Editor: **Amit Sheth,** Kno.e.sis at Wright State University, amit@knoesis.org

An Open Semantic Framework for the Industrial Internet of Things

Simon Mayer, Jack Hodges, Dan Yu, Mareike Kritzler, and Florian Michahelles, *Siemens Corporate Technology*

Driven by academia[1] and fueled by government funding,[2] research and development in the Internet of Things (IoT) initially focused on basic enabling technologies and architectures for the identification, networking, and discovery of smart devices. Today, almost two decades later, big players from a wide range of industries have jumped on the bandwagon and base their future business and growth prospects on the IoT. At the same time, industrial players, standardization organizations, and academia are converging with respect to the IoT's basic technologies. To a large extent, the IoT has become synonymous with the Web of Things (WoT), which emphasizes the reuse of proven standards from the World Wide Web that lower barriers of entry, ensure scalability, and have been scaled down so that, today, virtually all things can participate in the WoT.

As things can now readily be interconnected, the next challenge researchers and practitioners face is how to make sense of all the connected resources to create intelligent systems.[3] Indeed, supporting users in combining services provided by smart devices remains a core challenge for ubiquitous computing research.[4] Likewise, in industrial settings, the easy reconfiguration of manufacturing environments is gaining importance, but it depends on production resources being aware of one another's capabilities on a semantic level.[5]

Semantic technologies can add meaning to machine-to-machine communication by establishing ontologies of interlinked terms, concepts, relationships, and entities. This could incur a paradigm shift across a broad range of industries: for instance, a model of the steps required when manufacturing a product combined with machine-readable descriptions of production machines would enable much more agile manufacturing in a mass-customized world. Thinking beyond vertical domain silos,[6] such manufacturing equipment could also be semantically integrated with the smart grid, which would unlock energy-saving potential by autonomously shutting down idling machines. Scenarios such as these require the tedious task of transferring knowledge from human experts and representing it in a form that can be readily used by autonomous systems.[7] This is worth the investment—once domain-specific ontologies have been created and properly integrated, they will become a semantic backbone that lets machines collaborate with each other and with humans as never before.

In this article, we introduce the Open Semantic Framework (OSF) as an enabler for this paradigm shift. The OSF supports collection, curation, and access to ontologies that encapsulate knowledge and experience in a machine-understandable way. It thereby forms the basis for enabling automated reasoning and decision making on top of knowledge models and lets semantic applications use domain-specific and general knowledge models. The OSF furthermore tackles several major obstacles to the widespread use of integrated semantic models by supporting individuals who are not versed with ontologies in understanding and extending them, and by making these models more tangible with the help of advanced human–interface technologies. After a general introduction to the OSF, we demonstrate its capabilities within the context of a "safety-by-design" system for industrial manufacturing, in which the OSF helps ensure that automatically generated production plans comply with work-safety regulations.

1541-1672/17/$33.00 © 2017 IEEE
Published by the IEEE Computer Society

The Open Semantic Framework

To foster the integration of semantic models and the usage of semantic technologies, we require a common and integrated engineering solution that lets us acquire, augment, maintain, access, interact with, and reason over machine-understandable knowledge, and deploy scalable semantic applications to diverse environments. These are the central tasks of our OSF.

Although semantic-software developers depend on tools for engineering and running their applications (such as Protégé and TopBraid Composer), they lack a common solution that integrates semantics tools and is suitable for supporting generic application development, deployment, and operation.[3] Compounding this integration problem, no solution yet exists for the acquisition of knowledge held by subject matter experts, nor have any ontology visualization, search, and discovery tools ever achieved significant traction. Therefore, an integrated framework for semantic application development should not only support users during the initial engineering phase and provide moderated access to knowledge models by (authorized) clients, but it should also provide visualization and manipulation tools that help non-ontologists discover the content and relationships between models, easily navigate the model space, and augment models—thereby allowing for continued "in-field" maintenance and improvement of semantic applications.

Our proposed solution, the OSF, contains an extensible set of core ontologies that capture concepts that cut across domains, such as information about units and dimensions.[4,8] These core ontologies are integrated with domain-specific knowledge packs (KPs) that enable specific applications to access their required information. In this article, we discuss several aspects of the OSF, including how knowledge is represented, managed, and acquired; how the OSF moderates knowledge access via a controlled set of queries; and the visualization support it provides to support users in navigating and learning about stored knowledge.

