Logic and **L**ogical **P**hilosophy
Online First Articles (2026)
DOI: 10.12775/LLP.2026.013

Timothy Tambassi

<!--Does This Affect the Interpretation of What Follows?-->

Extensible Markup Language, Comments, and Communication

Abstract. By defining the content of formal ontologies, we can determine, for the ontology, the truth value of any statement that refers to, or does not refer to, that content. This is not possible for all XML documents. The main reason for this is that, unlike ontologies, XML does not inherently embrace any principle for handling known and unknown information as a metalanguage. This allows the application of different principles to the same XML document, which generate different interpretations of it, thereby undermining the communication between people and software applications that XML, like ontologies, is supposed to support. In this paper, I propose a solution for XML documents to come close to achieving the same result as ontologies: including comments that contain principles for handling known and unknown information. In demonstrating that this solution works only under very specific circumstances, I also focus on the communication that ontologies and XML documents are intended to support, highlighting the different roles they assign to the subjects engaging with their content: spectators and interpreters, respectively. More specifically, I argue that the fact that the subjects (whether people or software applications) have ultimate authority over the interpretation of XML documents renders the documents hermeneutically unstable — an instability that even the inclusion of a comment specifying a principle for handling known and unknown information shifts rather than definitively resolves.

Keywords: philosophy of language; markup languages; communication; interpretation; human-machine interaction

Received April 19, 2026. *Accepted* July 7, 2026. *Published online* September 7, 2026

1. Formal ontologies[1] make ordinary what is often extraordinary: by defining the content of an ontology,[2] we can determine, for the ontology, not only the truth value of any statement about that content, but also of any statement that makes no reference to that content. This is made possible by the ontology language used to create the ontology. There are many ontology languages, and they may differ from one ontology to another. Each is based on principles for handling known and unknown information — that is, content that is or is not part of the ontology. In other words, in any ontology, content defines the known, while principles allow truth values to be assigned to both the known and the unknown. Since different ontology languages are based on different principles, the truth value of any statement within an ontology ultimately depends on — and varies with — the ontology's content, the ontology language, and the principles of that language. Understanding this is essential to avoid misinterpretations of such content (Tambassi, 2025). It is also the basis on which ontologies achieve one of their primary objectives: supporting communication between people and software applications (Goy and Magro, 2015).

2. Extensible Markup Language (XML) shares the same objective as any ontology (Salminen and Tompa, 2011), though not the same "extraordinariness". As a markup metalanguage designed to allow users to define their own customized languages (Attenborough, 2003), XML is well-suited for serializing ontology languages, which are inherently based on principles for handling known and unknown information. However, ontology languages are only a subset of the languages that can be defined using XML. None of them need to use XML as a serialization format, and some cannot even be expressed using XML. In other words, XML

[1] Different (and competing) definitions of formal ontologies are provided by Neches et al. (1991); Gruber (1993); Guarino and Giaretta (1995); Bernaras et al. (1996); Borst (1997); Swartout et al. (1997); Studer at al. (1998); Guarino (1998); Uschold and Jasper (1999); Noy and McGuinness (2003); Smith (2003); Sowa (2005); Gruber (2009).

[2] By "content of an ontology", I mean everything belonging to the ontology's representational primitives, i.e. its instances, classes, properties and relations. Instances are the lowest-level components, the basic units, of an ontology (Laurini, 2017). Classes, which may contain subclasses and/or be subclasses of other classes, are sets of instances that share common features (Jaziri and Gargouri, 2010). Properties describe the various features of a class and of its instances (Noy and McGuinness, 2003; Jaziri and Gargouri, 2010). Relations represent how both classes and instances interact with each other (Laurini, 2017).

itself is not an ontology language. More importantly, most of the XML languages do not inherently support any principle for handling known and unknown information (Tambassi, 2026). This makes it difficult, and often impossible, to determine the truth value of any statement about what XML documents (i.e. documents written in XML languages) do or do not represent based solely on the documents themselves (Selmer and Ferrucci, 1998). This, in turn, increases the risk of misinterpretation, which is detrimental to the communication between people and the software applications XML is intended to support (Ardagna and Bena, 2025).

3. The fact that XML does not inherently support these principles as a metalanguage is not an insurmountable obstacle to XML documents achieving similar results to those of an ontology, even if the XML documents do not serialize ontology languages. For example, such principles could be applied to the XML document externally. However, in this case, it would not be the XML documents themselves that adopt these principles. Rather, the principles would be adopted by the subjects (human or software application) processing the documents. Furthermore, the external application could be of different principles. And this would mean assigning different truth values to each statement that does or does not refer to what the XML documents represent, depending on the (potentially different) principles that the subjects apply (Siegel, 2022).

