of contexts and works. As reported in [17], in addition, both terms are often used indistinctly, and some view these as the same thing.

The ISO/IEC 2382 Information Technology Vocabulary defines interoperability as "the capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units." In a debate on the mailing list of the IEEE Standard Upper Ontology working group, a more formal approach to semantic interoperability was advocated: Use logic in order to guarantee that after data were transmitted from a sender system to a receiver, all implications made by one system had to hold and be provable by the other, and there should be a logical equivalence between those implications.[1]

With respect to integration, Uschold and Grüninger argue that "two agents are semantically integrated if they can successfully communicate with each other" and that "successful exchange of information means that the agents understand each other and there is guaranteed accuracy" [25]. According to Sowa, to integrate two ontologies means to derive a new ontology that facilitates interoperability between systems based on the original ontologies, and he distinguishes three levels of integration [22]: *Alignment*—a mapping of concepts and relations to indicate equivalence—, *partial compatibility*—an alignment that supports equivalent inferences and computations on equivalent concepts and relations—, and *unification*—a one-to-one alignment of all concepts and relations that allows any inference or computation expressed in one ontology to be mapped to an equivalent inference or computation in the other ontology.

Although these definitions of semantic interoperability and integration are by no means exhaustive, and despite the blurred distinction between these two concepts, they are indicative of two trends: on one hand, we have deliberately abstract and rather ambiguous definitions of what semantic interoperability and integration could potentially achieve, but not how to achieve it; and on the other hand, we have formal and mathematically rigorous approaches, which allow for the automatisation of the process of establishing semantic interoperability and integration.

[1]Message thread on the SUO mailing list initiated at http://suo.ieee.org/email/msg07542.html

2 Formal Approaches to Semantic Interoperability

The above definitions also reveal a common denominator, that of *communication*. For two systems to interoperate there must be an established form of communication and the right means to achieve this efficiently and effectively. To provide the means for the former, practitioners have been studying and applying consensual formal representations of domains, like ontologies; these act as the protocol to which systems have to agree upon in order to establish interoperability. However, there is an ongoing debate with regard to the later means. The argument goes like that: *Having established a protocol to which communication will be based, i.e., ontologies, what is the best way to effectively make those semantically interoperable and to integrate them?*

A practical angle of viewing this problem is when we focus on the notion of equivalence. That is, we would like to establish some sort of correspondence between the systems, and subsequently their ontologies, to make them interoperable and that could be done by reasoning about equivalent constructs of the two ontologies. However, equivalence is not a formally and consensually agreed term, neither do we have mechanisms for doing that. Hence, if we are to provide a formal, language-independent mechanism of semantic interoperability and integration, we need to use some formal notion of equivalence. And for a precise approximation to equivalence the obvious place to look at is Logic.

In this sense first-order logic seems the natural choice: Among all logics it has a special status due to its expressive power, its natural deductive systems, and its intuitive model theory based on sets. In first-order logic, equivalence is approximated via the precise model-theoretic concept of *first-order equivalence*. This is the usual approach to formal semantic interoperability and integration; see e.g., [3, 5, 16, 25]. In Ciocoiu and Nau's treatment of the translation problem between knowledge sources that have been written in different knowledge representation languages, semantics is specified by means of a common ontology that is expressive enough to interpret the concepts in all agents' ontologies [5]. In that scenario, two concepts are equivalent if, and only if, they share exactly the same subclass of first-order models of the common ontology.

But this approach has its drawbacks. First, such formal notion of equivalence requires the entire machinery of first-order model theory, which includes set theory, first-order structures, interpretation, and satisfaction. This appears to be heavyweight for certain interoperability scenarios. Madhavan et al. define the semantics in terms of instances in the domain [14]. This is also the case, for example, in Stumme and Maedche's ontology merging method, FCA-Merge [23], where the semantics of a concept symbol is captured through the instances classified to that symbol. These instances are documents, and a document is classified to a concept symbol if it contains a reference that is relevant to the concept.[2] For FCA-Merge, two concepts are considered equivalent if, and only if, they classify exactly the same set of documents.