Knowledge Management and Engineering in the OSF

While our OSF's core ontologies are available to all of its client applications, the OSF is extended with KPs that encode domain-specific information for usage by specific clients (see Figure 1). KPs thus enable vertical interoperability between agents within a domain (for instance, an electric car and a charging station), whereas their integration with the core ontologies ensures horizontal interoperability across domains (for instance, a charging station with a manufacturing robot). In our OSF, KPs are kept distinct from core ontologies so that they can be loaded independently for usage by different client applications.

> Once a model has been acquired, it must be validated both syntactically and semantically.

One time-consuming and risk-prone aspect of semantic modeling is the inclusion of subject matter expertise into a KP and the extension of existing KPs into new areas of expertise through model mapping. Although it is perhaps impossible to fully automate this process, our approach to mitigating this problem is to base KP concepts on agreed-upon industrial standards. In these cases, it is sufficient to translate standards documents into a machine-understandable language rather than inventing new concepts from a clean slate.[4]

Once a model has been acquired, it must be validated both syntactically and semantically. For syntactic validation, we propose a mechanism akin to unit testing that is tightly integrated with the OSF: KPs supply testing queries that are executed (for all KPs) whenever one of the KPs or one of the OSF's core ontologies is updated. Semantic validation, on the other hand, is usually performed manually with the assistance of subject matter experts. However, because these experts usually are not versed in the usage of semantic modeling tools, our OSF needs to render the modeled information in a way that they can digest more easily.

Knowledge Access in the OSF

The OSF provides access to stored knowledge both in the core ontologies and in KPs through a controlled querying interface inside a REST API (marker 1 in Figure 2). This interface is based on prefabricated SPARQL query templates inside a KP (marker 2); thus, KPs not only determine what knowledge applications can access, but also exactly how they access it. The purpose of this mechanism is to prevent unwanted modifications to the knowledge models and to forbid clients from extracting all knowledge from the OSF; both aspects are of paramount importance for the commercial viability of any semantic framework.

Based on the OSF's REST API, accessing knowledge in the core and in KPs is straightforward: whenever the