It would be different if the XML documents contained an indication of the principles to be applied. This would allow the documents to specify their own principles and improve the communication that they are supposed to support, coming close to achieving the same result as any ontology. Below, I will discuss whether this result can be achieved through a seemingly simple — though not necessarily exclusive — method of providing the indication: including a comment that contains a principle for handling known and unknown information in the XML documents.[3] Nevertheless, two remarks are in order. First, the XML documents I am considering are well-formed, adhere strictly to basic XML

[3] An alternative to the comments could be the use of XML processing instructions (PIs), possibly combined with namespaces or embedded schemas. In that scenario, for example, the comment

```
<!--The next four lines are under the closed-world assumption-->
```

might be replace by the following processing instruction syntax, and do not overlap with any ontology languages. For XML documents that do overlap with ontology languages, there is no need to come close to the same results as an ontology. Second, the XML documents I am considering can only come close to achieving the same result as an ontology — they cannot fully achieve it. This is because the XML format does not inherently contain or presuppose any reasoning properties, unlike ontologies. Therefore, even if XML documents indicated these principles internally, their effective application would still depend on the subjects processing the documents (Taira et al., 2025). Moreover, unlike ontologies, the XML documents I am considering have no intrinsic semantics (Floridi, 2005). For this reason, the semantic interpretation(s) of such documents also lies in the hands of the subjects processing them (Sperberg-McQueen and Huitfeldt, 2011; Buda and Primiero, 2024). Accordingly, and for the sake of simplicity, I will analyze some examples of such XML documents, assuming that the subjects engaging with them

---

```
<?world-processor-e6c4d8b5-9f3a-4b2e-8c1a-7d0f9a6e2b43 assumption="closed"
scope="4" unit="line" direction="next"?>
```

where the Universally Unique Identifier identifies a processor that, if available to the XML document manager, applies the declared principle; otherwise, the instruction is simply ignored. (Likewise, an XML document containing comments such as “John Doe has a son” or “John Doe’s age is 28” could be reformulated by combining a PI with an application-specific namespace:

```
<?revision-processor-f47ac10b-58cc-4372-a567-0e02b2c3d479
on-conflict="revision-overrides-original" scope="document"?>
<person gender="male" xmlns:revision="https://www.revision.info/rev1">
  <revision:add relation="parent-of" target-gender="male"/>
  <name>John Doe</name>
  <age revision:value="28">30</age>
</person>)
```

Such solution (PIs, possibly combined with namespaces or embedded schemas) would render the principle for handling known and unknown information — as included in the comments — not only machine-readable but also machine-*actionable*. The article, however, does not pursue this solution. Its purpose, indeed, is not to propose a strategy for operationalizing epistemic principles for handling known and unknown information, but rather to examine whether XML documents may improve the communication they are intended to support, coming close to achieving the same results as any ontology, by specifying their own principles by means of comments. Moreover, as we shall see below, this solution would fall prey to the very same difficulties encountered by comments, which may be read, ignored, privileged, or subordinated by different subjects, whether human or software applications. In other words, the solution would not eliminate the interpretative problem analyzed in this paper; it would merely relocate the problem.

assign to the text in the XML documents the same meaning it would have in English (accordingly, "age" means age, "person" means person, etc.). I further assume that these subjects can process, interpret, and manipulate both English and XML, and apply the principles on this basis.

4. The paper is structured as follows. Sects. 5–7 provide examples of XML documents representing the same statement, as well as different interpretations of the same document. Sects. 8–10 describe the role of comments in XML and show three different strategies that a subject can employ to deal with comments. Sects. 11–15 introduce the notion of principle for handling known and unknown information and provide some examples of comments that incorporate these principles. Sects. 16–18 complicate matters by introducing two potential conflicts: one among data, comments, and comments containing a principle for handling known and unknown information, and another between the latter comments and subjects with their own principles. Sects. 19–20 offer concluding remarks on whether using comments containing these principles allows XML to come close to achieving the same result as ontologies, and on the difference between ontologies and XML in terms of the communication they support.

This article focuses on XML for two main reasons. First, like other (meta)languages, XML is not inherently bound by any principle for handling known and unknown information. It has no constraints on the principles to be applied and allows comments to be introduced. Thus, the analysis of these pages is broadly generalizable to other such languages. Second, since XML, like many of these languages, also lacks intrinsic semantics and is explicitly intended to support communication between humans and software applications, tensions in such communication may arise. These regard the fact that the subjects (whether human or software applications) confronted with XML documents retain ultimate authority over their interpretation, which renders the documents hermeneutically unstable. This instability perpetuates the tension in communication, a tension that even the introduction of a comment with a principle for handling known and unknown information only shifts, rather than definitively resolves.

