

A Functional Semantic Web Architecture

Aurona Gerber, Alta van der Merwe, and Andries Barnard

Meraka Institute and University of South Africa (Unisa),

Pretoria, South Africa

aurona.gerber@meraka.org.za,

{vdmeraj,barnaa}@unisa.ac.za

<http://www.meraka.org.za>

<http://www.unisa.ac.za>

Abstract. A layered architecture for the Semantic Web that adheres to software engineering principles and the fundamental aspects of layered architectures will assist in the development of Semantic Web specifications and applications. The most well-known versions of the layered architecture that exist within literature have been proposed by Berners-Lee. It is possible to indicate inconsistencies and discrepancies in the different versions of the architecture, leading to confusion, as well as conflicting proposals and adoptions by the Semantic Web community. A more recent version of a Semantic Web layered architecture, namely the CFL architecture, was proposed in 2007 by Gerber, van der Merwe and Barnard [23], which adheres to software engineering principles and addresses several of the concerns evident from previous versions of the architecture. In this paper we evaluate this recent architecture, both by scrutinising the shortcomings of previous architectures and evaluating the approach used for the development of the latest architecture. Furthermore, the architecture is applied to usage scenarios to evaluate the usefulness thereof.

Keywords: Semantic Web Architecture, Software Engineering, System Architecture, Layered Architecture.

1 Introduction

The establishment of the architecture of any information system is one of the crucial activities during the design and implementation thereof. There is general consensus in literature that an architecture (at least) depicts the structure of a system within a specific context. This depicted structure should portray the components that a system comprises of, as well as the relationships between the identified components. One of the main purposes of a system architecture is the provision of an agreed-upon functional description of system structure, components and component interactions. It is thus plausible to state that an architecture for the Semantic Web is crucial to its eventual realisation and that it is therefore necessary to attach indisputable meaning to the specification of the architecture for the languages of the Semantic Web.

The most well-known versions of the layered architecture that exist within literature have been proposed by Berners-Lee, and the literature offers no description or specification of meaning for any of these. It is possible to indicate inconsistencies and discrepancies in the different versions of this reference architecture, leading to confusion, as well as conflicting proposals and adoptions by the Semantic Web community. In addition, none of the current formal initiatives by the W3C address the Semantic Web architecture specifically, which could be regarded as an omission.

A layered architecture for the Semantic Web that adheres to software engineering principles and the fundamental aspects of layered architectures will assist in the development of Semantic Web specifications and applications. Furthermore, several of the current research and implementation issues associated with the implementation of the Semantic Web could potentially be resolved.

A more recent version of a Semantic Web layered architecture, namely the CFL architecture, was proposed by Gerber, van der Merwe and Barnard [23]. They claim that their abstracted CFL architecture of the Semantic Web adheres to software engineering principles and addresses several of the concerns evident from previous versions of the architecture.

In this paper we evaluate this recent architecture, both by scrutinising the shortcomings of previous architectures and evaluating the approach used for the development of the latest architecture. Furthermore, the architecture is applied to usage scenarios to evaluate the usefulness thereof. A similar approach was used in the construction of one of the most significant layered architectures in popular use today, notably the ISO/OSI reference model for network protocols. We reach the conclusion that the approach indeed has merit in resolving current issues with regards to the architecture of the languages of the Semantic Web. However, the proposed version needs to be refined through consensus by all role players, including the W3C.

This paper starts with a background in section 2 describing the four versions of the Semantic Web layered architecture as proposed by Berners-Lee [6, 9, 10, 11, 12, 13]. This discussion includes an investigation into some inconsistencies prevalent from these architectures. Section 3 contains a discussion of the proposed CFL architecture for the Semantic Web. Section 4 evaluates the CFL architecture by applying it to usage scenarios and section 5 concludes this paper.

2 Background: The Semantic Web Architecture

Since the publication of the original Semantic Web vision of Berners-Lee, Hendler and Lassila [15], Berners-Lee proposed four versions of a Semantic Web architecture, for the purposes of this discussion called the reference architecture. All of the reference architecture versions were presented by Berners-Lee in *presentations*, they were never published in literature or included as part of a W3C Recommendation. The different versions are depicted as V1 to V4 in Figure 1.

