Semantic Web 0 (0) 1
IOS Press

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

Editors: Pascal Hitzler, Kansas State University, Manhattan, KS, USA; Krzysztof Janowicz, University of California, Santa Barbara, USA
Solicited reviews: Aidan Hogan, Universidad de Chile, Chile; Armin Haller, Research School of Computer Science, Australian National University, Canberra, Austrlia; Axel Polleres, Vienna University of Economics and Business (WU Wien), Austria

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. If we are hesitant to step up, more pragmatic minds will gladly reinvent technology for the real world, only covering a fraction of the opportunities we dream of.

Keywords: vision, Web, semantics

1. Back to the future

Re-reading the original Semantic Web vision [6] from 2001, we immediately notice where the predictions went wrong. Far less obvious are those that came true; they have become givens in today's world, part of the *new normal* that now forms our everyday reality. We have forgotten the era ruled by the indestructible Nokia 3310, whose monochrome screen barely counted more pixels than a modern-day app icon, years before most 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 those 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 Semantic Web's roots can be traced further back to the initial Web proposal [1], whose opening diagram presents what we now refer to as a *knowledge graph*, an early glimpse into subject–predicate–object triples rather than the URL–HTTP–HTML triad that would ultimately become the Web. That same Web is currently facing severe threats [3–5], having rapidly gone from a utopian harbor of permissionless innovation to a potentially dystopian environment controlled by only a handful of dominant actors. The Semantic Web seems unaffected by most of this, strangely—until we realize that the Web and the Semantic Web have silently split ways not too long after the first RDF specifications appeared.

Nonetheless, semantic technologies are regularly coined as a means of tackling some of the Web's most pressing challenges, such as combatting disinformation or fueling its re-decentralization movement [25].

*Corresponding author. E-mail: ruben.verborgh@ugent.be.

Meanwhile, the Semantic Web research community is facing its own battles with some of the latest technological hypes, doubting between defending its own relevancy next to Big Data, machine learning, and blockchain, or surfing atop the waves created by those. If you can't beat them, join them; if you can't join them, repackage. The days when the keyword "semantics" led to guaranteed project funding have faded faster from our collective memory than the Nokia 3310 ever will.

Granted, cracks have started creeping into these other technologies, too. Maybe Big Data is not limitless in practice if technical capabilities scale faster than the human and legal processes for ethical data management, and we do need to link data across distributed sources instead of unconditionally aggregating them. Perhaps there are problems that machine learning can never solve reliably, and the safety provided by first-order logic proofs is irreplaceable for crucial decisions. And possibly it will turn out that decentralized consensus only touches a small part of all use cases, that disagreement under the "anyone can say anything about anything" flag provides a more workable model of the virtual world.

So when we are not riding others' waves, what is it that unites the Semantic Web research community? What makes us truly "us", what are the semantics we can attach to our own identity? Having emerged at the intersection of the Web, databases, and logic, we have since become disconnected from these domains, our awareness of which sometimes appears to be frozen in time. We tend to disregard that the Web from which we spun off is no longer the same as it was, and that different approaches are required today. We have held on to XML and RPC longer than most, confused the ends with the means that were supposed to achieve them.

The main danger within an existential crisis is the risk of losing our connection to the reality from which we originate. The philosophy of our community seems to align with Alan Kay's quote that *"The best way to predict the future is to invent it."* We build and we investigate, expecting the future to wrap its arms around the creations we are spawning. In this vision article, we rather embrace John Perry Barlow's inversion of the quote, in which *"The best way to invent the future is to predict it."* Looking back at the dreams from the past and recombining those with the aspirations of the present, what are the essential missing pieces that require our unique dedication as Semantic Web scholars? As in the original Semantic Web article, those topics that have long been considered trivial might very well be the hardest ones in practice [20].

In this article, we make the case for a return to our roots of "Web" and "semantics", from which we as a Semantic Web community—what's in a name—seem to have drifted in search for other pursuits that, however interesting, perhaps needlessly distract us from the quest we had tasked ourselves with. In covering this journey, we have no choice but to trace those meandering footsteps along the many detours of our community—yet this time around with a promise to come back home in the end.

2. A little semantics

The term "Semantic Web" evidently coincides with adding *semantics* to Web content in order to improve interpretation by machines. However, after two decades of debate, we still seem uncertain about exactly how much semantics are in fact useful. The gap between data that are published and applications that should consume them continues to grow. While the call for Linked Data has brought us the eggs, the chickens that were supposed to be hatching them are still missing, partly because making sense of others' data remains hard.