Menzel makes similar objections to the use of first-order equivalence and proposes an axiomatic approach instead, inspired on property theory [24], where entailment and equivalence are not model-theoretically defined, but axiomatised in a logical language for ontology theory [15].

Second, since model-theory does not provide proof mechanisms for checking model equivalence, this has to be done indirectly via the theories that specify the models. This assumes that the logical theories captured in the ontologies are complete descriptions of the intended models (Uschold and Grüninger call these *verified ontologies* [25]), which will seldom be the case in practice.

Furthermore, Corrêa da Silva at al. have shown situations in which even a common verified ontology is not enough, for example when a knowledge base whose inference engine is based on linear logic poses a query to a knowledge base with the same ontology, but whose inference engine is based on relevance logic [6]. The former should not accept answers as valid if the inference carried out in order to answer the query was using the contraction inference rule, which is not allowed in linear logic. Here, two concepts will be equivalent if, and only if, we can infer exactly the same set of consequences on their distinct inference engines.

Last, but certainly not least, first-order model theory was originally devised for mathematics in order to precisely describe the mathematical concepts of *truth* and *proof*. This semantics proved ill-suited for tackling problems which lay outside the scope of the mathematical realm, such as common-sense reasoning, natural language processing, or planning. Since the early days of AI, the community has been exploring several extensions of first-order logic in order to overcome these shortcomings [8].

But in spite of despising a model-theoretic approach to semantic interoperability, we want to step back and reflect on the necessity of settling upon a particular understanding of semantics for the sake of formalising and automating semantic interoperability. A careful look at the several formal approaches to semantic integration mentioned above reveals many different understandings of semantics depending on the interoperability scenario under consideration. Hence, what we need in order to successfully tackle the problem of semantic interoperability is not so much a framework that establishes a particular semantic perspective (model-theoretic, property-theoretic, instance-based, etc.), but instead we need a framework that successfully captures semantic interoperability despite the different treatments of semantics.

[2] This is done by means of a linguistic pre-analysis.

An Information-Centred Approach

In this paper we observe that, in order for two systems to be semantically interoperable (or semantically integrated) we need to align and map their respective ontologies such that *the information can flow*. Consequently, we believe that a satisfactory formalisation of semantic interoperability can be built upon a mathematical theory capable of describing under which circumstances information flow occurs.

Although there is no such theory yet, the most promising effort was initiated by Barwise and Perry with situation semantics [1], which was further developed by Devlin into a theory of information [7]. Barwise and Seligman's channel theory is currently the latest stage of this endeavour [2], in which they propose a mathematical model that aims at establishing the laws that govern the flow of information. It is a general model that attempts to describe the information flow in any kind of distributed system, ranging form actual physical systems like a flashlight connecting a bulb to a switch and a battery, to abstract systems such as a mathematical proof connecting premises and hypothesis with inference steps and conclusions. Barwise and Seligman's theory is therefore a good place to start establishing a foundation for formalising semantic interoperability.

In channel theory, each component of a distributed systems is represented by an *IF classification* $\mathbf{A} = (tok(\mathbf{A}), typ(\mathbf{A}), \models_{\mathbf{A}})$, consisting of a set of *tokens* $tok(\mathbf{A})$, a set of *types* $typ(\mathbf{A})$ and a *classification relation* $\models_{\mathbf{A}} \subseteq tok(\mathbf{A}) \times typ(\mathbf{A})$ that classifies tokens to types.[3] It is a very simple mathematical structure that effectively captures the local syntax and semantics of a community for the purpose of semantic interoperability.

