# A service computing manifesto: The next 10 years

**Article** *in* Communications of the ACM · March 2017

DOI: 10.1145/2983528

# A Service Computing Manifesto: The Next Ten Years

Athman Bouguettaya, Munindar Singh, Michael Huhns, Quan Z. Sheng, Hai Dong, Qi Yu, Azadeh Ghari Neiat, Sajib Mistry, Boualem Benatallah, Brahim Medjahed, Mourad Ouzzani, Fabio Casati, Xumin Liu, Hongbing Wang, Dimitrios Georgakopoulos, Liang Chen, Surya Nepal, Zaki Malik, Abdelkarim Erradi, Yan Wang, Brian Blake, Schahram Dustdar, Frank Leymann, Michael Papazoglou

## ABSTRACT

This manifesto[1] describes the challenges for the widespread adoption of service computing. By taking stock, the manifesto maps out a strategy that leverages emerging concepts and technologies to deliver on the full potential of the service paradigm. We explore the major obstacles that hinder the development and potential realization of service computing in the real world, propose research directions, and draw a ten-year roadmap to guide a redefinition of service computing to address emerging IT challenges. We recommend focusing on four main research directions to advance service computing: service design, service composition, crowdsourcing based reputation, and the Internet of Things.

## 1. BACKGROUND

Current IT advances can be likened in their effect and impact to the deep economic and societal transformations that were brought about by the industrial revolution. Recent years have seen the expansion of services in all sectors of the economy, building on the expansion of the Internet.

We define *service computing* (alternatively termed *service-oriented computing*) as the discipline that seeks to develop computational abstractions, architectures, techniques, and tools to support services broadly. A service orientation seeks to transform physical, hardware and software assets into a paradigm in which users and assets establish on-demand interactions, binding resources and operations, providing an abstraction layer that shifts the focus from infrastructure and operations to services.

But, as we argue below, service computing has not fully reached its potential. Technological advances provide an increasing opportunity for service computing. To avoid the mistakes of the past, we propose a manifesto - a community declaration of objectives and approaches - as a way to establish common ground among researchers in the field and to guide its future development [4].

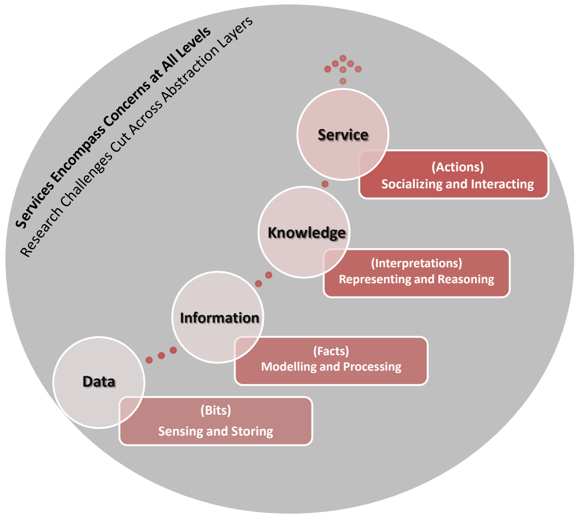


**Figure 1: Abstractions along the Computing Value Chain**

This manifesto first takes stock of the current state of service computing and then maps out a strategy for leveraging emerging concepts and technologies to deliver on the full potential of the service paradigm. It (1) *identifies the major obstacles that hinder the development and potential realization of service computing in the real world*, (2) *proposes research directions*, and (3) *draws a roadmap to enable the service computing field to redefine itself and become one of the powerful engines for social and economic activities.*

**Evolution of IT: Services as the next layer of the computing value chain**

Computing has progressed dramatically over the past few decades in delivering automated solutions for an increasing number of areas. In what follows, we provide a historical perspective on the evolution of computing. In the early days of computing, the challenge was to represent information in a machine-readable format that consisted of bits and bytes, called *data*. Over time, there was keen interest in complementing data with *meaning*, thus transforming it to *information*. With further advances in computing came the idea of adding *reasoning* to information, thus giving rise to *knowledge* [6]. The need for higher levels of abstraction has recently led to the notion of adding *action* to knowledge, resulting in *services*. Therefore, the abstract definition of a

Corresponding Author: Athman Bouguettaya, Email: athman.bouguettaya@sydney.edu.au

1. This manifesto is the culmination of a workshop that was organized at RMIT University (Melbourne, Australia) on Dec 8-9, 2014. A number of experts in service computing from around the world attended. The document was also circulated among other key leaders for further input, giving rise to this manifesto.