In support of the founding vision of the Semantic Web [15], a first version (V1) of a layered architecture was introduced in 2000 by Berners-Lee [6]. As

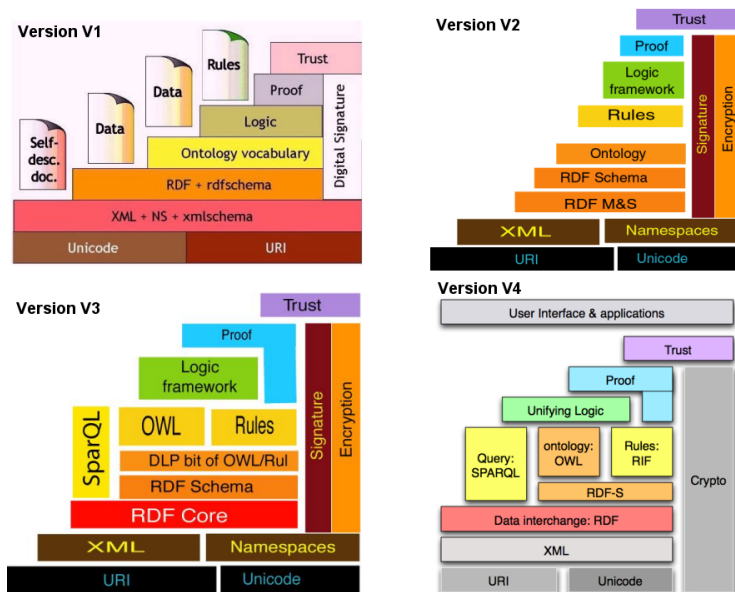


Fig. 1. The four versions of the Semantic Web reference architecture (V1-V4) proposed by Berners-Lee [6, 9, 10, 11, 12, 13]

participant in the ongoing activities of the W3C, Berners-Lee proposed a second version (V2) of the Semantic Web reference architecture in 2003 as part of a presentation at the SIIA Summit [30]. V2 was furthermore presented as part of two presentations in 2003 [9, 10, 11]. Berners-Lee proposed a third version (V3) at WWW2005 [12], and in his keynote address at AAAI2006 in July 2006, Berners-Lee introduced the latest (V4) version of the Semantic Web reference architecture [13].

In different publications, researchers argued that the the Semantic Web architectures cannot be described as *architectures* since Semantic Web *languages* or *W3C technologies* are depicted. They argue that this is not the nature of an architecture and that the term *architecture* is generally used to depict system functionality at different conceptual levels. The proposed versions of the Semantic Web architecture does not depict functionality and should be described as a *stack* or even *layered cake* [1, 24].

In order to respond to this argument, a definition for the term *system architecture* was compiled from an extensive literature analysis. There is general consensus in literature that an architecture *at least* depicts the structure of a system within a specific context [3, 4, 19, 20]. This structure should portray the components that a system comprise of, as well as the relationships between the identified components. System architects depicting the organisation of system components often make use of several identified architectural patterns, one of which is the *layered architecture* [2, 4, 17, 28]. In a layered architecture the

principal elements or components are arranged in the form of a stack where each layer resides above a lower layer. Generally, a layer represents a grouping of elements that provides related services. A higher layer may use either only various services defined by the immediate lower layer (closed architecture) or services by all of the lower layers (open architecture). However, the lower layers are unaware of higher layers [2, 17, 18, 28], and are not allowed to access functionality provided by upper layers. This implies a strict ordering of access to the functionality provided by components in a layered architecture in one direction only [3].

In conclusion of the proposition that the Semantic Web layered architecture versions are regarded as *architectures*, we provide an example of a similar, widely disseminated architecture, namely the ISO/OSI (International Standards Organisation / Open Systems Interconnect) layered architecture. This architecture specifies the functionality required to define *protocols* necessary for *network interoperability* between applications [32]. The Semantic Web layered architecture has purpose similar to that of the ISO/OSI layered architecture in that it aims to depict the *languages* necessary for *data interoperability* between applications. The ISO/OSI model is regarded as an *architecture*, and thus it is proposed that the Semantic Web model should be regarded in the same way as an *architecture* for the purpose of this discussion.

If the Semantic Web architectures are regarded as *architectures*, we have to define the context, components and relations. Within the *context* of the *languages required for meta-data specification*, the proposed versions of the reference architecture [6, 9, 12, 13] depict the organisation of the *language* components for the Semantic Web. The layering of the languages provides the *structure*. Thus, the general definition of an *system architecture* is adhered to. The fact that *functionality* is not always depicted on layers within the current versions should be regarded as an omission. This omission provides one of the motivations for the development of a *functional* architecture for the Semantic Web. In addition, an investigation into the structure of the reference architecture versions highlighted some discrepancies and irregularities.

□ Side-by-side layers

In all four versions depicted in Figure 2 *URI* and *Unicode* are depicted on the bottom layer as two separate blocks or side-by-side layers (refer to (1) in Figure 2). It can be assumed that the intended implication is that the two technologies both reside on the bottom layer. However, this is contentious and inconsistent when referring to the presentation of other layers containing more than one technology such as V1 that depicts the *RDF* and *RDF Schema* technologies in one layer (*RDF + rdfschema*). In V2, *XML* and *Namespaces* are also depicted as side-by-side layers, similar to *URI* and *Unicode* on the bottom layer. However, to be consistent with the assumptions of layering, the layering of this version should imply that *XML* only uses *URI*, whereas *Namespaces* uses only *Unicode* as a preceding layer. Based on our knowledge of XML and Namespaces, this is not the case even though it is depicted as such. The same discussion holds for several examples in all versions.