5. If we wanted to use XML to represent "John Doe is a male person, age 30", we could write the following XML document:

```
<person gender="male">
  <name>John Doe</name>
  <age>30</age>
</person>
```

In the document, the second and third lines are the child elements of the root element, which is formed by all four lines. “John Doe” and “30” are the contents that identify the data. “Person”, “name”, and “age” are the tags that label the data. “Gender” and “male” are the name and value attributes, respectively. Both attributes provide additional information about the element for which they are declared (Dykes, 2005; Canducci, 2022).

6. Of course, there are many other ways of using XML to represent “John Doe is a male person, age 30”. For example, we could write:

```
<person name="John Doe">
  <gender>male</gender>
  <age>30</age>
</person>
```

or again

```
<person>
  <name>John Doe</name>
  <gender>male</gender>
  <age>30</age>
</person>
```

Much depends on how we want to represent the statement, what we want to emphasize, and so on. In other words, it depends on the perspective we take toward the statement to be represented (Tambassi, 2024). Still, each (of these) XML documents should, at least in principle, support communication between people and software applications. This means that all such XML documents must contain information that can be read, manipulated, and exchanged both by people and software applications without misinterpretation (Zverev et al., 2024).

7. However, especially when we are dealing with information that an XML document does not represent, nothing prevents us from legitimately inferring different and conflicting things from the same XML document, thereby weakening the communication that XML is supposed to support (Tambassi, 2026). One reason for this is that XML, as a metalanguage, does not inherently support any principles for handling known and unknown information. It does not even impose restrictions on the principles to be applied, thus ensuring theoretical and practical freedom in their use. Accordingly, based on each of the previous three XML documents, a question such as “Does John Doe have a son?” may have different answers depending on the principles we choose. The fact that XML is not governed by any principle for handling known and unknown information does not prevent an XML document from providing information about the principles to be applied. As stated in Sect. 3, an easy way to achieve this is through comments, which would also allow these principles to vary from one XML document to another, as with formal ontologies.

8. Comments are not mandatory for XML, meaning that an XML document may or may not contain them. Each comment begins with <!-- and ends with -->, which define its boundaries. Any text within these boundaries is a comment, whose general purpose is to provide additional information within and/or about the XML document, as well as about the part(s) of the XML document to which the comment specifically refers. Each comment has the following syntax:

```
<!--This is an example of a comment-->
```

It can appear anywhere in an XML document except before the XML declaration, inside a tag and/or an attribute value, and within other comments. Additionally, a comment can contain any text except the following sequence: --. Here is an example of how comments can be used in an XML document:

```
<person gender="male">
  <!-- Add more information about John Doe -->
  <name>John Doe</name>
  <!--<age>30</age>-->
</person>
```

In the example, the second line contains information about the development of the document, and the fourth line contains information about John Doe’s age. As comments, both the second and fourth lines provide information without affecting and being part of the data of the XML document. Technically, they are sections of text visible in the code, but are not part of the assertoric subset.[4] (In this sense, the fourth line does not represent data; however, it would represent data if we simply deleted `<!--` and `-->`.)

9. Determining how comments relate to the data — and particularly which comments or data warrant more attention — is not so easy, especially as XML provides no guidance on this. As a result, this can vary not only depending on the subjects (people or software applications) encountering a comment, but also on their own relationships with different comments, data, documents, and contexts. Given all these variables, a list of possible strategies for dealing with comments is unlikely to be exhaustive. But this does not prevent us from trying to outline some of the most general approaches.[5] I will do so in the next section by analyzing examples of XML documents containing comments that potentially contradict the data, namely:

document₁

```
<person gender="male">
  <!--John Doe's age is 28-->
  <name>John Doe</name>
  <age>30</age>
</person>
```

document₂

```
<person gender="male">
  <!--It is a joke that John Doe's age is 30. His real age is 28-->
  <name>John Doe</name>
  <age>30</age>
</person>
```

[4] In XML documents, the code encompasses the entire document and may include data (i.e., contents, tags, and attributes), comments, metadata, and/or processing instructions. By contrast, the assertoric subset is a subset of the code that refers exclusively to data. Therefore, in XML documents that contain only data, the code and the assertoric subset are identical; in XML documents that also include comments or other non-data elements, the code and assertoric subset diverge. For the sake of argumentative simplicity, this article presents only two types of XML documents: (i) those in which the code and assertoric subset coincide (see Sects. 5–6 and 12), and (ii) those in which they differ solely due to the presence of comments (see Sects. 8–9, 13–15, and 17). In other words, when the code and assertoric subset differ, it is because the former includes both data and comments, while the latter refers strictly to data.

