The Semantic Web identity crisis: in search of the trivialities that never were

Ruben Verborgh* and Miel Vander Sande

Department of Electronics and Information Systems, Ghent University – imec, Belgium

E-mail: ruben.verborgh@ugent.be

Abstract. For a domain with a strong focus on unambiguous identifiers and meaning, the Semantic Web research field itself has a surprisingly ill-defined sense of identity. Started at the end of the 1990s at the intersection of databases, logic, and Web, and influenced along the way by all major tech hypes such as Big Data and machine learning, our research community needs to look in the mirror to understand who we really are. The key question amid all possible directions is pinpointing the important challenges we are uniquely positioned to tackle. In this article, we highlight the community’s unconscious bias toward addressing the Paretonian 80% of problems through research—handwavingly assuming that trivial engineering can solve the remaining 20%. In reality, that overlooked 20% could actually require 80% of the total effort and involve significantly more research than we are inclined to think, because our theoretical experimentation environments are vastly different from the open Web. As it turns out, these formerly neglected “trivialities” might very well harbor those research opportunities that only our community can seize, thereby giving us a clear hint of how we can orient ourselves to maximize our impact on the future.

Keywords: vision, Web, semantics

1. Back to the future

Re-reading the original Semantic Web vision [1] from 2001, we tend to immediately notice where the predictions went wrong. Far less obvious are those that came true; they have become given in today’s world, part of the new normal that now forms our everyday reality. We have forgotten the era ruled by the Nokia 3310, whose monochrome screen’s resolution only covers a fraction of modern app icons, years before many people had Internet access at home—let alone on their phone. The crazy thing was imagining that we would be instructing our mobile devices to perform actions for us; the planning and realization of said actions were plausibly explained in the rest of the article. With the unimaginable eventually being solved after a decade of research, the imaginable may have turned out to be the toughest nut to crack.

The roots of the Semantic Web can be traced back to the initial Web proposal [2], whose opening diagram presents what we now refer to as a knowledge graph.

*Corresponding author. E-mail: ruben.verborgh@ugent.be.
2. A little semantics

The term Semantic Web evidently coincides with adding *semantics* to Web content to improve comprehension by machines. However, after two decades of debate, we still seem uncertain about exactly how much semantics are in fact useful. The writing on the wall is the disconnect between the data that is published and the applications that should consume them: the call for Linked Data has brought us the eggs, but the chickens that were supposed to hatch them are still missing.

To intertwine data with meaning, we largely rely on *rdfs* for exchange and interoperability. But what is really there is only factual knowledge in a (hyper)graph structure, with *uris* to uniquely identify terms. The intended meaning of the data is captured through knowledge representation ontologies such as *rdfs* or *owl*, and can be discovered for example through dereferencing. Hence, data shared as *rdf* *refers* more to its semantics than it contains them. And distributing those semantics has turned out significantly harder than distributing data.

Early efforts were heavily devoted to the development of ontology engineering, and understandably so. Having generic software to automatically act on a variety of independent datasets was what made the Semantic Web vision so appealing. Once domain knowledge had been formalized, it could be applied to represent facts, upon which reasoners could automatically derive new facts. Yet once we took those endeavors to the Web, it became apparent we had missed the general practical implications of the chosen direction. As semantics are always consensus-based, domain models are only as valuable as the scope of the underlying consensus. Hence, its usage cannot be guaranteed by parties that were not involved or disagree with the consensus. Often, these parties resort to mitigation strategies that disregard the semantics settled in description logic, such as selectively reusing properties and classes when publishing data, or freely reinterpreting the semantics through programming when consuming data.

Core frameworks such as *rdfs* and *owl* have also frequently been labeled as "by academics, for academics" because of their perceived complexity by developers. Due to a lack of deeper understanding and an inability to connect with existing development practice, ontologies are in practice often dumbed down to *vocabularies*—a term that is used more and more, basically stripping the data from semantics and once again leaving it up to individual applications. The major search engines backing Schema.org is illustrative of this fact, but also the increasing popularity of the shape languages *shacl* and *sirex*. They cover an important gap between data in the wild and applications: they need to know what data to expect, which was one of the things neglected by our fixation on descriptive logic.

The paradox between the use of semantics and the effort to provide it, cultivated a heterogeneous, *sparql*-based, and underspecified Web of Data [7]. Practical implementation and usability has too often been hand-
3. Where is the Web?