For the problem of semantic interoperability that concerns us here the components of the distributed systems are the ontologies of the communities that desire to communicate. We model them as IF classification, such that the syntactic expressions that a community uses to communicate constitute the types of the IF classification, and the meaning that these expressions take within the context of the community are represented by the way tokens are classified to types. Hence, *the semantics is characterised by what we choose to be the tokens of the IF classification*, and depending on the particular semantic interoperability scenario we want to model, types, tokens, and its classification relation will vary. For example, in FCA-Merge [23], types are concept symbols and tokens particular documents, while in Ciocoiu and Nau's scenario [5] types are expressions of knowledge representation languages and tokens are first-order structures. The crucial point is that *the semantics of the interoperability scenario crucially depends on our choice of types, tokens and their classification relation for each community*.

[3] We are using the prefix 'IF' (information flow) in front of some channel-theoretic constructions to distinguish them from their usual meaning.

The flow of information between components in a distributed system is modelled in channel theory by the way the various IF classifications that represent the vocabulary and context of each component are connected with each other through *infomorphisms*. An infomorphism $f = \langle f^{\wedge}, f^{\vee} \rangle : \mathbf{A} \rightleftarrows \mathbf{B}$ from IF classifications $\mathbf{A}$ to $\mathbf{B}$ is a contravariant pair of functions $f^{\wedge} : typ(\mathbf{A}) \rightarrow typ(\mathbf{B})$ and $f^{\vee} : tok(\mathbf{B}) \rightarrow tok(\mathbf{A})$ satisfying the following fundamental property, for each type $\alpha \in typ(\mathbf{A})$ and token $b \in tok(\mathbf{B})$:

$$\begin{tikzcd}
\alpha \arrow[r, mapsto, "f^{\wedge}"] \arrow[d, dash, "\models_{\mathbf{A}}"'] & f^{\wedge}(\alpha) \arrow[d, dash, "\models_{\mathbf{B}}"] \\
f^{\vee}(b) & b \arrow[l, mapsto, "f^{\vee}"]
\end{tikzcd}$$

$$f^{\vee}(b) \models_{\mathbf{A}} \alpha \quad \text{iff} \quad b \models_{\mathbf{B}} f^{\wedge}(\alpha)$$

A *distributed IF system* $\mathcal{A}$ consists then of an indexed family $cla(\mathcal{A}) = \{\mathbf{A}_i\}_{i \in I}$ of IF classifications together with a set $inf(\mathcal{A})$ of infomorphisms all having both domain and codomain in $cla(\mathcal{A})$.

A basic construct of channel theory is that of an *IF channel*—two IF classifications $\mathbf{A}$ and $\mathbf{B}$ connected through a core IF classification $\mathbf{C}$ via two infomorphisms f and g:

$$\begin{tikzcd}
 & typ(\mathbf{C}) \arrow[dd, dash, "\models_{\mathbf{C}}"] & \\
typ(\mathbf{A}) \arrow[ur, "f^{\wedge}"] \arrow[dd, dash, "\models_{\mathbf{A}}"'] & & typ(\mathbf{B}) \arrow[ul, "g^{\wedge}"'] \arrow[dd, dash, "\models_{\mathbf{B}}"] \\
 & tok(\mathbf{C}) \arrow[dl, "f^{\vee}"] \arrow[dr, "g^{\vee}"'] & \\
tok(\mathbf{A}) & & typ(\mathbf{B})
\end{tikzcd}$$

This basic construct captures the information flow between components $\mathbf{A}$ and $\mathbf{B}$. Crucial in Barwise and Seligman's model is that it is the particular tokens that carry information and that information flow crucially involves both types and tokens.

In fact, as we shall see next, our approach uses this model to approximate the intuitive notion of equivalence necessary for achieving semantic interoperability with the precise notion of a type equivalence that is supported by the connection of tokens from $\mathbf{A}$ with tokens from $\mathbf{B}$ through the tokens of the core IF classification $\mathbf{C}$. This provides us with the general framework of semantic interoperability we are after, one that accommodates different understandings of semantics—depending on the particularities of the interoperability scenario—whilst retaining the core aspect that will allow communication among communities: a connection through their semantic tokens.