[5] Since XML is not inherently bound by or designed to support any of these strategies, I do not rule out the possibility that the choice of which strategy to follow reveals the subjects' epistemic attitude towards a comment.

and *document*$_3$

```
<person gender="male">
  <!--John Doe has a son-->
  <!--John Doe's age is 28-->
  <name>John Doe</name>
  <age>30</age>
</person>
```

10. *Document*$_1$, *document*$_2$, and *document*$_3$ differ in the code, not in the assertoric subset. While the former shows a potential inconsistency about John Doe's age (28 in the comments and 30 in the data), the latter does not. Therefore, for software applications such as XML parsers[6] that generally do not process comments, John Doe's age is simply 30 in every document, no matter what the comments say. Comments are not part of the actual data in such documents. However, not all software applications are XML parsers. Generative AI chatbots like ChatGPT-4o, for example, examine both the code and the assertoric subset, detect inconsistencies in the former, and suggest two different strategies for addressing them. The first strategy, specific to *document*$_1$ and *document*$_3$, involves the data taking precedence over the comments. This explains why John Doe is 30 in *document*$_1$ and *document*$_3$. But this is also why John Doe has a son in *document*$_3$: if the comments do not contradict the data, there is no reason to ignore them. Conversely, the second strategy applies to *document*$_2$, where the comment not only contradicts the data, but also says that some of the data is false. Here, ChatGPT-4o distinguishes between two different contexts: strict XML and human. In the former, John Doe' age is 30, although it is not entirely clear whether this is because the comments must be ignored or because the data takes precedence over the comments. In the latter, which ChatGPT-4o seems to suggest for *document*$_2$, John Doe's (real) age is 28. In fact, insofar as comments are pieces of information (notes) intended specifically for humans — rather than data processed by XML parsers or similar software applications — they can be seen as taking precedence over the data. And the comment in *document*$_2$ explicitly states that it is not true that John

[6] XML parsers are software applications (or components within software applications) designed to read and interpret XML documents, to provide interfaces for the user application to access those documents, and to enable other software to process and manipulate the data contained within them. Most XML parsers also check the well-formedness of XML documents and validate them against a Document Type Definition or an XML schema (Li, 2009).

Doe's age is 30, and clarifies that it is 28. I am not entirely convinced by the distinction between strict XML and human contexts. For example, I see no clear reason why humans could not simply ignore the comment in *document$_2$*. Moreover, the distinction does not seem to account for the possibility of software applications (such as ChatGPT-4o!) that do not ignore comments. In other words, I do not see why we should make distinctions based on people and software applications, when both different people and software applications can, in principle, follow any of these strategies. That said, I suggest that these considerations allow us to outline three main different general strategies for dealing with comments in XML: ignore comments, consider data to take precedence over comments, consider comments to take precedence over data.

11. Rules of inference are formal, logical principles that are used to derive conclusions from premises. Examples include modus ponens, modus tollens, disjunction elimination, etc. In contrast, rules for making inferences are metarules that specify (high-level) assumptions under which the rules of inference are applied. These metarules also include principles that provide guidelines for handling known and unknown information (Friend, 2016; Tambassi, 2026). Examples of such principles are the open-world assumption, the closed-world assumption,[7] the reality principle, the mutual belief principle,[8] and so on. Although I do not deny the interrelationship between these types of rules, in the following sections I will focus exclusively on the latter. Specifically, I will discuss one principle for handling known and unknown information, emphasizing that — as discussed in Sects. 2, 4, and 7 — XML, as a metalanguage, is neither inherently bound by nor intended to support any of these principles, nor does it impose restrictions on the principles to be applied. While the choice of the open-world assumption [OWA] among other principles is entirely coincidental, the decision to focus on only a single principle is intended to avoid redundancy.

[7] For both open and closed world assumptions, see (Baader et al., 2004; Henley, 2006; Lutz et al., 2012; Porello and Endriss, 2014; Razniewski et al., 2016; Baader et al., 2017; Li et al., 2018; Rector at al., 2019; Russell and Norvig, 2020)

[8] For classical formulations of the reality principle and the mutual belief principle, see (Lewis, 1978; Ryan, 1980; Wolterstorff, 1980; Walton, 1990). For recent discussions and critiques, see (Friend, 2016; Franzén, 2021).