**Table 1: Representative Service Computing Venues**

| *Venue* | *Main Topics in the Last Three Years (in alphabetical order)* |
|---|---|
| *IEEE TSC* | Business Process Management for Data Services<br>Cloud Service Management<br>Cloud Based Social Computing<br>QoS-Aware Service Composition/Selection/Recommendation for Web/Cloud Services |
| *IJWSR* | Service Composition/Selection/Recommendation |
| *ICSOC* | Business Processes for Services<br>Cloud Management<br>QoS Management<br>Service Composition |
| *ICWS* | QoS-Aware Service Selection/Composition/Recommendation<br>QoS Management<br>Services for Social Networks<br>Service Privacy<br>Service Trust<br>Service Workflows |
| *SCC* | Business Processes for Services<br>Cloud Services<br>QoS Management<br>Service Selection/Composition/Recommendation |

service comes to fulfil the need to *act* and *deliver* on *knowledge*, i.e., to provide a way for knowledge to be *useful*. Accordingly, services are *defacto* presently considered as the highest level in the computing value chain[2] (see Fig. 1).

Services are ubiquitous in today's social and economic environment. Examples of such services are healthcare, financial management, human resources, and tourism planning. What distinguishes services from other computing paradigms is their ability to work in a competitive environment where the key parameter to distinguish between similar services is their *quality*. Knowledge in itself is not sufficient, but needs to be acted upon to bring about benefits. In the case of services, it is the ability to use quality of service (QoS) as a key discriminant to choose between services that provide the "action" on knowledge about services [7, 19, 24].

As economies undergo significant structural changes, digital strategies and innovation must provide industries with tools to create a competitive edge and build more value into their services [1]. Advances in online service technologies are transforming the Internet into a global workplace, a social forum, a means to manage one's individual affairs and promote collaboration, and a business platform for service delivery. Additionally, organizations are competitively compelled to provide service interfaces to their online services, allowing third-party developers to write auxiliary or satellite "apps" that add new uses to the original service, enrich its features and accessibility, and enhance its agility.

*Web service* technologies have been developed somewhat independently from the notion of service. They have been at the center of intense research in the past 15 years. In the enterprise market, sales of ready-made software or hardware products are rapidly being replaced by the provisioning of customized IT Web services, aimed at individual problem solving. *Service Computing* takes a broader view and has emerged as a cross-disciplinary research field that studies the science and technology underlying the popularity of the IT service industry. The ultimate goal of service computing is to bridge the gap between IT and business services to enable IT services to run business services more effectively and efficiently. Web services have so far been the key technology for delivering on service computing [1, 16, 15].

Service computing aims to support the creation and delivery of services, which involve computing devices and software components distributed on the Web and provisioned (and often also controlled) by diverse organizations. Automated composition seeks to mash up these resources to provide customized IT services based on users' goals and preferences [24]. To achieve this goal, standardization bodies, such as the World Wide Web Consortium (W3C) and the Organization for the Advancement of Structured Information Standards (OASIS), have led specification and standardization efforts for implementing service systems. In academia, service computing has attracted intensive attention. A number of journals and conferences were founded to assist research development in this area. Table 1 lists some major venues and their predominant research topics[3]. A state-of-the-art article in service computing can also be found in [24].

Nevertheless, the state of the art in service computing remains largely in the realm of research and its potential for wide deployment has largely been unfulfilled. One key impediment has been the confusion created by the myriad of often competing Web service standards, attempting to standardize *every single aspect* of the service lifecycle. Moreover, existing Web service standards and technologies do not provide adequate support for the computing needs of key emerging areas, which include Mobile Computing, Cloud Computing, Big Data, and Social Computing. These are the four key technologies of the Third Platform named by IDC that are currently influencing the landscape of global business[4]. The two impediments (i.e., confusion around Web service stan-

2. Framing the service paradigm as the top level in the computing value chain does not in any way lessen the importance of issues pertaining to each level in the chain.