What arguably sets us apart besides semantics is, well, the Web. In contrast to relational or other databases, our domain of discourse is infinite and unpredictable on multiple levels. Because of the open-world assumption, no single RDF document contains the full truth. Even worse, any sufficiently large collection of Web documents will contain contradictions that, under classical logic, allows us to derive any truth—henceforth to be referred to as ex Tela quodlibet. Not only can anything be proven from a contradiction, in these days of fake news and dubious political advertising, it has never been easier to find self-consistent documents online in support of virtually any given conclusion or its opposite.

The Web is what we deliver as an answer to any Linked Data skeptic, as an irrefutable argument that all of our perceived or actual complexity is justified, because we are dealing with problems that span the entire virtual address space of the globe and in fact the universe. The Web is the reason why our ontologies are spread all over the place, why the prefix expansion for the owl:ontology counts 30 characters, why foaf is forever stuck at version 0.9, the Dublin Core vocabulary at 3 different ones, and why we cannot all just use Schema.org. The Web is why Open Data exists, why our public sparql endpoints are down 1.5 days a month [10], why stable vocabularies suddenly disappear. Everything we do, we do it the way we do it, because the Web sets the rules such that anything more simple or logical would not do. If the Web is such a self-explanatory answer to the existence of our discipline—then why are we so afraid to put our work on top of it?

We are not even talking here about taking our scholarly communication to the Web; let that be the crusade of the dogfooders [11], to whom we dedicate Section 7. We mean to say that “it works in our university basement” has become an acceptable and applauded narrative—and to be fair to both the innocent and the guilty, impressive efforts undertaken in such basements have rightly been awarded scientific stamps of excellence through rigorous non-Web peer review processes. However, we cannot claim the Web as the sole source of our intricacies, while simultaneously ignoring all of the Web’s difficulties by conducting all of our experiments in hermetically controlled environments. By doing so, we pretend that the comfortable 80% cannot significantly be affected by the unpredictable impurities of the 20%, that an n-fold performance gain in our basements can implicitly be extrapolated to the same gain for Linked Data in general. As Goodhart’s law states: “When a measure becomes a target, it ceases to be a good measure”, except that we can strongly question whether non-Web environments, however pure and controlled, have ever fulfilled the role of good measure providers in the first place.

No, we cannot safely assume that the owl:sameAs predicate has consistently been used in accordance with at least one of its several meanings [12]. No, we cannot assume that sparql endpoints will be available or even...
return valid RDF. Yes, people will use the same URL to refer to different things, and obviously different URLs to point to the same things—without even throwing in as little as a semantically ambiguous schema: sameAS.

Yes, our precious data sets unnecessarily use different ontologies, so we have to switch on reasoning, even though that makes our results suddenly worse than the state of the art—and did we mention that one of those ontologies no longer dereferences but, even back when it still did, was not linked to the others anyway? Upon closer reflection, our fears about the Web are probably justified; our scientific conclusions and their presumed external validity perhaps a little less.

We are all aware that the Web is a good platform for data publication, but a pretty bad platform for data consumption [13]. Yet that exactly is the reason to not ignore the 20% any longer, but to embrace the unique challenges and opportunities it brings. Crucial and sometimes counterintuitive insights arise when Web-based techniques are applied to research problems previously only studied in isolation. As an example, link-traversal-based query execution [14] taught us that SPARQL queries can exist separately from specific interfaces to evaluate them, which in turn are independent from back-ends. Understanding that some of our standardized protocols do not adhere to the constraints of the Web’s underlying REST architectural style, allows us to design interfaces with better scalability properties, which might perform worse in closed environments but yield desirable properties on the public Web [15]. Taking this even further, we can wonder whether the default semantics of simple SPARQL queries are tailored too much to closed databases as opposed to the Web we publicly claim to target.

We should, however, not become too puristic in our judgment; an important aspect of scientific studies is their ability to zoom in on the isolated contribution of specific factors. Several valid use cases for non-Web RDF applications exist, so not every single undertaking has to embody the omnipotent role ascribed to the mythical Semantic Web agent. Nonetheless, as a community, we want to ensure we combine the 80% sufficiently often with the 20%, such that we obtain at least a more adequate impression of the potentially huge number of research questions hiding in plain sight.

4. “Linked” as bigger than “Big”

When Big Data became mainstream around 2010, the Semantic Web community was listening with great attention. After all, we had already been working with staggering numbers of facts, hundreds of millions of triples not being an exception. Furthermore, when considering all data on the Web as a whole, we would surely reach the threshold at which Linked Data should be considered Big Data in its own right.