12. According to the OWA, any statement that is not known to be true (or false) is assumed to be unknown. Based on the assumptions of Sect. 3, applying the OWA to the following XML document

```
<person gender="male">
  <name>John Doe</name>
  <age>30</age>
</person>
```

would imply the truth that John Doe is a male aged 30. It would also imply the falsehood that John Doe is not 30 or is not a person. Again, the application allows us to treat as unknown, for example, whether John Doe has a son and — absent the assumptions — whether John Doe's age is 28. If the latter point seems strange, we should simply note that the XML document does not specify that John Doe's age is only 30, nor does it say that his age is not 28. If we remain unconvinced, this may be because we are assuming a one-to-one correlation between people and their age, a correlation that could be considered implicit in the semantics of "age" (and thus in the assumptions), but which the XML document itself does not mention.

13. This application of the OWA is external to the XML document, meaning that it is not the document itself that is under the OWA. Rather, the document is under the OWA due to this application. In order for the XML document itself to be under the OWA, I have argued in Sects. 3 and 7 for the use of a comment. The following XML document is an example of this proposal:

```
<!--The next four lines are under the open-world assumption-->
<person gender="male">
  <name>John Doe</name>
  <age>30</age>
</person>
```

What is true, false, and unknown in the XML document corresponds to what I said in Sect. 12 about the (external) application of the OWA to an XML document. The only difference here is that, since the OWA is included within a comment in the document itself, the XML document as such would be under the OWA.

14. Of course, instead of the OWA, we could have included other principles for handling known and unknown information in the XML document. We could also have included the OWA or other principles in different XML documents, or even different principles in different parts of an XML document, as in the following example:

```
<!--The next four lines are under the open-world assumption-->
<person gender="male">
  <name>John Doe</name>
  <age>30</age>
</person>

<!--The next four lines are under the closed-world assumption-->
<person gender="female">
  <name>Jane Doe</name>
  <age>28</age>
</person>

<!--The next four lines are under the open-world assumption-->
<person gender="non-binary">
  <name>Numerio Negidio</name>
  <age>22</age>
</person>
```

15. It is even possible to include different, conflicting principles in the same XML document, such as:

```
<!--The next four lines are under the open-world assumption-->
<person gender="male">
  <name>John Doe</name>
  <age>30</age>
</person>
<!--The previous four lines are under the closed-world assumption-->
```

Since, under the closed-world assumption, any statement that is not known to be true is assumed to be false, the conflict does not concern what has already been assumed to be true and false in the XML document according to the OWA. Rather, the conflict arises over what we have assumed to be unknown according to the OWA, and which would otherwise be false under the closed-world assumption.

16. Sect. 10 has outlined three main strategies for dealing with comments in XML: ignoring them, considering data to take precedence over comments, and considering comments to take precedence over data. I would like to add that accepting these strategies has at least two implications. First, since none of the strategies refers to the content of comments, we should assume that these strategies can be applied to any comment, regardless of whether it actually contains a principle for handling known and unknown information. Second, since none of the strategies refers to (without contradictions) the subjects engaging with an XML document, we should assume that these strategies can work for any subject, whether the subjects are people or software applications. Instead, Sect. 13 has shown that the fact that XML as a metalanguage is not inherently bound by any principle for handling known and unknown information does not prevent an XML document from specifying and adopting its own principle(s). And specifying and adopting such a principle through a comment would bring us back to the three different strategies for dealing with comments in XML documents.

17. Things, moreover, can easily become more complicated by the following XML document:

```
<!--The next five lines are under the open-world assumption-->
<person gender="male">
  <!--John Doe’s age is 28-->
  <name>John Doe</name>
  <age>30</age>
</person>
```

Now, the first strategy is to ignore any comments, i.e. the first and third lines. This would remove any possible discrepancy between the data and the comments. As a result, according to the document, John Doe is a male person aged 30. However, since the document does not specify a notion of truth for the document, we cannot say that it is true for the XML document that John Doe is a male person aged 30. It would be true, for example, if the data were under the OWA. But this strategy prevents us from considering any comment, including the one about the OWA.

The second strategy is to consider the data as taking precedence over the comments. This would resolve any potential discrepancy between the data and the comments in favor of the data. Based on the data, we can say that in the document John Doe is a male person aged 30. Based on the first comment, it is true that John Doe is a male person aged 30, just as it is unknown whether John Doe has a son, for example. Furthermore, if we assume a one-to-one relationship between persons and their ages — a relationship that could be considered implicit in the semantics of “age” and thus part of the assumptions, but which is not specified in the XML document — then John Doe’s age is 30 and not 28, because the data take precedence over the comments. Conversely, if we do not assume such a relationship, then the first comment would also make it true that John Doe's age is 28.