To intertwine data with meaning, we rely on RDF's capabilities for exchange and interoperability. But what is out there is 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 through dereferencing. In that sense, data in RDF actually *refer* to their semantics rather than *containing* them. And distributing those semantics has turned out significantly harder than distributing data.

Early efforts were devoted to the development of ontology engineering, and understandably so. Having generic software to automatically act on a variety of independent data sets was what made the Semantic Web vision so appealing. Once domain knowledge had been formalized, it could be applied to represent facts, from 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. As semantics are always consensus-based, domain models are only as valuable as the scope of the underlying consensus. Hence, their usage cannot be guaranteed by parties that were not involved or disagree with the consensus. Often, people resort to mitigation strategies that disregard the semantics enshrined in description logic, by selectively reusing properties

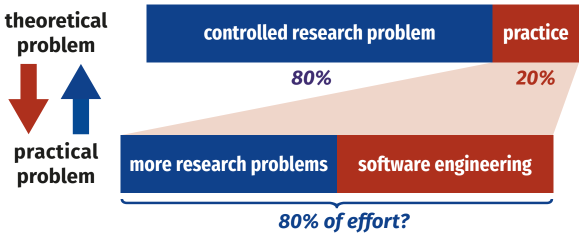

Fig. 1. After having solved the core 80% of a research problem, we often assume that the remaining 20% are practicalities that can be addressed through trivial engineering. In reality, lifting research from controlled experimental environments to the open Web likely leads to other research problems. In addition to bringing problems from theory to practice, we can let practical problems inspire theory.

and classes upon publication, or freely reinterpreting semantics upon consumption.

Core frameworks such as RDF and OWL are sometimes labeled as "by academics, for academics" because of their perceived complexity by developers. Due to a lack of deeper understanding and a shortage of connections to existing development practice, ontologies are in practice often reduced to more prescriptive *vocabularies* that basically again leave semantics up to individual applications. The desire for more simple choices with less flexibility is illustrated by the backing of Schema.org by the major search engines and the increasing popularity of the shape languages SHACL and ShEx. They cover an important gap between data in the wild and applications that need to know what kind of data to expect—one of the aspects we probably want to keep our eyes on.

The disconnect between the need of semantics and the effort to provide it, has cultivated a *heterogeneous* and *underspecified* Web of Data [19]. We cannot afford any longer to handwavingly address practical implementation and usability with deep theories. As depicted in Fig. 1, a strong implicit assumption underlies a lot of our work: that solving the core 80% of a problem is where research is needed, and that the remaining 20% consist of simple engineering to take that research from theory to practice. However, is what we often dismiss as "engineering" *really* just a matter of writing more code? As scientists, we might want to validate that hypothesis, given the considerable problems that arise when we try to deploy semantics at Web-scale.

We need to consider the Web we *have*, before we can have the Web we *want*. After all, what good is high-performance inferencing if ontologies cannot be found or are outdated? What good are unique identifiers for concepts when stating equality with `owl:sameAs` is inadequate for applications [11]? How realistic is SPARQL as a *universal* query language if queries in practice have to be tailored to specific endpoints, because reasoning is only ever switched on in theory? Meanwhile, enterprises and developers start to give up on the formal semantics, and we risk the baby being thrown out with the bath water. That is the logical result if we leave the completion of the bigger Semantic Web picture to companies with a deadline. Their enthusiastic endorsement of shapes, for instance, could eventually suppress the practice of semantics in data. Researchers understand "a little semantics goes a long way" [14] to not necessarily mean that less semantics would be better than more. But exactly how much is too much for the actual Web? Only through research we can find out.

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 settle on Schema.org. The Web is why Open Data exists, why our public SPARQL endpoints are down 1.5 days a month [7], why stable vocabularies suddenly disappear. Everything we do, we do it the way we do, 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 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 [8], 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 directly 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, pure and controlled as they are, 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 [11]. No, we cannot assume that SPARQL endpoints will be available or even return valid RDF, or that any RDF document out there is syntactically valid, coherent, and free of ontological abuse [15]. 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 benchmark 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 testing on the Web are probably justified; our scientific conclusions and their presumed external validity perhaps a little less.

In all honesty, the academic community did take its *publish or perish* adage to heart, and is co-responsible for the billions of RDF triples currently on the public Web as Linked Open Data. But while the Web is a good platform for data publication, it is a pretty bad platform for data consumption [22], which is not coincidentally also where many challenges remain open. This is why we should not ignore the 20% any longer, but 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 [12] 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 [27]. Taking this even further, we can wonder whether the greedy semantics of SPARQL queries are tailored too much to closed databases as opposed to the Web we claim to target. Are we ready for that Web?