However, Big Data and Linked Data are not necessarily structurally compatible. A main advantage of the RDF data model is that it allows for flexibility, enabling people to capture data that does not lend itself well to the rigid structures of spreadsheets or relational databases. Big Data solutions derive their strength from a rigorous, extensive schema, which strongly contrasts with RDF’s highly normalized triple format. While there have been solutions that leverage Big Data technologies to address RDF use cases such as querying [16], they require reformattting data to fit the Big Data paradigm.

A conceptual issue with the Big Data vision, at least for our purposes, is that it takes the path of the lowest common denominator, as a natural result of an aggregation process. While aggregation definitely has its merits for discovery and analysis, it also flattens unique characteristics and attributes of individual datasets, dissolving them into a much larger and more homogeneous space. An example of how this unintentionally can become troublesome is found within the Europena initiative [17], which serves the noble cause of aggregating highly diverse metadata from cultural institutions all across Europe. However, several individual institutions felt wronged when they had to send their data set—which they knew so well and had taken care of for so many years—only for it to be mingled with those of others who surely would have different accents and inferior quality thresholds [18]. What gives Big Data its attractiveness and efficiency might thus be removing what differentiates us. Time will tell if similar arguments can be made about the Wikidata project [19], which aims to be a global knowledge base.

For some time, we have been mildly apologetic about not doing Big Data, at one point hastily rebranding ourselves as “Semantics and Big Data” [20] before realizing that, indeed, there is another research community out there that is better positioned to tackle those challenges. Considering the 2001 article [1] as the official birth date of the Semantic Web, let us conveniently ignore those teenage years during which we should be forgiven for going through different phases that were all just part of constructing our own identity. We should not aspire to be that popular kid from high school, who, as it turned out later, had merely peaked early in life. Nearing our twenties now, let us stop apologizing already for just being ourselves.
If we conceptually think about Big Data versus what
we are aiming to achieve with Linked Data, our chal-
 lenges might very well be the harder ones. Notwith-
 standing impressive research and engineering efforts to
 scale up Big Data solutions the way they do, harvesting
 an enormous amount of homogeneous data in a single
 place creates ideal conditions for processing and anal-
 ysis. A small number of very large data sets is easier
to manage than a very large number of small data sets.
Size does matter, just not always in the way others think:
the heterogeneity and distribution of Linked Data is
currently at a level that cannot be adequately tackled
with Big Data techniques. Instead of being ashamed
about practicing Small Data, we should proudly flaunt
its multitude and diversity. In times of increasing calls
for inclusion, let this be a good thing.

Because even if we technically would be able to
centralize everything in one place, we could only serve
the relatively small space of public data, not all of
the private data that is the focus point of Big Data
applications. After all, there are very good reasons for
data to live in different places, not in the least legal or
privacy concerns. Those needs are only becoming more
pressing, given important drivers such as the GDPR legal
framework in Europe, and a strong world-wide call for
more privacy and control over personal data. By keeping
data in millions of small personal data stores close to
people, we are in a much better position to safeguard
people’s most precious digital assets. The challenge
then of course is in connecting these distributed pieces
of data at runtime, which the Solid project [21] does
through Linked Data.

In a distributed future, there will not be less data,
but more; if it cannot reside in one place for whatever
reason, it will have to be linked. This is yet another
reason why we need to be prepared for Web-scale dis-
covery and querying over federations that are magni-
tudes more challenging than our current experimental
environments.

5. AI beyond ML

There is no question the age of Deep Learning is very
much upon us. As the last one to mature, deep learning
has spawned numerous research efforts, techniques,
and even production-ready applications with machine
learning, elevating the state of AI once again. Semantic
Web research has not been resilient to the siren song,
and started exploiting RDF knowledge bases as fertile
soil for Deep Learning and other machine learning
approaches. The popular topics that emerged, such
as embeddings [22] and concept learning [23] enable
model training from description logics to complete and
extend any semantic information present. Developing
such approaches is crucial to reduce the high manual
currently required for participating in the Semantic Web.