□ **Triangular structure of the layered architecture**

In all four versions depicted in Figure 2 the layers are staggered into a triangular structure with the lower layers wider than upper layers (refer to (2) in Figure 2). It is not clear whether this means that the upper layers use only *part* of what is provided by lower layers, or whether a lower layer specifies additional functionality that is not used for the purpose of the Semantic Web.

□ **Mixing technologies and functionality descriptions in the naming of layers**

It is not clear what the layers in the four versions represent since certain layers are labelled using *technologies* whilst others are labelled using *functionality descriptions* (refer to (3) in Figure 2). In all versions the bottom layers represent technologies while higher layers are labelled with functionality descriptions.

□ **Vertical layers**

All versions of the architecture in Figure 2 depict vertical layers such as *Digital Signatures* in V1 and *Crypto* in V4 (refer to (4) in Figure 2). The precise meaning of any of these vertical layers is not specified in literature. It is possible to speculate that these layers are included in all the other layers, or that these layers reside alongside the other layers, or even that these layers only depict technologies building on top of their lower layers and excluding the upper layers. Furthermore, it is possible to remove any *security related* vertical layer in any of the versions of the Semantic Web layered architecture without compromising the integrity of the system as regards to the *languages* of the Semantic Web.

In summary, the versions of the reference architecture Berners-Lee [6, 9, 12, 13] have not been formally documented and all depict certain inconsistencies and

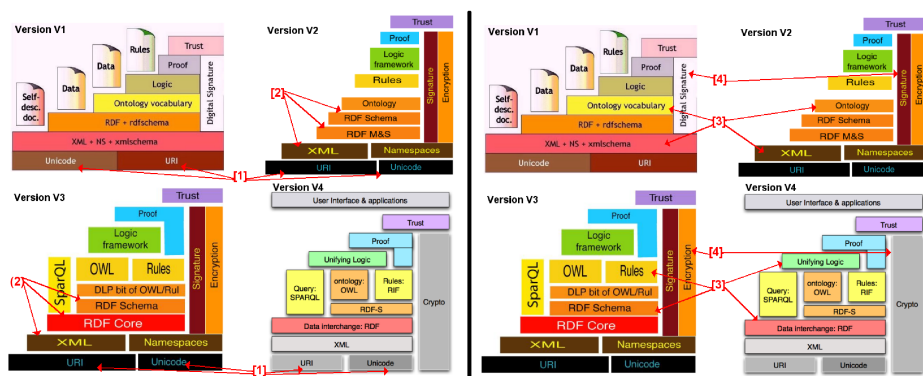


Fig. 2. Versions V1-V4: side-by-side layers (1), triangular structure (2), functionality vs technology (3) and vertical layers (4)

discrepancies. This often leads to confusion, as well as conflicting proposals and adoptions of technology and functionality by the Semantic Web community [25, 26, 29]. It is not unreasonable to propose that the architecture of any system is one of the primary aspects to consider during design and implementation thereof, and the proposed architecture of the Semantic Web is thus crucial to its eventual realisation. It is necessary to attach indisputable meaning to the specification of the architecture for the languages of the Semantic Web. In addition, because there is no precise functionality and interface definition, applications implementing Semantic Web technologies using these architectures will not be able to interoperate. The interoperation of Semantic Web applications is crucial for the eventual realisation of the founder vision.

It is interesting to note at this stage that, as part of the ongoing W3C design initiatives, Berners-Lee argues for the inclusion of certain software engineering design principles into W3C design efforts, and he identifies four principles namely simplicity, modularity, decentralisation and tolerance [7]. *Simplicity* strives to use only a few basic elements to achieve the required results. *Modularity* suggests a system design adhering to *loose coupling* and *tight cohesion* of system elements. *Decentralisation* avoids any common point that may be a single point of complete failure and *Tolerance* specifies liberal requirements and conservative implementations. In spite of these design principles, none of the current formal initiatives by the W3C address the Semantic Web architecture specifically [5, 7, 8, 14, 16, 31].

An approach to develop a layered architecture from software engineering principles was proposed by Gerber et al. [23], referred to as the Comprehensive, Functional, Layered (CFL) architecture for the Semantic Web. *Comprehensive* implies that the architecture is based on software engineering principles, *functional* means that the architecture depicts required functionality and *layered* refers to the architecture structure [23].

3 The CFL Architecture for the Semantic Web

The proposed CFL architecture (Figure 3) for the Semantic Web varies from the previously suggested versions (V1-V4) of the architectures mainly because it adheres to the evaluation criteria for layered architectures that are based on established software engineering principles [21, 22, 23]. In addition, it is noticeable that the CFL architecture abstracts and depicts related functionalities rather than the W3C technologies used to instantiate these functionalities. A detailed discussion on the development of the CFL architecture can be found in Gerber et al. [23].