3 Semantic Interoperability via Information Channels

The key channel-theoretic construct we are going to exploit in order to outline our formal framework for semantic interoperability is that of a *distributed IF logic*. This is the logic that represents the information flow occurring in a distributed system. In particular we will be interested in a restriction of this logic to the language of those communities we are attempting to integrate. As we proceed, we will hint at the intuitions lying behind the channel-theoretical notions we are going to use; for a more in-depth understanding of channel theory we point the interested reader to [2].

IF Theory and Logic

Suppose two communities $\mathbf{A}_1$ and $\mathbf{A}_2$ need to interoperate, but are using different ontologies in different ontologies. To have them semantically interoperating will mean to know the semantic relationship in which they stand to each other. In terms of the channel-theoretic context, this means to know an *IF theory* that describes how the different types from $\mathbf{A}_1$ and $\mathbf{A}_2$ are logically related to each other.

Channel theory has been developed based on the understanding that information flow results from regularities in a distributed system, and that it is by virtue of regularities among the connections that information of some components of a system carries information of other components. These regularities are implicit in the representation of the systems' components and its connections as IF classifications and infomorphisms, but, in order to derive a notion of equivalence on the type-level of the system we need to capture this regularity in a logical fashion. This is achieved with IF theories and IF logics in channel theory.

An *IF theory* $T = \langle typ(T), \vdash\rangle$ consists of a set $typ(T)$ of types, and a binary relation $\vdash$ between subsets of $typ(T)$. Pairs $\langle\Gamma, \Delta\rangle$ of subsets of $typ(T)$ are called *sequents*. If $\Gamma \vdash \Delta$, for $\Gamma, \Delta \subseteq typ(T)$, then the sequent $\Gamma \vdash \Delta$ is called a *constraint*. T is *regular* if for all $\alpha \in typ(T)$ and all sets $\Gamma, \Gamma', \Delta, \Delta', \Sigma', \Sigma_0, \Sigma_1$ of types:

1. *Identity:* $\alpha \vdash \alpha$
2. *Weakening:* If $\Gamma \vdash \Delta$, then $\Gamma, \Gamma' \vdash \Delta, \Delta'$
3. *Global Cut:* If $\Gamma, \Sigma_0 \vdash \Delta, \Sigma_1$ for each partition $\langle\Sigma_0, \Sigma_1\rangle$ of Σ', then $\Gamma \vdash \Delta$.[4]

Regularity arises from the observation that, given any classification of tokens to types, the set of all sequents that are satisfied[5] by all tokens always fulfil these three properties. In addition, given a regular IF theory T we can generate a classification $Cla(T)$ that captures the regularity specified in its constraints. Its tokens are partitions $\langle\Gamma, \Delta\rangle$ of $typ(T)$ that are *not* constraints of T, and types are the types of T, such that $\langle\Gamma, \Delta\rangle \models_{Cla(T)} \alpha$ iff $\alpha \in \Gamma$.[6]

[4] A partition of Σ' is a pair $\langle\Sigma_0, \Sigma_1\rangle$ of subsets of Σ', such that $\Sigma_0 \cup \Sigma_1 = \Sigma'$ and $\Sigma_0 \cap \Sigma_1 = \emptyset$; Σ_0 and Σ_1 may themselves be empty (hence it is actually a quasi-partition).

[5] Defined further below.

[6] These tokens may not seem obvious, but these sequents code the content of the classification table: The left-hand sides of the these sequents indicate which to which types they are classified, while the right-hand sides indicate to which they are not.

The IF theory we are after in order to capture the semantic interoperability between communities $\mathbf{A}_1$ and $\mathbf{A}_2$ is an IF theory on the union of types $typ(\mathbf{A}_1) \cup typ(\mathbf{A}_2)$ that respects the local IF classification systems of each community—the meaning each community attaches to its expressions—but also interrelates types whenever there is a similar semantic pattern, i.e., a similar way communities classify related tokens. That is the type language we speak in a semantic interoperability scenario, because we want to know when type α of one component corresponds to a type β of another component. In such an IF theory a sequent like $\alpha \vdash \beta$, with $\alpha \in typ(\mathbf{A}_1)$ and $\beta \in typ(\mathbf{A}_2)$, would represent an implication of types among communities that is in accordance to how the tokens of different communities are connected between each other. Hence, a constraint $\alpha \vdash \beta$ will represent that every α is a β, together with a constraint $\beta \vdash \alpha$ we obtain type equivalence.