Semantic technologies were originally considered
part of the AI family and in essence still are [24]. In-
ference of logical consequences from data can drive a
machine’s autonomy. Yet in the shadow of advanced
machine learning, the “cool kids” perceive us as apostles
of an old, inflexible, and outdated rule-based approach.
However, maturity in the machine learning field also un-
covered the gaps where semantic technology can prove
its relevance. Use cases prone to decision accuracy,
such as healthcare or privacy enforcement, profit from
the exact outcomes of first-order logic. Furthermore,
the ability of some semantic reasoners to explain their
actions through proofs [25] is a much desired trait by
the primarily black-box machine learning methods.

As both angles have their merits, the future is very
likely hybrid, and we need to further explore compli-
mentary roles. For instance, semantics and inference
can pre-label data that improve the accuracy of models.
Or, post-execution explainability could be achieved by
reasoning over semantic descriptions of nodes. In the
area of digital assistants, such as the promising work
with Almond [26] and Snips [27], declarative AI can
append a human representation of the world to represen-
tations trained on raw data. This would fill knowledge
gaps of current assistants such as Siri and Alexa, in-
crease their associative ability, and eventually improve the
authenticity of their interactions. Some more funda-
mental questions also need to be answered, such as train-
ing a model under the open world assumption. Fitting
strategies exist, but there are many more unknowns.

Semantic inference and first-order logic might lead to
less spectacular conclusions, but they will be nonethe-
less crucial to advanced machine learning systems. Also
here, it is important to solve the engineering side of
things. Almond and Snips are directly usable to de-velopers, who, through testing, discover further challenges.
When machine learning solutions “just work” develop-
ers do not need to know what is inside, that is the result
of research, not just engineering. Getting rid of the “triv-
ial” problems with semantic inference hopefully means
providing these more spectacular results, on the Web.
Maybe this is the better way to position ourselves in one
of the next waves to come: reinforcement learning.
6. Challenging until proven trivial

Ultimately, all of this shows that we need to guard ourselves from conducting research in a vacuum. Not all science requires practical purposes, but if we would only design solutions for problems that will never even exist if the Semantic Web does not take off any further, then we should at least consider prioritizing those urgent problems that are blockers to adoption. Part of our hesitation might be that, having fought hard for recognition as a scientific domain, we are afraid to be pushed back into the corner of engineering. Our conferences and journals tend to have a high threshold for what qualifies as research, with a strong focus on qualitative experimentation. While high thresholds in general are commendable, they also result in a higher percentage of false negatives, both in submitted works that never get accepted, and in stellar research ideas that never materialize because fear of such rejections encourages safer bets.

We tend to zoom in on very focused, often incremental research problems, which tend to bring us progress. Again Pareto’s law from Fig. 1 lures around the corner: we consider the core 80% of a hard problem and assume that the remaining 20% is a non-issue. Converting technological research into digestible chunks for developers is considered trivial and outside of our scientific duty. Everything that reeks of pure engineering is shunned.

However, most researchers in our community have not built a single Semantic Web app, so we cannot pretend to understand the insights of that 20%. It is impossible to tell whether the remainder is trivial or not; and many of the experiences above reveal that some of the most complex research problems appear exactly there. But how would we know? We do not get in touch with some of the most pressing issues, because we already ruled them out as trivial, and then wonder about the low adoption of the otherwise excellent 80% research.

Since the Semantic Web started, Web development has massively changed. Many apps are now built by front-end developers, for whom Semantic Web technologies are inaccessible, explaining the success of substantially less powerful but far more developer-friendly technologies such as GraphQL. We compensate by drawing those back into the research domain [28], but gloss over a crucial point: bringing SPARQL levels of expressivity to front-end developers is in fact a research problem.

Designing an appropriate Linked Data developer experience [29] is so challenging because, while regular apps are hard-coded against one specific well-known back-end, Linked Data apps need to expect the unexpected as they interface with heterogeneous data from all over the Web. Building such complex behavior involves a sophisticated integration of many branches of our research, which requires designing and implementing complex program code. Exposing such complex behavior into simple primitives, as is needed for front-end developers, requires automating the generation of that complex code, likely at runtime. Such endeavours have not been attempted at the research level, let alone they would be ready for implementation by skilled engineers.

This research gap between current research solutions and practice means that much of our work cannot be directly applied. We could deem it acceptable that nothing works in practice yet. Unfortunately, such a lax attitude leaves us with an all too comfortable hiding spot: why would my research have to work in the real world if others’ does not? As a direct consequence of this line of thought, we cannot meaningfully distinguish research that could eventually work from research that never will.