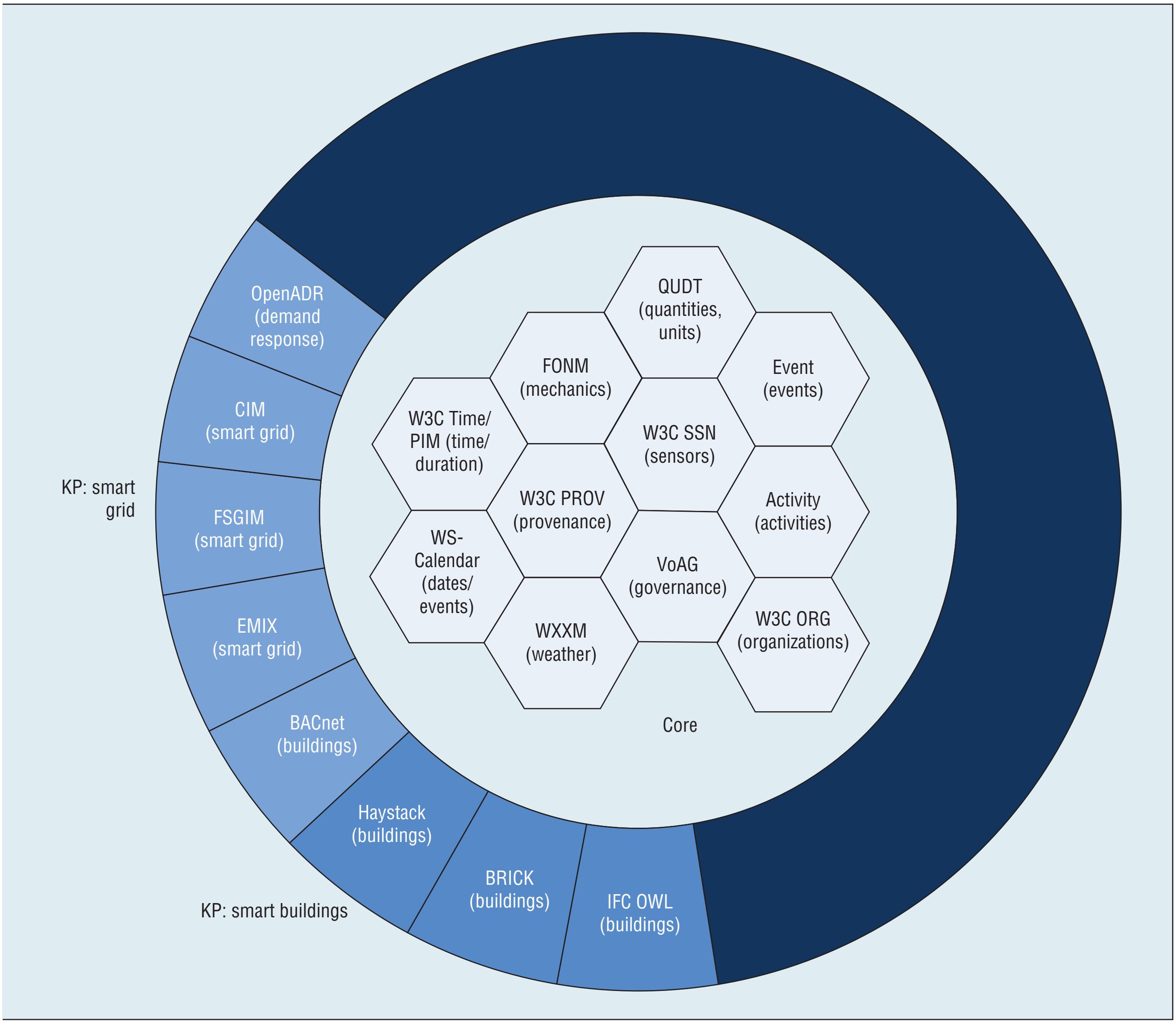

Figure 1. In the Open Semantic Framework (OSF), knowledge is structured in domain-specific knowledge packs (KPs) that depend on several core ontologies. Core ontologies contain information about general concepts, such as quantities, units, events, and device types and capabilities.

OSF loads a KP, its query templates are represented as resources that can be accessed by client applications (marker 3). For instance, a query template with the name `InstancesOfClass` that returns all instances of a given class—that is, the query returns all ?x that satisfy `?x rdf:type :className` for a given `:className`—can be accessed by sending an HTTP GET request to `https://example.org/queries/{knowledgePackName}/InstancesOfClass` that includes the class name as a parameter. The OSF responds to queries in the `SPARQL Query Results` JSON Format (www.w3.org/TR/sparql11-results-json).

The OSF's Semantic API can also be used to insert new information using construct or update query templates. This is particularly useful for integrating semantic models with real-time datastreams (marker 3): by inserting a datastream into its knowledge body, the OSF supports the automatic classification of datapoints and their incorporation into the model to support reasoning over them. For instance, the OSF could integrate an electric substation's measurement readings and relate one stream to line voltage, while another stream is automatically classified as containing electric current data. Thus, the OSF is also inclusive to legacy systems that can function as they did before while their data is connected to semantic models inside the OSF (marker 4). This makes