From Figure 3 a reader may conclude that the proposed CFL architecture depicts only a simplification of the original architecture. The *abstraction* of the functionality from the original versions of the Semantic Web layered architecture does indeed entail a simplification of the model, but this simplification introduces several advantages such as a more understandable and universal model that facilitates debate. One of the most significant advantages is that such an

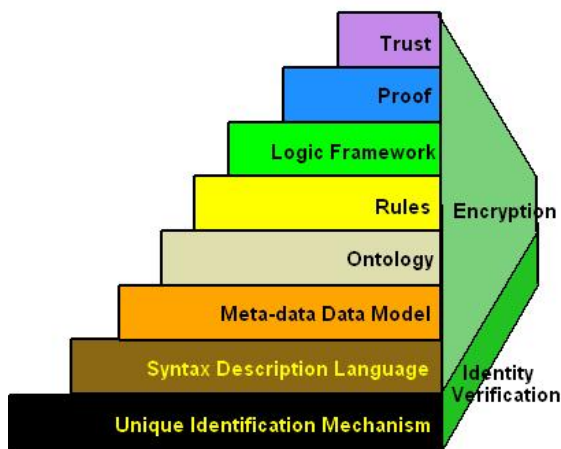


Fig. 3. The proposed CFL architecture for the Semantic Web comprises two orthogonal architecture stacks, the *language* stack and the *security* stack

abstraction enables the use of diverse and non-related technologies to implement the functionality of a specific layer. In other words, different technologies could be used to instantiate the functionality of a specific layer in the proposed Semantic Web layered architecture.

To determine whether this CFL architecture is indeed useful for standardisation of Semantic Web technologies, as well as the resolution of the layering debate, the CFL architecture is evaluated using usage scenarios.

4 Evaluation of Usefulness

The first usage scenario was obtained from literature and the proposed CFL architecture is applied to solve some of the issues with regard to the layering of the Semantic Web in section 4.1. In addition, the current V4 Semantic Web reference architecture [13] is investigated as being an instantiation of the proposed CFL architecture (section 4.2). Lastly, a practical application example was developed using the proposed CFL architecture in section 4.3.

4.1 Scenario: The Two-tower Architecture of Horrocks, Parsia, Patel-Schneider and Hendler[25]

Originally, versions V2 and V3 of the reference architecture were discussed by Kifer, Bruijn, Boley and Fensel [27], who argued strongly in support of *multiple independent, but interoperable, stacks of languages*. They argued that a single stack architecture as depicted in V2 (Figure 1) is *unrealistic* and *unsustainable* and they regarded the side-by-side layering of *OWL* and *Rules* in V3 (Figure 1) as an implementation of the *interoperable stacks of languages* concept. They therefore supported the *multi-stack architecture* as depicted by V3. In particular, they discuss

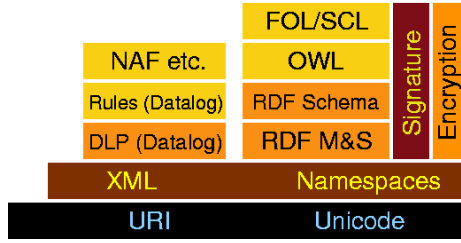


Fig. 4. The two-tower architecture of Horrocks, Parsia, Patel-Schneider and Hendler [25]

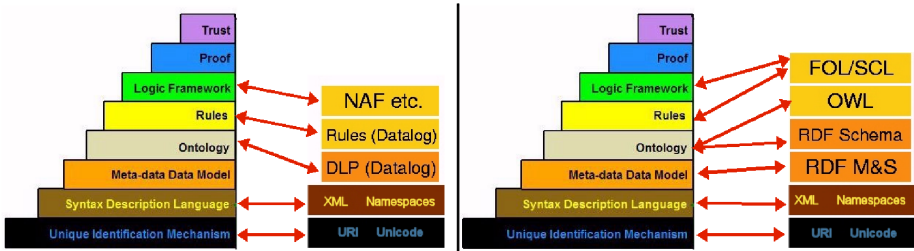


Fig. 5. The two-tower architecture as instantiations of the Semantic Web CFL architecture

the incorporation of the Datalog rule-based languages as *Rules*, which could not be layered on top of OWL, as a separate stack alongside the stack of OWL-based languages with their prescribed Description Logics.

Horrocks, Parsia, Patel-Schneider and Hendler [25] entered the debate about the positioning of *Rules* with their *Semantic Web Architecture: Stack or Two Towers?* article. They further argued that a realistic architecture for the Semantic Web had to be based on multiple independent, but interoperable, stacks of languages. In particular, they pointed out that DLP/Datalog and RDFS/OWL are not semantically compatible and cannot be layered as proposed in the V3 version of the present Semantic Web layered architecture (Figure 1). They consequently proposed a two-tower architecture (Figure 4) as the solution for the acceptance of both RDFS/OWL and DLP/Datalog as Semantic Web languages or technologies [25].