3. These topics are selected based on the criterion that there were at least five related articles for each topic from 2013 to the venue date. This survey covers all articles for journals and only regular research papers for conferences.

4. http://www.idc.com/prodserv/3rd-platform/

dards and lack of support for emerging computing grand challenges) have hampered wider and quicker adoption of service computing.

We call for revisiting the practice of superimposing the service paradigm with the technology of delivery (i.e., Web services). Although Web services may remain relevant to developing various applications, we call for a focus on their underlying service needs. In particular, we call for the development of a new service paradigm that encompasses the four technologies identified by IDC, along with the introduction of new technologies, rather than emphasizing the expansion of existing Web service standards and technologies.

An aim of service computing is to harness the power and simplicity of the service paradigm with its *functional* and *non-functional* components to build modular software applications and provide a *higher* level of abstraction for selecting and composing services, thus elevating them to *first class* object status. A Service Oriented Architecture (SOA) is a technology-independent framework for defining, registering and invoking services [15]. Service computing, however, is broader than SOA and includes the upper level of business process modelling, management, and analysis to the lower level of service data management and analytics. SOA has become a core concept of service computing and provides the fundamental technologies for realizing service computing. The emergence of cloud computing, as a new service delivery model, has inspired researchers and practitioners to explore the use of service computing in the cloud. A large body of research has been devoted to how service-computing concepts are leveraged to facilitate the development of cloud applications. As already mentioned, service computing can benefit from and contribute to new directions inspired by the emergence of Mobile Computing, Cloud Computing, Big Data, and Social Computing, as exemplified in Fig. 2. This manifesto outlines the directions.

Since the emergence of the concept of service computing, a number of position and survey articles have been published [7, 9, 16, 24]. The present effort differs from previous attempts in two main ways. First, the driving factor in this work is to decouple service computing from the technologies to implement service-oriented systems that take full advantage of the promises and expectations of service computing. Second, this work emphasizes the contribution and influence of service computing on emerging trends in computing.

The structure of this manifesto is as follows. In Section 2, we first analyze the obstacles to service-computing implementations. We then describe the major challenges in emerging areas of service computing. In Section 3, we present a research roadmap to address the identified challenges. We draw conclusions and provide our assessments of the prospects for success in Section 4.

## 2. CHALLENGES IN SERVICE COMPUTING RESEARCH

According to our survey on the articles published in the aforementioned journals and conferences (see Table 1), current service computing research focuses mostly on seven problem areas: *architecture*, *specification languages*, *protocols*, *frameworks*, *lifecycle*, *quality of service*, and *the establishment of trust and reputation across the boundaries of autonomous enterprises*. The problems are especially apparent in the aforementioned emerging service domains.

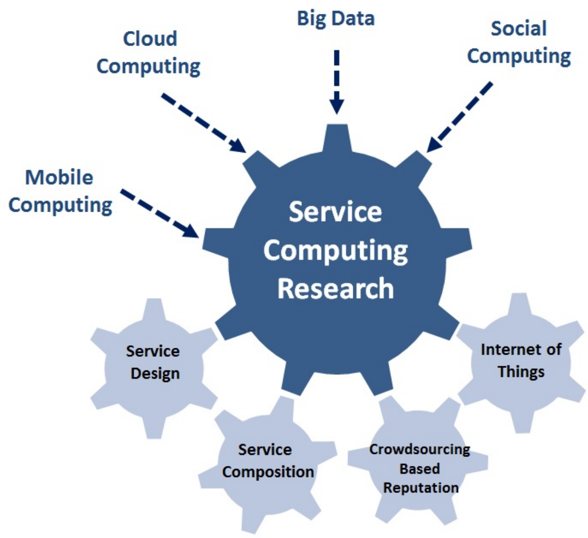


**Figure 2: Service Computing Roadmap: Influencing Factors and Research Directions**

An often overlooked strategic challenge in service computing is *analyzing why service computing has not reached its full potential in the real world, and what needs to be done to change that.* A barrier that hinders service computing research from being transformed into an effective solution for industry challenges has been the concomitant lack of an easy methodology to transform complex data processing problems into routine services and an easy methodology to solve complex service interactions via data-centric architectures, such as Hadoop. A significant challenge is *enabling the seamless cooperation of multiple organizations working on different platforms to satisfy consumers' requirements.* This would deliver on the much needed service composition. Barriers to automated inter-organizational service composition include disparities in competency, trust, accountability, functional and non-functional goals (including security and privacy) at the organizational level and in interaction protocols and representations at the technical level.