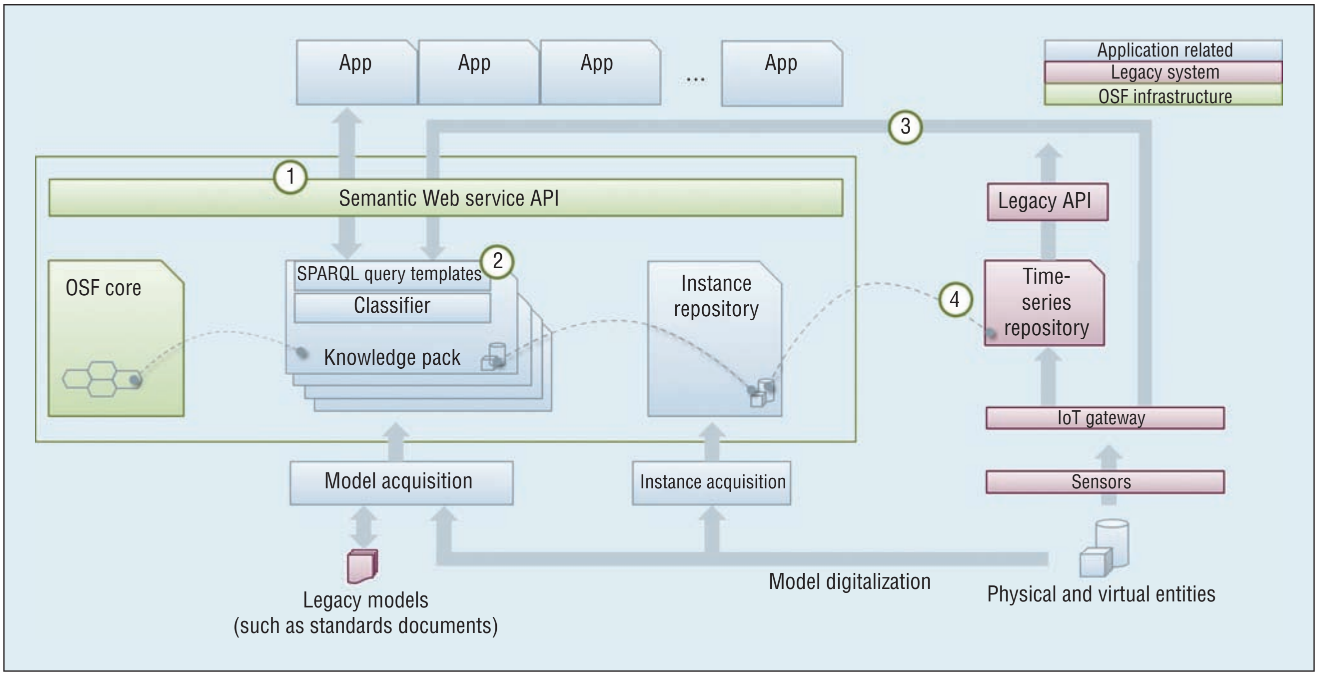

Figure 2. Overview of the OSF architecture and its interface to semantic applications. Relevant characteristics of physical and virtual entities are captured in knowledge models ("Model Digitalization"), while data from sensors that monitor them enters the OSF through a semantic API. Inside the OSF, data is stored and classified, thereby semantically integrating it with OSF core ontologies. Apps act on the acquired models and data through the semantic API as well.

legacy systems available for other applications to use.

Knowledge Visualization in the OSF

Visualizing knowledge in an easily accessible and tangible way is beneficial for several purposes, including the semantic validation of ontologies by subject matter experts and for supporting non-ontologists with the extension of vocabularies. The OSF supports knowledge visualization applications via a specialized, visualization-specific KP with queries that enable client applications to explore knowledge models. For instance, given a specific semantic node, these queries deliver that node's properties, information about its type, and links to adjacent nodes. (Note that these queries are powerful, because they can be used to easily extract all knowledge from our system that would undermine economic incentives to contribute to it. Access to generic exploration queries is thus strictly controlled.) Using the visualization KP, visualizers take the form of OSF client applications that explore loaded ontologies using these queries.

Existing approaches to visualize semantic models often merely display the underlying schemas, and users can easily get lost because the complexity of conceptual relationships makes it difficult to visualize ontologies on 2D computer screens. We therefore propose to use interfaces that support 3D interaction to produce practically usable visualizations of knowledge models. For visualizing ontologies in the 3D space, we propose a two-level approach (see Figure 3a): concrete instances of a loaded KP are displayed in a level above their associated classes in the underlying schema. Both instances and classes can be explored using an appropriate control mechanism (for example, clicking gestures or voice commands).

We implemented a virtual reality ontology visualization and exploration application that runs on a smartphone/headset combination and displays information from loaded KPs (see Figure 3b). We use a Leap Motion controller to navigate the virtual reality space and select nodes.

The OSF at Work: Enabling Safety by Design in Industrial Workplaces

In this section, we discuss a concrete application that uses the OSF's knowledge management and access features to increase worker safety in industrial settings. Any job performed in such workspaces could have potential hazards that can affect the involved human workers and the workspace as a whole.