This two-tower architecture of Horrocks et al. [25] depicts two possible instantiations of the Semantic Web layered architecture. In order to demonstrate the value of the proposed CFL architecture, the instantiations in the two towers of Horrocks et al. [25] are related to the CFL architecture demonstrating the inclusion of different technologies. The relation of the two towers to the CFL architecture for the Semantic Web is depicted in Figure 5.

In *Tower 1* on the left-hand side in Figure 4, Horrocks et al. omitted an explicit data representation layer. Datalog was developed as a rule and query language for deductive databases and syntactically Datalog is a subset of Prolog,

hence it does not constitute a data representation layer. To improve the composition of Tower 1, a *meta-data data model* could be added. Thus, the adapted Semantic Web architecture supports more than one technology to be used in this layer. In addition, the Ontology functionality is implemented by DLP (Description Logic Programs) where these DLP extend Datalog to include knowledge representation. The Rules functionality is implemented with Datalog Rules and NAF. This relationship is depicted in Figure 5.

In contrast to Tower 1, Tower 2 on the right-hand side in Figure 4 implements the meta-data data model functionality by means of RDF. The Ontology layer functionality is implemented using RDF Schema and OWL. In this scenario, the contention is that the FOL/SCL layer provides for Rules and to a limited extent, Logic Framework functionality. This instantiation is depicted in Figure 5.

Note that the purpose of this scenario is not to resolve the layering debate of the different technologies. In this paper, we evaluate the usefulness of the CFL architecture. The architecture defines *functionality* that has to be implemented by the different layers, and which allows for the acceptance of diverse technology *instantiations* that implement the requisite functionality. This is aptly demonstrated in this usage scenario where both the *towers* are accommodated by the proposed Semantic Web CFL architecture, even though they represent different semantic bases.

4.2 Scenario: The V4 Semantic Web Architecture

In this section, the latest version (V4) of the layered architecture as proposed by Berners-Lee [13] is investigated as being an instantiation of the proposed CFL architecture for the Semantic Web. Figure 6 depicts the mapping of the depicted W3C technologies to the functionality layers of the CFL architecture. The mapping irregularities, which is due to the inconsistencies in the V4 architecture as discussed in section 2.

The *Unique Identification Mechanism* layer maps to Unicode and URI as before. URI essentially uses Unicode and therefore the meaning of the side-by-side layering of Unicode and URI in V4 is unclear. However, it is plausible to state that URI making use of Unicode is an instantiation of the layer 1 *unique identification mechanism* functionality of the CFL architecture. XML instantiates the *Syntax Description Language* layer and the *meta-data data model* is instantiated with RDF.

The *Ontology* layer of the CFL architecture is instantiated with either RDF Schema (depicted as RDF-S in V4) or RDF Schema and OWL (RDF-S and OWL). If RDF-S as instantiation is used, RIF is the instantiation of the *Rules* layer. *RIF* is an acronym for *Rule Interchange Format* and it is a draft specification for a rule language using RDF Schema as its ontology basis.

Above RIF, no *technologies* are depicted in V4 and therefore *instantiations* cannot be discussed. *Unifying Logic* maps to *Logic Framework*, and *Proof* and *Trust* remains *Proof* and *Trust*. It is unclear what is meant by both *Unifying Logic* and *Proof* residing on top of *Rules:RIF*. It is possible to reason that both reside alongside each other above *Rules:RIF* or it might mean that the layered

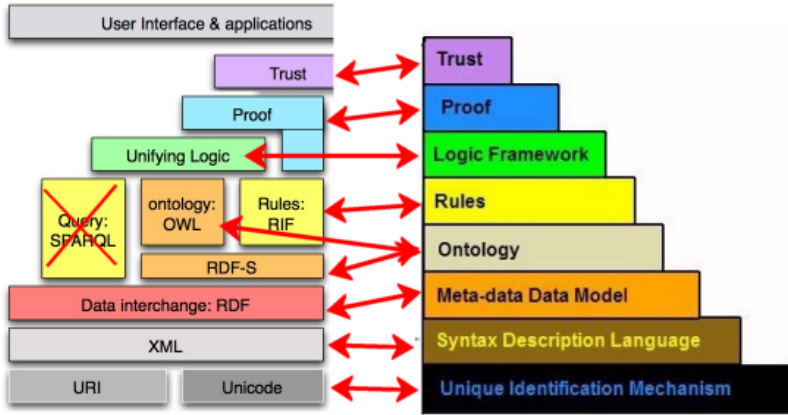


Fig. 6. The V4 version of the Semantic Web reference architecture [13] as instantiation of the adapted architecture

architecture is *open* in this instance so that *Proof* might access functionality of *RIF* directly without having to access the *Unifying Logic* layer. However, in general the technologies from URI and Unicode through to RIF can be regarded as an instantiation of the adapted, functional architecture.