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. Many 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 on the Web.

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 columnar structures of spreadsheets and relational databases. Big Data solutions derive their strength from a strict and rigid structure, which strongly contrasts with RDF's virtually unbounded freedom. While there have been solutions that leverage Big Data technologies to address RDF use cases such as querying [18], they require reformatting 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 data sets, dissolving them into a much larger and more homogeneous space. An example of how this unintentionally can become troublesome is found within the Europeana initiative [16], 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 upload 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 [24]. What gives Big Data its attractiveness and efficiency might thus take away what differentiates us. Time will tell if similar arguments can be made about the Wikidata project [28], 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" [9] before realizing that, indeed, there is another research community out there that is better positioned to tackle those challenges. Considering the 2001 article [6] as the official birth date of the Semantic Web, let us conveniently ignore those teenage years during which we should be forgiven for rapidly cycling through different phases as we were in fact just 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 challenges might very well be the bigger ones. Notwithstanding 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 analysis. 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 choice and control over personal data. By keeping each individual's data close by in a small personal store, people will be in a much better position to safeguard their most precious digital assets. The challenge then of course is in connecting these distributed pieces of data at runtime, which the Solid project [17] 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 discovery and querying over federations that are magnitudes 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 latest 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. Popular topics that emerged, such as *embeddings* and *concept learning* enable model training from description logics to complete and extend any semantic information present. Developing such approaches reduces the high manual effort currently required for participating in the Semantic Web.

Semantic technologies were originally considered part of the AI family and in essence still are [10]. Inference 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, maturation in the machine learning field also uncovered 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 [26] 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 complimentary roles. For instance, semantics and inference can pre-label data to improve the accuracy of models. Post-execution explainability could be achieved by reasoning over semantic descriptions of nodes. In the area of personal digital assistants, declarative AI can append a human representation of the world to representations trained on raw data. This would fill knowledge gaps of current assistants such as Siri and Alexa, increase their associative ability, and eventually improve the authenticity of their interactions. Some more fundamental questions also need to be answered, such as training a model under the open world assumption. Appropriate strategies exist, but there are many more unknowns.

Semantic inference and first-order logic might lead to less spectacular conclusions, but they will nonetheless be crucial to advanced machine learning systems. Also here, it is important to solve the engineering side of things. Several machine learning tools are readily available for developers, who, through testing, discover further challenges. When machine learning solutions "just work", developers do not need to know what is inside; importantly, such simplicity is the result of *research*, not just engineering. Getting rid of the "trivial" problems with semantic inference hopefully means providing these more spectacular results, on the Web. Maybe this is the better way to position ourselves in the next waves to come, such as reinforcement learning.

6. Challenging until proven trivial

Ultimately, all of above indicates a need to guard ourselves from conducting research in a vacuum. Not all science requires practical purposes, but many of the research problems we study will never actually occur if the Semantic Web does not take off any further, so we should at least consider—for our own sake—prioritizing those urgent problems that are blockers to its adoption. Part of our hesitance might be that, having fought hard for recognition as a scientific domain, we are afraid to be pushed back into the corner of engineering. We usually zoom in on very focused, often incremental research problems, which tend to bring us progress. Our conferences and journals strive to find a high threshold for what qualifies as research, with a strong focus on qualitative experimentation. Thereby, we risk optimizing for familiarity and purity rather than for originality and impact, because the scientific merits of novel directions are inherently much harder to assess. 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.

As much time as we spend justifying ourselves toward other communities, those efforts sometimes pale to how our reviewers expect authors to justify their choice to address pragmatic concerns that, all things considered, should be no less of a scientific contribution. 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, despite the considerable scientific challenges of creating simple abstractions to complex technology, as the machine learning community shows time and time again.

Yet everything that reeks of engineering is shunned. However, most researchers in our community have not built a single Semantic Web app, so we cannot pretend to understand the insides of the 20%. As such, it is impossible to tell whether that remainder is trivial or not. 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 reasons for 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. The GraphQL community, who have prided themselves on simplicity compared to the Semantic Web technology stack, are slowly discovering that they were merely solving simpler problems. Queries with local semantics indeed become problematic if data needs to come from multiple sources. Instead of reusing the lessons from years of SPARQL federation research, the GraphQL community rather reinvents ontologies by calling them "schema stitching" [21]. Persisting on the pragmatic road, which they initially took because our alternative was deemed too complex, they might ironically end up with something as difficult but less powerful, because they did not have the same forethought. Even more ironic is that we remain stuck in that forethought and wonder when adoption is coming. We compensate by drawing such new technologies back into the research domain [13], 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* [23] 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 would they be ready for implementation by skilled engineers.