Workplace safety regulation is enforced by regulatory bodies such as the US Occupational Safety and Health Administration (OSHA). OSHA penalties can result in fines

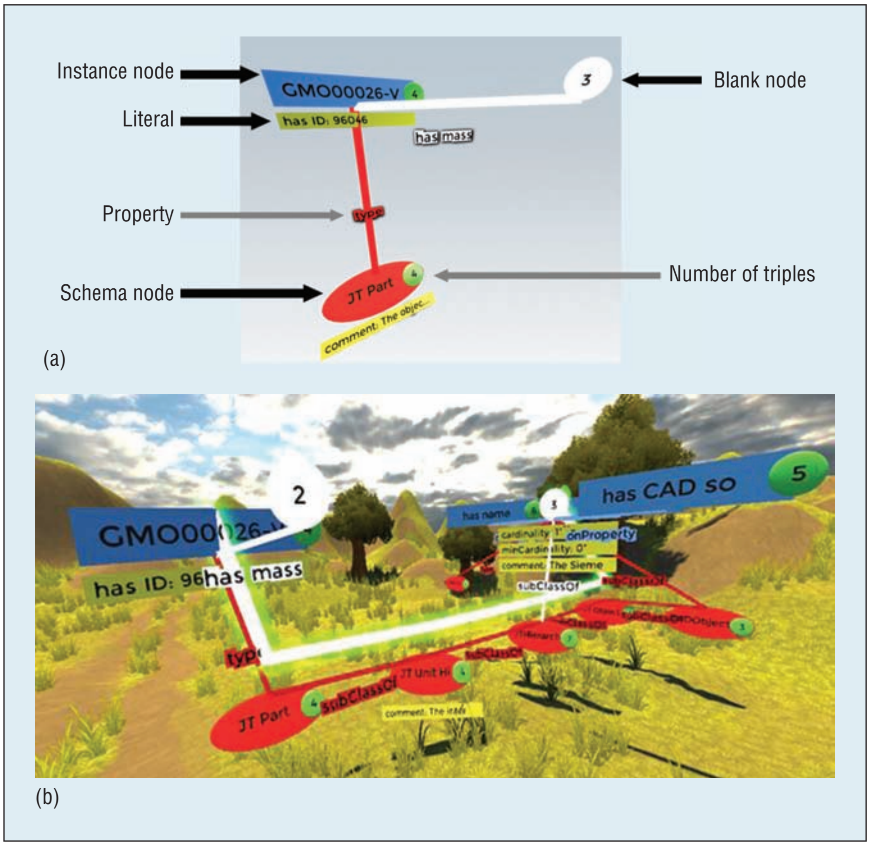

Figure 3. 3D knowledge visualization interface. (a) Visualization of an instance (blue) and associated class (red) node. Literals are rendered next to instances and classes, and both node types indicate the number of outgoing connections. (b) Virtual reality visualization of a KP; the user's exploration history is highlighted.

of up to $70,000 for each violation, blacklisting, and prison time. Currently, employers ensure workplace safety regulations are met through offline analyses of tasks performed by human workers. However, many businesses struggle with the implementation of safety and health rules because regulations are perceived as overly complex and enforcement and documentation require high investment.

Our "safety-by-design" approach automatically enforces workplace safety law according to knowledge about OSHA regulations that is stored in a KP within the OSF and documents employee compliance with these regulations inside a knowledge model that can be used by downstream applications. We apply this approach in the context of dynamically creating and executing a collaborative (human–robot) production plan for customized furniture: using a factory manager's input about the final product, our system creates a collaborative manufacturing plan that accounts for workplace safety regulation via the OSF, allowing the planner to consider potential hazards and integrate mitigation strategies in the assembly plans.

The basis for this functionality is formed by a KP with several knowledge models that relate to workplace safety. The Agent Model represents agents in an industrial context, including their capabilities (for example, that a worker can use power drills), preferences (such as handedness), and ergonomic properties (such as being able-bodied). The Workplace Safety Model formally defines regulatory constraints in terms of operational hazards, their associated risks, and possible mitigation strategies suggested by OSHA. Hazards (such as back bending or noise) impose risks on agents (such as hearing loss) and are linked to mitigation strategies (for example, "avoid heavy lifting") that in turn are associated with safety accessories in the Tools and Safety Accessories Model. The system uses this model to identify corresponding safety accessories that can be used to avoid hazards associated with using a specific tool (for example, antivibration gloves should be used while drilling). Finally, the Body Model that is part of the OSF's core ontologies describes the anatomical structure of bodies of human workers (that is, body parts and associated joints). It also captures positional information about individual workers, including the 3D positions of their body parts and joint angles. This information is provided to the KP in real time using a Microsoft Kinect body tracker and is inserted into the OSF via its REST API.