In contrast, if the *Ontology* layer is instantiated with RDF Schema together with OWL (RDF-S and OWL), a deviation is depicted in the layers because of the omission of a *Rules* layer above *OWL*. It is however foreseen that a rule language will be required above an OWL ontology, hence the SWRL initiative of the W3C (RIF builds on RDF Schema only). It is plausible to speculate that this omission will have to be rectified in future. The CFL architecture thus allows for diverse technology specifications for the instantiation of a layer.

It can be argued that the latest version of the Semantic Web reference architecture also presents two towers of possible technology instantiations, and that both can be regarded as instantiations of the CFL architecture.

4.3 Application Usage Scenario

In this section the usefulness of the CFL architecture is illustrated by simulating two applications using two different meta-data data models.

In this usage scenario, the proposed system is a distributed management application for book publishers using Semantic Web technologies of the bottom three layers as depicted in Figure 7. The implementation or instantiation of the technology at the layer of communication should be similar. In this scenario, the meta-data data model could be either an entity relationship model, or an RDF diagram. Because the system implements up to and including Layer 3, the layer 3 models have to be implemented using the same technology.

The important aspect is that there is a *choice* of meta-data data models when the proposed CFL architecture for the Semantic Web is used. In contrast, the original reference architecture would have limited the choice to RDF.

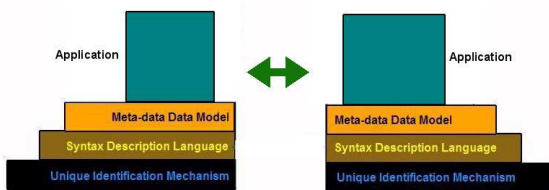


Fig. 7. Applications using the first three layers of the proposed Semantic Web CFL architecture

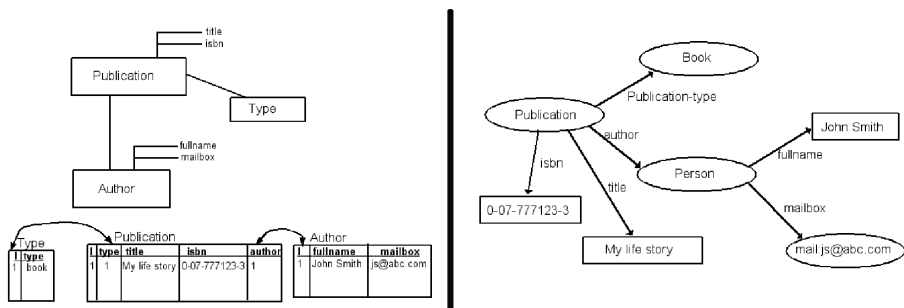


Fig. 8. An entity relationship diagram with a database implementation (left) and a RDF diagram (right) modelling the *publication* data of the scenario

In the scenario data model, a *publication*, which is of a specific *type* (a book in this case), has a *title* ('*My life story*'), an *isbn* (0-07-777123-3) and an *author*. The *author* is a *Person* with a *fullname* (John Smith) and a *mailbox* (mail:js@abc.com).

The Layer 3 implementation will be a data model using either an ER (entity relationship) diagram or an RDF diagram. A possible ER diagram is depicted on the left in Figure 8. Similarly, this data can be modelled in an RDF diagram. An RDF diagram modelling the same data is depicted on the right in Figure 8.

Figure 8 therefore depict two possible choices of meta-data data modelling technologies that can be used in this scenario as the proposed CFL architecture does not prescribe the technology. As long as the implementation adheres to the principles of layering and the specified functionality of the layer it implements, any meta-data data modelling technology can be used. Both models use a Layer 2 that is instantiated with XML using various serialisation techniques, and a Layer 1 that decodes to Unicode URIs. The Layer 1 implementation for both systems uses URI to uniquely identify the resources under discussion. In addition, the systems encode the serialised XML using Unicode to ensure that it is uniquely interpreted across the world.

In the usage scenario in this section an application is implemented using the proposed CFL architecture for the Semantic Web. Developers are able to choose between alternative technologies to implement the different layers of the

CFL architecture for the Semantic Web effectively. In this usage scenario, it is demonstrated how two possible meta-data data-modelling technologies can be used to implement Layer 3.

5 Conclusion

Within this paper the different versions of the Semantic Web reference architecture proposed by Berners-Lee [6, 9, 12, 13] were discussed. The versions primarily depict technologies adopted by the W3C as standards or W3C Recommendations. This correlates with the mandate of the W3C that includes the development of *specifications* for the Web. However, when compared to definitions and criteria for architectures from software engineering principles, it is possible to identify several inconsistencies and discrepancies in these proposed versions. This often leads to confusion, as well as conflicting proposals and adoptions of technology and functionality by the Semantic Web community. Furthermore, Semantic Web applications based on technologies using these architectures will not be able to interoperate because there is no precise functionality and interface definition.