Until we have examined whether or not something is trivial, we should not make any implicit assumptions. We have been wrong before. Perhaps we should consider scoring research works on the 80/20 Pareto scale, and ensure that we have enough of both sides at our conferences and in our journals. By also judging applicability, we abandon our filter bubbles and extend our action radius to urgent problems in the way of adoption—which will only enlarge our research community.

7. Practice what we preach

Not only do many of us lack Semantic Web experience as app developers, our even bigger gap is experience as users. Although a significant amount of our communication (not in the least toward funding bodies) consists of technological evangelism, we rarely succeed in leveraging our own technologies. If we keep on finding excuses for not using our own research outcomes, how can we convince others? The logicians among us will undoubtedly recognize the previous statement as a tu quoque fallacy: our reluctance to dogfood is factually independent of our technology’s claim to fame. Yet if all adoption were solely based on sound reasoning, our planet would look very different today. Credibility and fairness aside, we are not in the luxury position to tell others to “do as I say, not as I do.” The burden of proof is entirely upon ourselves, and the required evidence extends beyond the scientific.
In addition to being an instrument of persuasion, dogfooding addresses a more fundamental question: which parts of our technology are ready for prime time, and which parts are not? By becoming users of our own technologies, we will gain a better understanding of the elusive 20% that clearly, had it actually been so trivial, would already have been there. Never underestimate the power of frustration: feeling frustrated about unlocked potential is what prompted Tim Berners-Lee to invent the Web [30]. Only by managing almost his entire life with Linked Data, he is able to keep a finger on the Semantic Web’s pulse, and his eyes on its Achilles’ heel.

If we similarly had a deeper understanding of real-world Linked Data flows and obstacles, would we not be in a better position to make a difference? We might want to address concrete problems happening today, in addition to targeting those that will hopefully arise—conditional on today’s problems ending up solved—after several more years.

8. In conclusion

After almost two decades, the Semantic Web should step out of its identity crisis into adolescence. In search of a target market for adoption, research in semantic technologies has ridden others’ waves all too often, in an attempt to assimilate with all use cases but our own. This brought us as a community into a disconnect with the place where we can make a difference: the Web. There, new technologies still emerge every day—just not ours. Investing in theoretically interesting problems without also delivering the necessary research to achieve practical implementations seems to have singled us out.

A Semantic Web has data and semantics intertwined, yet distributing those semantics has been proven to be harder than sharing data. Can we focus on the practice and implications of sharing and preserving semantics? If not, we might leave the original vision to die in the hands of a more short-term and pragmatic agenda. No doubt, the need for full-scale data integration will eventually reappear, possibly reinventing the solutions and methods we are working on today. But that realization might take another decade to surface.

The Web might not be our only target market, but it is the one that sets us apart. Yet it does not pop up in the average “treats to validity” section—if there even is one. The rules are set in a unique way, which requires overcoming specific hurdles to make things work. To really test the external validity of our work, we should submerge in the practical side of things and thus make the Web a better suited place for data consumption. Our experimental environment should not be that of Big Data. We should thrive with a lot of small datasets instead of a few large ones, and in heterogeneity instead of homogeneity. We could differentiate ourselves as the main driver for the much needed re-decentralization of the Web, where, backed by privacy and data legislation, Web-scale federation is the next Big thing. To this end, positioning semantic technologies as compliment to machine learning is a necessity. The future of AI is hybrid: descriptive logic can bring accuracy, explainability and, of course, meaningful data to the table.

In order to succeed, we will need to hold ourselves to a new, significantly higher, standard. For too many years, we have expected engineers and software developers to take up the remaining 20%, as if they were the ones needing to catch up with us. Our fallacy has been our insistence that the remaining part of the road solely consisted of code to be written. We have been blind to the substantial research challenges we would surely face if we would only take our experiments out of our safe environments into the open Web. Turns out that the engineers and developers have moved on and created their own solutions, bypassing many of the lessons we have learned, because we stubbornly refused to acknowledge the amount of research needed to turn our theories into practice. Since we seemingly did not want the Web, more pragmatic people took over.

And if we are honest, can we blame them? Clearly, the world will not wait for us. Let us not wait for the world.

References

A. Bernstein, C. Welty, C. Knoblock, D. Vrandečić, P. Groth,
N. Noy, K. Janowicz and C. Goble, eds, Springer International
Publishing, Cham, 2014, pp. 245–260. ISBN 978-3-319-11964-
9.

[8] W. Beek, J. Raad, J. Wielemaker and F. van Harmelen,