We identify four emerging research challenges in service computing: Service Design, Service Composition, Crowdsourcing-Based Reputation, and the Internet of Things (IoT). First, we introduce service design, which is a fundamental but yet unsolved research problem in service computing. Second, we elaborate on challenges of service composition in the context of large-scale Web and cloud service systems, big data, and social networks. Next, crowdsourcing is a cost-effective means for IoT deployment, through collecting sensing data from pervasive sensing devices, e.g., mobile phones [14]. We focus on crowdsourcing as a key mechanism for reputation computation. Lastly, we discuss how service computing helps realize the IoT vision and challenges (see Fig. 2).

### 2.1 Challenges in Service Design

**Service design**. Service design is about mapping a formal understanding of the nature of services and relationships thereof. This is an important prerequisite to building sound service systems. Service systems have so far been built without an adequate rigorous foundation that would enable reasoning about them. The preferred approach has been to rely on traditional software engineering approaches, which do not typically take into account the fact that service systems inherently bring together autonomous parts. The Web potentially provides a unique and uniform platform to develop sound service systems. However, there has not been any comprehensive theoretical framework to define and analyze complex service systems on the Web. As a result, only a few complex service systems have been developed.

## 2.2 Challenges in Service Composition

**Large-scale Web and cloud service composition**. Partly fuelled by the recent widespread successes of big data, the composition of large numbers of services into a coherent system is an emerging research area. We assess the scale involved as follows. In 2008, a survey discovered 5,077 WSDL-described Web services available on the Internet [2]. In 2016, there are almost 15,000 Web services registered in a public website[5]. In addition, we believe that a large proportion of modern Web services are non-WSDL-described, such as those that are cloud-based. The current popularity of cloud computing has inspired the fast growth of cloud-based services and a 2013 survey discovered 6,686 cloud services [13]. In the era of smartphones, millions of apps are downloadable from cloud-based app stores. It is estimated that there are 1.6 million Android apps and 1.5 million Apple apps as of July 2015[6]. *Precisely and efficiently searching for services from these large-scale repositories* is becoming a critical challenge [3, 24].

In the setting of **big data** that might be accessed simultaneously by numerous services, *existing service selection, composition, and recommendation approaches that mostly assume a static data environment are inadequate*. Composition technologies that address consistency should be explored. In the setting of the IoT, *Smart City* can be viewed as a typical example of large-scale service composition, where millions of diverse and heterogeneous digital devices and services are integrated dynamically for provisioning multiple real-time functions or user-customized functions. *Selecting and composing services from such numerous and ever-changing devices and services to fulfill user requirements in a real-time and context-aware fashion* is a difficult mission, which requires urgent attention [22].

In **large social networks**, such as Facebook and Twitter, in which billions of users register and most users have hundreds of friends or followers on average (on average, 338 friends per user in Facebook[7] in 2010 and 208 followers per user in Twitter[8] in 2013), the resultant big data is complex as well as large. *Service composition based on social relations* poses fundamental serious challenges.

## 2.3 Challenges in Crowdsourcing-Based Reputation

Trust plays an important role in the functioning of a service ecosystem. It is, however, difficult to establish trust when services that offer similar functionalities compete. Reputation is an effective approach in social networks for predicting credibility based on past behavior gleaned from consumers. It is often difficult to ascertain reputation in an open and often anonymous environment. Hence, reputation and crowdsourcing are important approaches for deriving trust.

Computing the reputation of a service entity in a community is realized by collecting all individual users' opinions towards this entity in this community. Crowdsourcing provides an effective means for data collection through collaborations within the community [5]. Nevertheless, several research challenges remain in the computation of crowdsourcing-based reputation.