Putting the idea of an IF classification with that of an IF theory together we get an *IF logic* $\mathfrak{L} = \langle tok(\mathfrak{L}), typ(\mathfrak{L}), \models_{\mathfrak{L}}, \vdash_{\mathfrak{L}}, N_{\mathfrak{L}}\rangle$. It consists of an IF classification $cla(\mathfrak{L}) = \langle tok(\mathfrak{L}), typ(\mathfrak{L}), \models_{\mathfrak{L}}\rangle$, a regular IF theory $th(\mathfrak{L}) = \langle typ(\mathfrak{L}), \vdash_{\mathfrak{L}}\rangle$ and a subset of $N_{\mathfrak{L}} \subseteq tok(\mathfrak{L})$ of *normal tokens*, which satisfy all the constraints of $th(\mathfrak{L})$; a token $a \in tok(\mathfrak{L})$ satisfies a constraint $\Gamma \vdash \Delta$ of $th(\mathfrak{L})$ if, when a is of all types in Γ, a is of some type in Δ. An IF logic $\mathfrak{L}$ is *sound* if $N_{\mathfrak{L}} = tok(\mathfrak{L})$.

Distributed IF Logic

The sought after IF theory is the IF theory of the distributed IF logic of an IF channel

$$\begin{tikzcd} & \mathbf{C} & \\ \mathbf{A}_1 \arrow[ur, "f_1"] & & \mathbf{A}_2 \arrow[ul, "f_2"'] \end{tikzcd}$$

that represents the information flow between $\mathbf{A}_1$ and $\mathbf{A}_2$. This channel can either be stated directly, or indirectly by some sort of partial alignment of $\mathbf{A}_1$ and $\mathbf{A}_2$.

The logic we are after is the one we get from *moving* a logic on the core $\mathbf{C}$ of the channel to the sum of components $\mathbf{A}_1 + \mathbf{A}_2$: The IF theory will be induced at the core of the channel; this is crucial. The distributed IF logic is the *inverse image* of the IF logic at the core.

Given an infomorphism $f : \mathbf{A} \rightleftarrows \mathbf{B}$ and an IF logic $\mathfrak{L}$ on $\mathbf{B}$, the *inverse image* $f^{-1}[\mathfrak{L}]$ of $\mathfrak{L}$ under f is the IF logic on $\mathbf{A}$, whose theory is such that $\Gamma \vdash \Delta$ is a constraint of $th(f^{-1}[\mathfrak{L}])$ iff $f\hat{}[\Gamma] \vdash f\hat{}[\Delta]$ is a constraint of $th(\mathfrak{L})$, and whose normal tokens are $N_{f^{-1}[\mathfrak{L}]} = \{a \in tok(\mathbf{A}) \mid a = f\check{}(b) \text{ for some } b \in N_{\mathfrak{L}}\}$. If $f\check{}$ is surjective on tokens and $\mathfrak{L}$ is sound, then $f^{-1}[\mathfrak{L}]$ is sound.

The type and tokens system at the core and the IF classification of tokens to types will determine the IF logic at this core. We usually take the *natural IF logic* as the IF logic of the core, which is the IF logic $Log(\mathbf{C})$ generated from an IF classification $\mathbf{C}$, and has as classification $\mathbf{C}$, as regular theory the theory whose constraints are the sequents satisfied by all tokens, and whose tokens are all normal. This seems natural, and is also what happens in the various interoperability scenarios we have been investigating.