Finally, the third strategy is to consider the comments as taking precedence over the data. This would resolve any potential discrepancy between the data and the comments in favor of the comments. So, based on the comments, we can say that in the document John Doe is a male person aged 28. Based on the first comment, it is true that John Doe is a male person aged 28, just as it is unknown whether John Doe has a son, for example. Furthermore, if we assume a one-to-one relationship between people and their ages, then John Doe's age should be 28, not 30, because comments take precedence over the data. Conversely, if we do not assume such a relationship, the first comment would also allow it to be true that John Doe's age is 30.

18. So, there is no shortage of controversies. First, the delineation of the three strategies for dealing with comments that contain or do not contain a principle for handling known and unknown information does not imply any normative claim about how to choose among these strategies. Rather, it simply leaves the choice to the subjects (humans or software applications) engaging with the XML documents. Each of these subjects may or may not have access to the code and/or assertoric subset, and may or may not take them into account. Second, it is also possible that subjects may have their own principles for handling known and unknown information — principles that may vary depending on their relationships to different XML documents and contexts — and that they may be confronted with XML documents that have different principles. This would require us to work out how conflicting principles from different sources (namely, subjects and documents) ultimately interact — that is, which principles take precedence, whether some principles should be ignored, and so on. Third, things can be further complicated by the XML documents discussed in Sects. 14, 15, 17, which include different principles in different parts of the same XML document, conflicting principles within the same XML document, and comments that may conflict with the data. All of these factors can lead to different interpretations of the same XML document, thereby undermining the successful communication between people and software applications that XML is intended to support.

19. In Sect. 1, I noted that by defining the content of an ontology, we can determine, for the ontology, the truth value of any statement that refers, or does not refer, to that content. In Sect. 2, I added that we cannot do the same with XML. The reason is that — unlike ontologies — XML, as a metalanguage, does not inherently embrace any principle for handling known and unknown information. This allows multiple principles to be applied to the same XML document and permits subjects to interpret it differently, thereby weakening the communication between people and software applications that XML — like ontologies — is supposed to support. In other words, the very feature that makes XML versatile is also what makes it epistemically unreliable. Where ontologies function as shared cognitive frameworks (Borst, 1997; Guarino, 1998; Studer at al., 1998), XML requires interpreters to supply their own principles to fill XML's cognitive void. In Sect. 3, I then hypothesized a way for XML to come close to achieving the same result as any ontology: by including comments that contain principles for handling known and unknown information in XML documents. After testing this hypothesis in Sects. 6–18, based on the remarks and assumptions in Sect. 3, I can conclude here that it is indeed possible for XML documents to establish the truth value of any statement that refers or does not refer to their content. However, the same sections have also shown that the hypothesis requires, at a minimum, that the subjects (humans or software applications) engaging with the XML documents are capable of taking the code into account. Even then, the hypothesis has limitations that may lead to different and potentially equally legitimate interpretations of the same XML documents, thus weakening the communication that XML, as a metalanguage, is supposed to support.

20. All of this highlights a key difference not only between ontologies and XML, but also between the theoretical assumptions underlying their communication. For ontologies, the bond between content, ontology language, and the principles for handling known and unknown information is inherent and independent of the subjects engaging with them. Thus, there is no room for subjects to provide different, alternative interpretations of the truth value of a statement that does or does not refer to the content of the ontologies — at most, there are misinterpretations if, for example, the subjects are unaware of the bond (Tambassi, 2025). Such a bond is not inherent in XML. Accordingly, the inclusion of a comment containing one of these principles does not guarantee that an XML document is under that principle, unless the subjects interpret the document by considering the code. In other words, the bond is not independent of the subjects that — as we have seen — can also provide different and equally legitimate interpretations of the same XML document. In this sense, communication between people and software applications diverges because of the role that ontologies and XML documents assign to the subjects engaging with their content: spectator and interpreter, respectively. This raises an urgent question about communication between people and software applications: can we afford to build communicative infrastructures in which interpretation is always left to the subjects engaging with the document — that is, interpreter-dependent? Or should we, like ontologies, harden our metalanguages with epistemic norms, even at the cost of flexibility? The answer may well determine whether communication is truly effective.

Funding. This paper has received funding from the European Research Council Executive Agency under the European Union Horizon Europe Research and Innovation Programme (GA n. 101041596 ERC – PolyphonicPhilosophy)

References

Ardagna, C. A., and N. Bena, 2025, "XML-Based access control languages", pages 2803–2808 in S. Jajodia, P. Samarati, and M. Yung (eds.), *Encyclopedia of Cryptography, Security and Privacy*, Springer: Cham. DOI: 10.1007/978-3-030-71522-9_833

Attenborough, M., 2003, *Mathematics for Electrical Engineering and Computing*, Newnes: Oxford.