**Quality of crowdsourcing**. The major difference between crowdsourcing and traditional user feedback collection is that crowdsourcing is more likely to use financial rewards and other incentives to motivate participation. However, it is unclear *how these factors influence the quality of crowdsourcing*. In addition, service users' opinions are typically *ambiguous, context-dependent*, and based on *individual preference*. Ambiguity refers to what users express instead what they really think. Users' opinions may also be affected by time, location, and social factors (e.g., social relations between service users or between service users and service providers [12]. Some users may have particular preferences on certain service products [10]. There is a strong need to predict the outcome of crowdsourcing reputation given that the reputation is influenced by several dependent factors.

**Trustworthiness of crowdsourcing contributors**. Some service users' opinions might be *unfair* and even *malicious* towards particular service products. Unfairness or maliciousness is context-dependent, affected by the variation in services, time, and locations. A key issue is the selection of crowdsourcing contributors based on their trustworthiness, taking into account the context in which they operate.

**Testing platform**. Hitherto, there is no standardized testing platform to compare trust and reputation models. There is a strong demand for designing appropriate evaluation metrics to compare trust and reputation models for services.

## 2.4 Challenges in the IoT

The Internet of Things (IoT) is an emerging and promising area that proposes to turn every tangible entity into a node on the Internet. More specifically, the tangible entities ("things") are any Internet-connected sensor, camera, display, smart phone, or other smart communicating devices. The IoT poses two fundamental challenges [17]: (1) communication with things and (2) management of things. Service technologies can help address these challenges, through provisioning communication and interoperability means to the IoT, such as REST and service composition models. One challenge is that things are resource-constrained and traditional standards, such as SOAP and BPEL, are too heavy weight to be applicable in the IoT. Furthermore, *existing models of service composition cannot be directly used for IoT interoperation, because of architectural differences*. In contrast to the single-type Web service component model (i.e., services), the IoT component model is *heterogeneous* and *multi-layered* (e.g., devices, data, services, and organizations). Innovative models are required for IoT composition. Specifications of the *functionalities of the IoT* are a key challenge. Through the diverse, mobile, and context-aware devices, data and services can simultaneously gener-

5. http://www.programmableweb.com/
6. http://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/
7. http://www.pewresearch.org/fact-tank/2014/02/03/6-new-facts-about-facebook/
8. http://expandedramblings.com/index.php/march-2013-by-the-numbers-a-few-amazing-twitter-stats/

ate diverse and context-aware functions [11]. Hence, in the IoT, *desired functionality of components is more dynamic and context-aware than in traditional settings*, which brings significant complexity in the composition process.

In a nutshell, the IoT involves (1) incorporating possibly billions of things and (2) harnessing their data and functionality to provide novel smart services that benefit enterprises, industries, and our society. The fundamental IoT challenges as related to service computing are:

1. *Continuously maintaining cyber personalities and contexts for IoT devices.* In particular, IoT things need to have Web identities and Web representations (e.g., Web proxies) that reflect their physical spaces. They also need to connect and communicate within social, environmental, user-centric, and application contexts, and such contexts need to be maintained and managed.

2. *Continually discovering, integrating, and (re)using IoT things and their data.* Specifically, the IoT environment is a federated environment where things and their data, cloud services, and IT services (e.g., for data analysis and visualization) are often provided by independent providers with diverse interfaces, as well as business, cost, and QoS models. To provide new Internet-scale services, the IoT must (re)use IoT things deployed by others and data collected by others for their own purpose.

# 3. SERVICE COMPUTING RESEARCH ROADMAP

Following the major challenges identified in Section 2, we now propose a research roadmap for service computing. The roadmap focuses on four emerging research areas, namely *service design, service composition, crowdsourcing-based reputation*, and *Internet of Things*.