Given an IF channel $\mathcal{C} = \{f_{1,2} : \mathbf{A}_{1,2} \rightleftarrows \mathbf{C}\}$ and an IF logic $\mathfrak{L}$ on its core $\mathbf{C}$, the *distributed IF logic* $DLog_{\mathcal{C}}(\mathfrak{L})$ is the inverse image of $\mathfrak{L}$ under the sum infomorphisms $f_1 + f_2 : \mathbf{A}_1 + \mathbf{A}_2 \rightleftarrows \mathbf{C}$. This sum is defined as follows: $\mathbf{A}_1+\mathbf{A}_2$ has as set of tokens the Cartesian product of $tok(\mathbf{A}_1)$ and $tok(\mathbf{A}_2)$ and as set of types the disjoint union of $typ(\mathbf{A}_1)$ and $typ(\mathbf{A}_2)$, such that for $\alpha \in typ(\mathbf{A}_1)$ and $\beta \in typ(\mathbf{A}_2)$, $\langle a, b\rangle \models_{\mathbf{A}_1+\mathbf{A}_2} \alpha$ iff $a \models_{\mathbf{A}_1} \alpha$, and $\langle a, b\rangle \models_{\mathbf{A}_1+\mathbf{A}_2} \beta$ iff $b \models_{\mathbf{A}_2} \beta$. Given two infomorphisms $f_{1,2} : \mathbf{A}_{1,2} \rightleftarrows \mathbf{C}$, the sum $f_1 + f_2 : \mathbf{A}_1 + \mathbf{A}_2 \rightleftarrows \mathbf{C}$ is defined by $(f_1 + f_2)\hat{}(\alpha) = f_i(\alpha)$ if $\alpha \in \mathbf{A}_i$ and $(f_1 + f_2)\check{}(c) = \langle f\check{}_1(c), f\check{}_2(c)\rangle$, for $c \in tok(\mathbf{C})$.

It is interesting to note that since the distributed IF logic is an inverse image, soundness is not guaranteed [2], which means that the semantic interoperability is not reliable in general. Even if $\alpha \dashv\vdash \beta$ in the IF logic, there might be tokens (instances, situations, models, possible worlds) of the respective components for which this is not the case. Reliable information flow is only achieved for tokens that are connected through the core. The way in which infomorphisms from components to the core are defined in an interoperability scenario is crucial. If these infomorphisms are surjective on tokens, then the distributed IF logic will preserve the soundness of the IF logic of the core. Proving the token-surjectiveness is hence a necessary task in order to guarantee reliable semantic interoperability.

In this sense, in Stumme and Maedche's FCA-Merge scenario [23] reliable semantic interoperability is achieved, because tokens are shared among communities, and hence all infomorphisms have the identity as its token-level function, which is obviously surjective. But this is not the case in Ciocoiu and Nau's treatment of knowledge source translation [5], where reliable semantic interoperability is only achieved when sticking to first-order models of the common ontology, which play the role of tokens of the core of an IF channel, that connect the models of the various knowledge sources.

Four Steps Towards Semantic Interoperability

To summarise, in order to achieve the semantic interoperability we desire, for each scenario we will need to go through the following four steps:

1. We define the various contexts of each community by means of a distributed IF system of IF classifications.
2. We define an IF channel—its core and infomorphisms—connecting the IF classifications of the various communities.
3. We define an IF logic on the core IF classification of the IF channel that represents the information flow between communities.
4. We distribute the IF logic to the sum of community IF classifications to obtain the IF theory that describes the desired semantic interoperability.

These steps illustrate a theoretical framework and need not to correspond to actual engineering steps; but, since any effort to automatise semantic interoperability will need to be based to some extend on a formal theory of semantic interoperability, we claim that a sensible implementation of semantic interoperability can be achieved following this framework. In the next section we describe how this information-centred approach has been applied to various realistic interoperability scenarios.