The design of the architecture of any system is a crucial activity during its development and it is necessary to attach precise and indisputable meaning to the system components, their functionality and relationships with the associated interfaces. Similarly, it is necessary to attach indisputable meaning to the architecture for the languages of the Semantic Web.

When an architecture is designed, it is necessary to adhere to software engineering design principles. These principles imply that a system architecture should represent functional system components within a specific context with clearly defined purposes and interfaces. An example of such an architecture is the ISO/OSI layered architecture for network interoperability.

Similarly to the ISO/OSI model description and development approach, Gerber et al. [23] proposed a CFL architecture for the Semantic Web that aims to present a structure that permits the meta-data languages for the Semantic Web to be viewed as logically composed of a succession of layers, each wrapping the lower layers and isolating them from the higher layers [32]. The CFL architecture depicts a simplification of the original architecture versions proposed by Berners-Lee as a result of the *abstraction* of required functionality of language layers. Gerber et al. [23] argues that an abstracted layered architecture for the Semantic Web with well-defined functionalities will assist with the resolution of several of the current Semantic Web research debates such as the layering of language technologies. A functional architecture can accommodate diverse technologies for implementation of the required functionalities of each layer as long as the layer implementation adheres to interface specifications.

It is acknowledged that all issues with regard to the layering of the Semantic Web languages have not been resolved by this approach. However, the establishment of an approach and a first iteration CFL architecture for the Semantic Web is an important and necessary step towards realising the eventual notion of

the Semantic Web. The CFL architecture for the Semantic Web should assist in the development of Semantic Web specifications and applications, or even W3C Recommendations, and several of the current research and implementation issues associated with the implementation of the Semantic Web could potentially be resolved. The CFL architecture might even assist the W3C to include all technology developments rather than to make a choice for the adoption of only one standard to the exclusion of other technologies. This position is similar to the position adopted by the network community with the acceptance of the ISO/OSI layered architecture as architectural framework for network interoperability standards.

In order to evaluate the usefulness of the proposed CFL architecture with its associated claims, the architecture was applied to various usage scenarios. In all of the scenarios, the CFL architecture proved to be of value.

It is plausible to speculate that the present versions of the Semantic Web reference architecture were developed to depict the progression pertaining to the development of W3C technologies. The definition of a general, accepted layered architecture with functional components (CFL architecture) could possibly assist the W3C to develop different technology specifications for the implementation of a specific layer functionality. This approach would *include*, rather than *exclude* technologies that would have as benefit a more rapid adoption of the Semantic Web vision. In addition, the definition of W3C specifications requires an appropriate level of abstraction to be able to penetrate sufficiently into the implementation domain, and it is the contention of the author that this can be provided by the CFL architecture for the Semantic Web as proposed in this paper.

References

- [1] Antoniou, G., von Harmelen, F.: A Semantic Web Primer. MIT Press, Cambridge (2004)
- [2] Bachman, C.: Personal Chronicle: Creating Better Information Systems, with Some Guiding Principles. IEEE Transactions on Knowledge and Data Engineering, 17–32 (1989)
- [3] Bachmann, F., Bass, L., Carriere, J., Clements, P., Garlan, D., Ivers, J., Nord, R., Little, R.: Software Architecture Documentation in Practice: Documenting Architectural Layers. Technical Report CMU/SEI-2000-SR-004, Carnegie Mellon Software Engineering Institute (2005)
- [4] Bass, L., Clements, P., Kazman, R.: Software Architecture in Practice. Addison Wesley Professional, Reading (2003) (Last accessed 12/8/2006)
<http://www.aw-bc.com/catalog/academic/product/0,4096,0321154959,00.html>
- [5] Berners-Lee, T.: Axioms of Web Architecture: Metadata Architecture (1997) (Last accessed 2/10/2006), W3C Web site
<http://www.w3.org/DesignIssues/Metadata.html>
- [6] Berners-Lee T.: Semantic Web - XML 2000. W3C Web site (2000) (Last accessed 11/8/2006),
<http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html>