Baader, F., I. Horrocks, C. Lutz, and U. Sattler, 2017, *An Introduction to Description Logic*, Cambridge University Press: Cambridge.

Baader, F., I. Horrocks, and U. Sattler, 2004, "Description logics", pages 3–28 in S. Staab and R. Studer (eds.), *Handbook on Ontologies. International Handbooks on Information Systems*, Springer: Berlin, Heidelberg.

Bernaras, A., I. Laresgoiti, and J. Corera, 1996, "Building and reusing ontologies for electrical network applications", pages 298–302 in *Proceedings of the European Conference on Artificial Intelligence* (ECAI'96), Budapest, Hungary.

Borst, W. N., 1997, *Construction of Engineering Ontologies, Centre for Telematics and Information Technology*, University of Twente: Enschede, The Netherlands.

Buda, A. G., and G. Primiero, 2024, "A pragmatic theory of computational artefacts", *Minds & Machines* 34(Suppl 1): 139–170. DOI: 10.1007/s11023-023-09650-0

Canducci, M., 2022, *XML per tutti*, Apogeo: Milan.

Dykes, L., 2005, *XML for Dummies*, 4th ed., Wiley: Hoboken.

Floridi, L., 2005, "Is semantic information meaningful data?", *Philosophy and Phenomenological Research* 70(2): 351–370. DOI: 10.1111/j.1933-1592.2005.tb00531.x

Franzén, N., 2021, "Fictional truth: In defense of the reality principle", pages 88–106 in E. Maier and A. Stokke (eds.), *The Language of Fiction*, Oxford University Press: Oxford.

Friend, S., 2016, "The real foundation of fictional worlds", *Australasian Journal of Philosophy* 95(1): 29–42. DOI: 10.1080/00048402.2016.1149736

Goy, A., and D. Magro, 2015, "What are ontologies useful for?", pages 7456–7464 in M. Khosrow-Pour (ed.), *Encyclopedia of Information Science and Technology*, IGI Global.

Gruber, T. R., 1993, "A translation approach to portable ontology specifications", *Knowledge Acquisition* 5(2): 199–220.

Gruber, T. R., 2009, "Ontology", in L. Liu and M. T. Özsu (eds.), *Encyclopedia of Database Systems*, Springer: Boston.

Guarino, N., 1998, "Formal ontology and information systems", pages 3–15 in Proceedings of FOIS'98, Trento, Italy, IOS Press: Amsterdam.

Guarino, N., and P. Giaretta, 1995, "Ontologies and knowledge bases — towards a terminological clarification", pages 25–32 in N. J. Mars (ed.), *Towards Very Large Knowledge Bases – Knowledge Building and Knowledge Sharing*, IOS Press: Amsterdam.

Henley, S., 2006, "The problem of missing data in geoscience databases", *Computers & Geosciences* 32(8): 1368–1377. DOI: 10.1016/j.cageo.2005.12.008

Jaziri, W., and F. Gargouri, 2010, "Ontology theory, management and design: An overview and future directions", pages 27–77 in F. Gargouri and W. Jaziri (eds.), *Ontology Theory, Management and Design: Advanced Tools and Models, Information Science Reference*, IGI Global. DOI: 10.4018/978-1-61520-859-3.ch002

Laurini, R., 2017, *Geographic Knowledge Infrastructure: Applications to Territorial Intelligence and Smart Cities*, ISTE-Elsevier: London.

Lewis, D., 1978, "Truth in fiction", *American Philosophical Quarterly* 15(1): 37–46.

Li, C., 2009, "XML parsing, SAX/DOM", pages 3598–3601 in L. Liu and M. T. Özsu (eds.), *Encyclopedia of Database Systems*, Springer: New York.

Li, X., Z. Wu, M. Goh, and S. Qiu, 2018, "Ontological knowledge integration and sharing for collaborative product development", *International Journal of Computer Integrated Manufacturing* 31(3): 275–288. DOI: 10.1080/0951192X.2017.1407876

Lutz, C., I. Seylan, and F. Wolter, 2012, 'Mixing open and closed world assumption in ontology-based data access: Non-uniform data complexity', in Y. Kazarov, D. Lembo, and F. Wolter (eds.), *Proceedings of the International Workshop on Description Logics*, DL 2021, CEUR Workshop Proceedings, vol. 846.

Neches, R., R. E. Fikes, T. Finin, T. R. Gruber, T. Senator, and W. R. Swartout, 1991, "Enabling technology for knowledge sharing", *AI Magazine* 12(3): 36–56.