4 Explorations and Applications

A significant effort to develop an information-centred framework around the issues of organising and relating ontologies is Kent's Information Flow Framework (IFF) [11, 13]. IFF uses channel theory in that it exploits the central distinction between types and tokens, in order to formally describe the stability and dynamism of conceptual knowledge organisation. Kent also describes a theoretical two-step process that determines the *core ontology of community connections* capturing the organisation of conceptual knowledge across communities. The process starts from the assumption that the *common generic ontology* is specified as an IF theory and that the several *participating community ontologies* extend the *common generic ontology* according to theory interpretations, and consists of the following steps: A *lifting step* from IF theories to IF logics that incorporates instances into the picture (proper instances for the community ontologies, and so called *formal instances* for the generic ontology); a *fusion step* where the IF logics of community ontologies are linked through a *core ontology of community connections*, which depends on how instances are linked through the concepts of the common generic ontology. IFF is currently further developed by the IEEE Standard Upper Ontology working group as a meta-level foundation for the development of upper ontologies [12].

Very close in spirit and in the mathematical foundations of IFF, Schorlemmer studied the intrinsic duality of channel-theoretic constructions, and gave a precise formalisation to the notions of *knowledge sharing scenario* and *knowledge sharing system* [19]. He used the categorical constructions of Chu spaces [18] in order to precisely pin down some of the reasons why ontologies turn out to be insufficient in certain scenarios where a common verified ontology is not enough for knowledge sharing [6]. His central argument is that formal analysis of knowledge sharing and ontology mapping has to take a duality between syntactic types (concept names, logical sentences, logical sequents) and particular situations (instances, models, semantics of inference rules) into account.

With respect to the particular task of mapping ontologies Kalfoglou and Schorlemmer present a step-wise method, IF-Map, which (semi-)automatically maps ontologies based on representing ontologies as IF classifications and automatically generating infomorphisms between IF classifications [10]. They demonstrated their approach by using the IF-Map method to map ontologies in the domain of computer science departments from five UK universities. The underlying philosophy of IF-Map follows the assumptions made in Section 3 where we argued that the way communities classify their instances with respect to local types reveals the semantics which could be used to guide the mapping process. Their method is also complemented by harvesting mechanisms for acquiring ontologies, translators for processing different ontology representation formalisms and APIs for web-enabled access of the generated mappings; these are in the form of infomorphisms as we introduced them in Section 3.

Finally, Schorlemmer and Kalfoglou used the framework described in Section 3 in an e-government alignment scenario [20]. In particular, they used the four steps described earlier to align UK and US governmental departments by using their units as types and their respective set of responsibilities as tokens which were classified against those types. This test bed was used to demonstrate the feasibility of the framework in a versatile and emerging paradigm, that of e-governments, where semantic interoperability is a prerequisite.

5 Conclusion

In order to achieve semantic interoperability and integration in an automated fashion we will need of a formal theory of semantic interoperability that suitably captures the idea of a "semantically integrated community" [25]. So far, efforts to formalise and automatise the issues arising with semantic interoperability have been focused on particular understandings of semantics, mainly based on first-order model theory. But it would be desirable to be provided with a theoretical framework that accommodates various different understandings of semantics depending on the semantic interoperability scenario addressed.

In this paper we have explored the suitability of Barwise and Seligman's channel theory to establish such a framework, by focusing our attention on the issues of information flow. Consequently, we have proposed a four-step methodology to enable semantic interoperability and have discussed various applications, theoretical and practical, of this methodology.

Acknowledgements. This work is supported under the Advanced Knowledge Technologies (AKT) Interdisciplinary Research Collaboration (IRC), which is sponsored by the UK Engineering and Physical Sciences Research Council under grant number GR/N15764/01. The AKT IRC comprises the Universities of Aberdeen, Edinburgh, Sheffield, Southampton and the Open University.

The first author is also supported by a 'Ramón y Cajal' Fellowship from the Spanish Ministry of Science and Technology.

References

[1] J. Barwise and J. Perry. *Situations and Attitudes*. MIT Press, 1983.

[2] J. Barwise and J. Seligman. *Information Flow: The Logic of Distributed Systems*. Cambridge University Press, 1997.

[3] T. Bench-Capon and G. Malcolm. Formalising ontologies and their relations. In *Database and Expert Systems Applications*, volume 1677 of *Lecture Notes in Computer Science*, pages 250–259, 1999.