During planning, our system mitigates workplace safety hazards by inserting safety-relevant actions directly in the workflow (for example, "wear antivibration gloves"). Where applicable, the system also attempts to avoid hazards altogether: for instance, it avoids heavy lifting actions by workers if a robot is available to take over the lifting action. Upon successful planning, the assembly sequence is displayed, and the operator can decide to execute it. Finally, the system's dashboard displays an overview of all successfully mitigated health and safety hazards, and this information is kept in the system for documentation purposes. For instance, for the hazards mentioned earlier, the system will produce a notification that it potentially helped avoid two health problems: hand-arm vibration syndrome (ICD-10 I73.9) and carpal tunnel syndrome (ICD-10 G56).

This use case demonstrates that our OSF can provide straightforward access to knowledge models that codify complex constraints for workplace safety while integrating real-time data acquisition. The same principle can be applied to more general constraints in a broad range of domains, including spatial constraints (for logistics applications) and timing constraints (to control industrial mixing in the chemical industry), and we can even codify social norms—for instance, to support fair workload sharing in teams.

Businesses in a broad range of domains have shown interest in semantic applications, but they require an easy-to-use system that is accessible to non-semantics experts while maintaining the associated models' integrity. The integration of knowledge engineering and management support with a straightforward access mechanism and visualization capabilities within our OSF yields a one-stop shop for creating and deploying semantic applications and their lifecycle management. This approach is intended to make semantics as transparent to nonexperts as possible, thus improving the ability to acquire subject matter expertise in usable systems.

The OSF ecosystem also accounts for economic incentives in the knowledge management domain and lowers the threshold of building an ecosystem for intelligent applications by enabling capitalization and protecting investment of knowledge formalization via moderated access. It enables knowledge providers to encapsulate domain know-how into KPs and only exposes it through a moderated query API. Client apps are relieved from the need to have locally defined ad hoc information models, which enables a separation of concerns between knowledge formalization and application development, thereby facilitating development and maintenance of semantic applications.

Our OSF makes knowledge available as a web resource that is easy to reuse and effective to share globally and forms the basis of an open ecosystem that lets domain experts contribute knowledge in the form of KPs while running integration tests to ensure conformance of newly added knowledge. Bringing with it the potential to solve the knowledge integration problem on an institution-wide scale, the OSF represents an important step on the way to integrating knowledge models worldwide, enabling semantic applications globally to "stand on the shoulders of giants." ■

References

1. C. Floerkemeier et al., eds., *First Int'l Conf. Internet of Things*, 2008.
2. H. Sundmaeker et al., eds., *Vision and Challenges for Realising the Internet of Things*, 2010.
3. A. Gyrard et al., "Building the Web of Knowledge with Smart IoT Applications," *IEEE Intelligent Systems*, vol. 31, no. 5, 2016, pp. 83–88.
4. V. Issarny et al., "Service-Oriented Middleware for the Future Internet: State of the Art and Research Directions," *J. Internet Services and Applications*, vol. 2, no. 1, 2011, pp. 23–45.
5. S. Mayer et al., "UberManufacturing: A Goal-Driven Collaborative Industrial Manufacturing Marketplace," *Proc. 6th Int'l Conf. Internet of Things*, 2016, pp. 111–119.
6. F. Michahelles and S. Mayer, "Toward a Web of Systems," *XRDS*, vol. 22, no. 2, 2015, pp. 62–67.
7. E. von Hippel, "'Sticky Information' and the Locus of Problem Solving: Implications for Innovation," *Management Science*, vol. 40, no. 4, 1994, pp. 429–439.
8. S. Mayer, E. Wilde, and F. Michahelles, "A Connective Fabric for Bridging Internet of Things Silos," *Proc. 5th Int'l Conf. Internet of Things*, 2015; doi:10.1109/IOT.2015.7356559.

Simon Mayer is a senior research scientist in the Web of Systems research group at Siemens Corporate Technology. Contact him at simonmayer@siemens.com.

Jack Hodges is a senior research scientist in the Web of Systems research group at Siemens Corporate Technology. Contact him at jack.hodges@siemens.com.

Dan Yu is a portfolio project manager and innovation manager in the Web of Systems research group at Siemens Corporate Technology. Contact him at dan.yu@siemens.com.

Mareike Kritzler is a senior research scientist in the Web of Systems research group at Siemens Corporate Technology. Contact her at mareike.kritzler@siemens.com.

Florian Michahelles is the head of the Web of Systems research group at Siemens Corporate Technology. Contact him at florian.michahelles@siemens.com.