- [7] Berners-Lee, T.: Axioms of Web Architecture: Principles of design (2002) (Last accessed 5/10/2006), W3C Web site
<http://www.w3.org/DesignIssues/Principles.html>
- [8] Berners-Lee, T.: Web Architecture from 50,000 feet (2002) (Last accessed 4/10/2006), W3C Website
<http://www.w3.org/DesignIssues/Architecture.html>
- [9] Berners-Lee, T.: The Semantic Web and Challenges (2003) (Last accessed 10/10/2006), W3C Web site Slideshow
<http://www.w3.org/2003/Talks/01-sweb-tbl/0verview.html>
- [10] Berners-Lee, T.: Standards, Semantics and Survival. SIIA Upgrade, pp. 6–10 (2003)
- [11] Berners-Lee, T.: WWW Past and Future (2003) (Last accessed 10/10/2006), W3C Web site
<http://www.w3.org/2003/Talks/0922-rsoc-tbl/slide30-0.html>
- [12] Berners-Lee, T.: WWW 2005 Keynote. W3C Web site (2005) (Last accessed 10/10/2006),
<http://www.w3.org/2005/Talks/0511-keynote-tbl/>
- [13] Berners-Lee, T.: Artificial Intelligence and the Semantic Web: AAAI 2006 Keynote (2006) (Last accessed 12/10/2006), W3C Web site
<http://www.w3.org/2006/Talks/0718-aaai-tbl/0verview.html>
- [14] Berners-Lee, T., Bray, T., Connolly, D., Cotton, P., Fielding, R., et al.: Architecture of the World Wide Web, Volume One (2004), W3C Web site
<http://www.w3.org/TR/2004/REC-webarch-20041215/>
- [15] Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. *The Scientific American* 5(1), 36 (2001) (Last accessed 20/9/2006),
<http://www.scientificamerican.com/2001/0501issue/0501berbers-lee.html>
- [16] Bos, B.: What is a good standard? An essay on W3C's design principles (2003) (Last accessed 2/10/2006), W3C Web site
<http://www.w3.org/People/Bos/DesignGuide/introduction>
- [17] Bruegge, B., Dutoit, A.H.: *Object-oriented Software Engineering using UML, Patterns, and Java*, 2nd edn. Prentice-Hall, Englewood Cliffs (2004)
- [18] Fowler, M.: *Patterns of Enterprise Application Architecture*. Addison-Wesley, Boston (2003)
- [19] Garlan, D.: Software architecture: a roadmap. In: *ICSE 2000: Proceedings of the Conference on The Future of Software Engineering*, ACM Press, New York (2000)
- [20] Garlan, D., Shaw, M.: *An Introduction to Software Architecture* (1994) (Last accessed 15/9/2006), http://www.inf.ed.ac.uk/teaching/courses/seoc1/2005-2006/resources/intro_softarch.pdf
- [21] Gerber, A., Barnard, A., van der Merwe, A.: Design and Evaluation Criteria for Layered Architectures. In: *Proceedings of the MSVVEIS Workshop hosted at the 8th International Conference on Enterprise Information Systems*, Paphos, Cyprus, pp. 163–172 (2006) ISBN 972-8865-49-8
- [22] Gerber, A., Barnard, A., van der Merwe, A.: A Semantic Web Status Model. In: *Proceedings of the Ninth World Conference on Integrated Design & Process Technology*, San Diego, California, IEEE, Los Alamitos (2006)
- [23] Gerber, A., van der Merwe, A., Barnard, A.: Towards a Semantic Web Layered Architecture. In: *Proceedings of IASTED International Conference on Software Engineering (SE 2007)*, Innsbruck, Austria, pp. 353–362 (2007) ISBN 978-0-88986-641-6

- [24] Hendler, J.: Agents and the Semantic Web. *IEEE Intelligent Systems* 16, 30–37 (2001)
- [25] Horrocks, I., Parsia, B., Patel-Schneider, P., Hendler, J.: Semantic Web Architecture: Stack or Two Towers? In: Fages, F., Soliman, S. (eds.) *PPSWR 2005*. LNCS, vol. 3703, Springer, Heidelberg (2005)
- [26] Horrocks, I., Patel-Schneider, P.F.: Three theses of representation in the semantic web. In: *WWW 2003: Proceedings of the 12th international conference on World Wide Web*, pp. 39–47. ACM Press, New York (2003)
- [27] Kifer, M., Bruijn, J., Boley, H., Fensel, D.: A Realistic Architecture for the Semantic Web. In: Adi, A., Stoutenburg, S., Tabet, S. (eds.) *RuleML 2005*. LNCS, vol. 3791, pp. 17–29. Springer, Heidelberg (2005)
- [28] Nutt, G.J.: *Centralized and Distributed Operating Systems*. Prentice-Hall International Editions (1992)
- [29] Patel-Schneider, P.F.: A Revised Architecture for Semantic Web Reasoning. In: Fages, F., Soliman, S. (eds.) *PPSWR 2005*. LNCS, vol. 3703, pp. 32–36. Springer, Heidelberg (2005)
- [30] SIIA (Last accessed 21/9/2006), Website <http://www.siiia.net/>
- [31] W3C. W3C Technical Architecture Group (TAG) (2006) (Last accessed 4/10/2006), W3C Web site <http://www.w3.org/2001/tag/>
- [32] Zimmermann, H.: OS1 Reference Model - The ISO Model of Architecture for Open Systems Interconnection. *IEEE Transactions on Communications* 71, 1334–1340 (1980) (Last accessed 26/9/2006), http://www.comsoc.org/livepubs/50_journals/pdf/RightsManagement_eid=136833.pdf