[4] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, May 2001.

[5] M. Ciocoiu and D. Nau. Ontology-based semantics. In *Principles of Knowledge Representation and Reasoning: Proceedings of the 7th International Conference*, pages 539–548. Morgan Kaufmann, 2000.

[6] F. Corrêa da Silva, W. Vasconcelos, D. Robertson, V. Brilhante, A. de Melo, M. Finger, and J. Agustí. On the insufficiency of ontologies: Problems in knowledge sharing and alternative solutions. *Knowledge-Based Systems*, 15(3):147–167, 2002.

[7] K. Devlin. *Logic and Information*. Cambridge University Press, 1991.

[8] M. Genesereth and N. Nilsson. *Logical Foundations of Artificial Intelligence*. Morgan Maufmann, 1987.

[9] A. Halevy. Answering queries using views. *The VLDB Journal*, 10:270–294, 2001.

[10] Y. Kalfoglou and M. Schorlemmer. Information-flow-based ontology mapping. In *On the Move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE*, volume 2519 of *Lecture Notes in Computer Science*, pages 1132–1151. Springer, 2002.

[11] R. Kent. The information flow foundation for conceptual knowledge organization. In *6th International Conference of the International Society for Knowledge Organization*, Toronto, Canada, July 2000.

[12] R. Kent. A KIF formalization of the IFF category theory ontology. In *IJCAI'01 Workshop of the IEEE Standard Upper Ontology*, Seattle, Washington, USA, 2001.

[13] R. Kent. The IFF foundation for ontological knowledge organization. In *Knowledge Organization and Classification in International Information Retrieval*, Cataloging and Classification Quarterly. The Haworth Press Inc., 2003.

[14] J. Madhavan, P. Bernstein, P. Domingos, and A. Halevy. Representing and reasoning about mappings between domain models. In *18th National Conferenc on Artificial Intelligence (AAAI'02)*, Edmonton, Canada, 2002.

[15] C. Menzel. Ontology theory. In *ECAI'02 Workshop on Ontologies and Semantic Interoperability*, Lyon, France, 2002.

[16] P. Mitra and G. Wiederhold. An algebra for the composition of ontologies. In *ECAI'02 Workshop on Knowledge Transformation for the Semantic Web*, Lyon, France, 2002.

[17] J. Pollock. The web services scandal: How data semantics have been overlooked in integration solutions. *eAI Journal*, pages 20–23, Aug. 2002.

[18] V. Pratt. The Stone gamut: A coordination of mathematics. In *10th Annual Symposium on Logic in Computer Science*, pages 444–454. IEEE Computer Society Press, 1995.

[19] M. Schorlemmer. Duality in knowledge sharing. In *7th International Symposium on Artificial Intelligence and Mathematics*, Ft. Lauderdale, Florida, USA, 2002.

[20] M. Schorlemmer and Y. Kalfoglou. Using information-flow theory to enable semantic interoperability. In *6è Congrés Català en Intel·ligència Artificial*, Palma de Mallorca, Spain, Oct. 2003. Also available as Informatics Report EDI-INF-RR-0161, The University of Edinburgh.

[21] A. Sheth and J. Larson. Federated database systems for managing distributed, heterogeneous, and autonomous databases. *ACM Computing Surveys*, 22(3):183–236, 1990.

[22] J. Sowa. *Knowledge Representation and Reasoning: Logical, Philosophical, and Computational Foundations*. Brooks/Cole, 2000.

[23] G. Stumme and A. Maedche. FCA-Merge: Bottom-up merging of ontologies. In *17th International Joint Conference on Artificial Intelligence (IJCAI'01)*, Seattle, Washington, USA, Aug. 2001.

[24] R. Turner. Properties, propositions and semantic theory. In M. Rosner and R. Johnson, editors, *Computational linguistics and formal semantics*, chapter 5. Cambridge University Press, 1992.

[25] M. Uschold and M. Grüninger. Creating semantically integrated communities on the world wide web. In *WWW'02 Semantic Web Workshop*, 2002.