Noy, N. F., and D. L. McGuinness, 2003, *Ontology Development 101: A Guide to Creating Your First Ontology*, Stanford University.

Porello, D., and U. Endriss, 2014, "Ontology merging as social choice: Judgment aggregation under the open world assumption", *Journal of Logic and Computation* 24(6): 11229–11249. DOI: 10.1093/logcom/exs056

Razniewski, S., O. Savkovic, and W. Nutt, 2016, "Turning the partial-closed world assumption upside down", in R. Pichler and A. Soares da Silva (eds.), *Proceedings of the 10th Alberto Mendelzon International Workshop on Foundations of Data Management*, Panama City, Panama, May 8–10, 2016, CEUR.

Rector, A., S. Schulz, J. M. Rodrigues, C. G. Chute, and H. Solbrig, 2019, "On beyond Gruber: 'Ontologies' in today's biomedical information systems and the limits of OWL", *Journal of Biomedical Informatics* 100(S): 100002. DOI: 10.1016/j.yjbinx.2019.100002

Russell, S. J., and P. Norvig, 2020, *Artificial Intelligence: A Modern Approach*, Pearson Publications: London.

Ryan, M. L., 1980, "Fiction, non-factuals and the principle of minimal departure", *Poetics* 9(4): 403–422.

Salminen, A., and F. Tompa, 2011, *Communicating with XML*, Springer: New York.

Selmer, B., and D. A. Ferrucci, 1998, "Logic and artificial intelligence: Divorced, still married, separated ... ?", *Minds and Machines* 8: 273–308. DOI: 10.1023/A:1008250617134

Siegel, E., 2022, *Schematron: A Language for Validating XML*, XML Press.

Smith, B., 2003, "Ontology", pages 155–166 in L. Floridi (ed.), *Blackwell Guide to the Philosophy of Computing and Information*, Blackwell: Malden.

Sowa, J. F., 2005, "Guided tour of ontology" available at http://www.jfsowa.com/ontology/guided.htm

Sperberg-McQueen, C. M., and C. Huitfeldt, 2011, "Ten problems in the interpretation of XML documents", pages 157–174 in A. Mehler et al. (eds.), *Modeling, Learning, and Processing of Text Technological Data Structures, Studies in Computational Intelligence*, vol. 370, Springer: Berlin, Heidelberg. DOI: 10.1007/978-3-642-22613-7_9

Studer, R., V. R. Benjamins, and D. Fensel, 1998, "Knowledge engineering: Principles and methods", *IEEE Transactions on Data and Knowledge Engineering* 25(1–2): 161–197.

Swartout, B., P. Ramesh, K. Knight, and T. Russ, 1997, "Toward distributed use of large-scale ontologies", in *AAAI Symposium on Ontological Engineering*, Stanford, CA.

Taira, K., T. Itaya, S. Yada, K. Hiyama, and A. Hanada, 2025, "Impact of attached file formats on the performance of ChatGPT-4 on the Japanese National Nursing Examination: Evaluation study", *JMIR Nursing* 8: e67197.DOI: 10.2196/67197

Tambassi, T., 2024, "On perspectivism of information system ontologies", *Foundations of Science* 29: 571–585. DOI: 10.1007/s10699-023-09900-5

Tambassi, T., 2025, "Do ontologies always support communication of their content among human agents?", *AI & Society* 40, (2025): 3369–3370. DOI: 10.1007/s00146-024-02100-0

Tambassi, T., 2026, "On what there is not: Extensible markup language and rules for making inferences", *Logic and Logical Philosophy* 35(1), (2026): 113–127. DOI: 10.12775/LLP.2025.007

Uschold, M., and R. Jasper, 1999, "A framework for understanding and classifying ontology applications", in *Proceedings of the IJCAI'99 Workshop on Ontologies and Problem-Solving Methods*, Stockholm, Sweden.

Walton, K., 1990, *Mimesis as Make-believe: On the Foundations of the Representational Arts*, Harvard University Press: Cambridge.

Wolterstorff, N., 1980, *Works and Worlds of Art*, Clarendon Press: Oxford.

Zverev, E., S. Abdelnabi, M. Fritz, and C. H. Lampert, 2024, "Can LLMs separate instructions from data? And what do we even mean by that?", in *ICLR 2024 Workshop on Secure and Trustworthy Large Language Models*. https://openreview.net/forum?id=32eytC1Nt1

Timothy Tambassi
Department of Philosophy and Cultural Heritage
Ca' Foscari University of Venice
Venice, Italy
timothy.tambassi@gmail.com
https://orcid.org/0000-0001-6460-5920