This research gap between current research solutions and practice means that much of our work cannot be applied. Some find 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. Perhaps we should consider scoring manuscripts 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 grow 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 [2]. 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 perhaps a little too often. While those bring in useful lessons to be learned, we should not forget to learn our own on the place where we can make a major 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 hard. 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.

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 "threats to validity" section of articles—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 the same as that of Big Data; we should thrive with a lot of small data sets 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 a complement 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 are creating their own solutions, bypassing many of the lessons we already learned, because we stubbornly refused to acknowledge the amount of research needed to turn our theories into practice. As we were not ready for the Web, more pragmatic people started taking 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

[1] Tim Berners-Lee. Information management: A proposal. Technical report, CERN, March 1989. https://www.w3.org/History/1989/proposal.html.

[2] Tim Berners-Lee. The next Web, 2009. https://www.ted.com/talks/tim_berners_lee_on_the_next_web.

[3] Tim Berners-Lee. Three challenges for the Web, according to its inventor. Web Foundation, March 2017. https://webfoundation.org/2017/03/web-turns-28-letter/.

[4] Tim Berners-Lee. The Web is under threat. join us and fight for it. Web Foundation, March 2018. https://webfoundation.org/2018/03/web-birthday-29/.

[5] Tim Berners-Lee. 30 years on, what's next #ForTheWeb? Web Foundation, March 2019. https://webfoundation.org/2019/03/web-birthday-30/.

[6] Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, May 2001. DOI: 10.1038/scientificamerican0501-34.

[7] Carlos Buil-Aranda, Aidan Hogan, Jürgen Umbrich, and Pierre-Yves Vandenbussche. SPARQL Web-querying infrastructure: Ready for action? In Harith Alani, Lalana Kagal, Achille Fokoue, Paul Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha Noy, Chris Welty, and Krzysztof Janowicz, editors, *Proceedings of the 12th International Semantic Web Conference*, volume 8219 of *Lecture Notes in Computer Science*, pages 277–293, Heidelberg, Germany, 2013. Springer. DOI: 10.1007/978-3-642-41338-4_18.

[8] Sarven Capadisli. *Linked Research on the Decentralised Web*. PhD thesis, University of Bonn, 2019. https://csarven.ca/linked-research-decentralised-web.

[9] Philipp Cimiano, Oscar Corcho, Valentina Presutti, Laura Hollink, and Sebastian Rudolph, editors. *The Semantic Web: Semantics and Big Data*, volume 7882 of *Lecture Notes in Computer Science*, Heidelberg, Germany, 2013. Springer. DOI: 10.1007/978-3-642-38288-8.

[10] Harry Halpin. The Semantic Web: the origins of artificial intelligence redux. In *Third International Workshop on the History and Philosophy of Logic, Mathematics, and Computation*, 2004.

[11] Harry Halpin, Patrick J. Hayes, James P. McCusker, Deborah L. McGuinness, and Henry S. Thompson. When owl:sameAs isn't the same: An analysis of identity in Linked Data. In Peter F. Patel-Schneider, Yue Pan, Pascal Hitzler, Peter Mika, Lei Zhang, Jeff Z. Pan, Ian Horrocks, and Birte Glimm, editors, *Proceedings of the 9th International Semantic Web Conference*, Lecture Notes in Computer Science, pages 305–320, Heidelberg, Germany, 2010. Springer. DOI: 10.1007/978-3-642-17746-0_20.

[12] Olaf Hartig, Christian Bizer, and Johann-Christoph Freytag. Executing SPARQL queries over the Web of Linked Data. In Abraham Bernstein, David R. Karger, Tom Heath, Lee Feigenbaum, Diana Maynard, Enrico Motta, and Krishnaprasad Thirunarayan, editors, *Proceedings of the 8th International Semantic Web Conference*, pages 293–309, Heidelberg, Germany, 2009. Springer. DOI: 10.1007/978-3-642-04930-9_19.

[13] Olaf Hartig and Jorge Pérez. Semantics and complexity of GraphQL. In Pierre-Antoine Champin, Fabien Gandon, Mounia Lalmas, and Panagiotis G. Ipeirotis, editors, *Proceedings of the 2018 World Wide Web Conference*, pages 1155–1164, Geneva, Switzerland, 2018. ACM. DOI: 10.1145/3178876.3186014.

[14] James Hendler. The dark side of the Semantic Web. *IEEE Intelligent Systems*, 22(1):2–4, 2007. DOI: 10.1109/MIS.2007.17.

[15] Aidan Hogan, Andreas Harth, Alexandre Passant, Stefan Decker, and Axel Polleres. Weaving the Pedantic Web. In Christian Bizer, Tom Heath, Tim Berners-Lee, and Michael Hausenblas, editors, *Proceedings of the 3rd Workshop on Linked Data on the Web*, volume 628 of *CEUR Workshop Proceedings*, Aachen, Germany, 2010. CEUR-WS.

[16] Antoine Isaac and Bernhard Haslhofer. Europeana Linked Open Data – data.europeana.eu. *Semantic Web Journal*, 4(3):291–297, July 2013. DOI: 10.3233/SW-120092.

[17] Essam Mansour, Andrei Vlad Sambra, Sandro Hawke, Maged Zereba, Sarven Capadisli, Abdurrahman Ghanem, Ashraf Aboulnaga, and Tim Berners-Lee. A demonstration of the Solid platform for social Web applications. In *Companion Proceedings of the 25th International Conference on World Wide Web*, pages 223–226, Geneva, Switzerland, 2016. ACM. DOI: 10.1145/2872518.2890529.

[18] Alexander Schätzle, Martin Przyjaciel-Zablocki, Antony Neu, and Georg Lausen. Sempala: Interactive SPARQL query processing on Hadoop. In Peter Mika, Tania Tudorache, Abraham

Bernstein, Chris Welty, Craig Knoblock, Denny Vrandečić, Paul Groth, Natasha Noy, Krzysztof Janowicz, and Carole Goble, editors, *Proceedings of the 13th International Semantic Web Conference*, volume 8796 of *Lecture Notes in Computer Science*, pages 164–179, Heidelberg, Germany, 2014. Springer. DOI: 10.1007/978-3-319-11964-9_11.

[19] Max Schmachtenberg, Christian Bizer, and Heiko Paulheim. Adoption of the Linked Data best practices in different topical domains. In Peter Mika, Tania Tudorache, Abraham Bernstein, Chris Welty, Craig Knoblock, Denny Vrandečić, Paul Groth, Natasha Noy, Krzysztof Janowicz, and Carole Goble, editors, *Proceedings of the 13th International Semantic Web Conference*, volume 8796 of *Lecture Notes in Computer Science*, pages 245–260, Heidelberg, Germany, 2014. Springer. DOI: 10.1007/978-3-319-11964-9_16.

[20] Clay Shirky. The Semantic Web, syllogism, and worldview, 2003. http://www.shirky.com/writings/herecomeseverybody/semantic_syllogism.html.

[21] Sashko Stubailo. The next generation of schema stitching, 2018. https://blog.apollographql.com/the-next-generation-of-schema-stitching-2716b3b259c0.

[22] Frank van Harmelen. 10 years of Semantic Web: does it work in theory?, 2011. https://www.cs.vu.nl/~frankh/spool/ISWC2011Keynote/.

[23] Ruben Verborgh. Designing a Linked Data developer experience, 2018. https://ruben.verborgh.org/blog/2018/12/28/designing-a-linked-data-developer-experience/.

[24] Ruben Verborgh. One flew over the cuckoo's nest – The role of aggregation on a decentralized Web, 2018. https://rubenverborgh.github.io/EuropeanaTech-2018/.

[25] Ruben Verborgh. Re-decentralizing the Web, for good this time. In Oshani Seneviratne and James Hendler, editors, *Linking the World's Information: Tim Berners-Lee's Invention of the World Wide Web*, pages 225–237. ACM, 2020.

[26] Ruben Verborgh, Dörthe Arndt, Sofie Van Hoecke, Jos De Roo, Giovanni Mels, Thomas Steiner, and Joaquim Gabarro. The pragmatic proof: Hypermedia API composition and execution. *Theory and Practice of Logic Programming*, 17(1):1–48, 2017. DOI: 10.1017/S1471068416000016.

[27] Ruben Verborgh, Miel Vander Sande, Olaf Hartig, Joachim Van Herwegen, Laurens De Vocht, Ben De Meester, Gerald Haesendonck, and Pieter Colpaert. Triple Pattern Fragments: a low-cost knowledge graph interface for the Web. *Journal of Web Semantics*, 37–38:184–206, March 2016. DOI: 10.1016/j.websem.2016.03.003.

[28] Denny Vrandečić and Markus Krötzsch. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 57:78–85, 2014. DOI: 10.1145/2629489.