## 3.1 Service Design

Research should focus on laying the foundation for service design drawing from similar research in databases, software engineering, and distributed systems. The design of service systems should build upon a formal model of services that enables efficient access to a large service space with diverse functionalities. The service model can support accessing services much as the relational model supports accessing structured data. As service composition is usually needed to fulfill complex user requests, it is important for the service model to support dependency relationships among different services and their operations. Similar to the relational model, the service model should enable the design of service query algebra and calculus that allow general users to declaratively query multiple services in a transparent and efficient manner. In addition, service query optimization should go beyond just generating efficient query execution plans. As a large number of service providers may compete to offer similar functionalities, the design of the service model should support the implementation of optimization strategies for finding the "best" Web services or composition thereof with respect to the expected user-supplied quality. In sum, a formal service model that meets the above requirements will take center stage to provide efficient and transparent access to computing resources through services, which is a critical step to reach the full potential of service computing.

Recent work [25] has shown great promise in designing and using a formal service model to help users access service. A set of algebraic operators, defined based on a formal service model, provides standard ways to manipulate services as first-class objects. These include choosing specific operations from a service as well as quality based service selection and mash-up operators across multiple services to form a composed service. The implementation of these operators enables the generation of service execution plans that can be directly used by users to access services. Due to functional dependency and quality dimensions, a service model is more complex than a relational data model.

In particular, three key features of services are crucial: functionality, behavior, and quality. Functionality is specified by the operations offered by a service; behavior reflects how the service operations can be invoked, and is decided by the dependency constraints between service operations; quality determines the non-functional properties of a service. A viable service query language should allow users to locate and invoke their desired functionalities, select the best service providers that meet their quality requirements, and generate service compositions when functionalities from multiple services are required.

An important approach to address the composition of services is to do so not from the perspective of one party, but from a representation of interactions between two or more autonomous parties. Protocols are specifications of interactions that assiduously disregard the implementations of the services, but describe the interactions: protocols thus promote interoperability among heterogeneous and autonomous services [23]. Newer approaches for protocols accommodate data representations, but additional research is needed to bridge the gap between interaction and data. An important direction is to formalize sociotechnical interactions as a basis for governance of autonomous services [20].

## 3.2 Service Composition

Future research in service composition should target the following topics:

**Large-scale Web and cloud service composition**. Service composition research should extend to *non-WSDL-described services or services described by plain text.* As an example, ProgrammableWeb.com hosts over 10,000 API services and over 6,000 service mashups, with a great majority described in plain text. Web information extraction, natural language processing, data and text mining, collaborative tagging, and information retrieval technologies can be used to extract useful semantics, group relevant services, and detect novel composition patterns. Nevertheless, short service descriptions comprised of limited terms coupled with the diverse naming conventions used by service providers pose novel challenges that demand technological innovations to advance the state of the art in both service computing and all relevant domains. Furthermore, a software system may need to evolve during its lifecycle and be able to handle the changes in its running environment and the increasing complexity of its workflows. Self-adaptive software evaluates its behavior and makes adjustment according to the evaluation result to address issues and improve performance.

A cloud computing environment provides an attractive option for deploying services, because of the potential scalability and accessibility it offers. However, it introduces problems related to:

1. Maintenance – The resources are not under the explicit control of the service provider.

2. Security – The cloud might not be within the enterprise boundaries of the service provider.

3. Service-Level Agreements (SLAs) – Resource allocation is the responsibility of the cloud provider. For example, a service might be unavailable not only due to updates by the service provider, but also due to updates by the cloud provider.

These problems must be addressed as services are moved to clouds and, recently, containers within clouds.

**Big data driven service composition**. Due to limitations of existing methods, *new service selection, composition, and recommendation technologies are key approaches to leveraging state-of-the-art big data research outcomes.* One important topic in current big data research is the development of algorithms and models for processing data online. This is fundamentally different from the traditional batch processing approach. Online service composition may provide a promising direction to achieve scalable and adaptive composition solutions to deal with large-scale, highly dynamic, and diverse big data services. Another important consideration is how the outcomes of service computing and big data frameworks are interpreted by humans. As such, it will become increasingly important to relate big data analytic frameworks with the nature of the humans for whom they serve [21].

**Social network based service composition**. Service selection, recommendation, and composition in large-scale social networks should target *combining social network and complex network analysis methods, as well as trust computing techniques.* One promising direction is to incorporate social network data that records the interaction of service users with the service data to detect hidden relationship between services and generate potential service compositions. In particular, user activities exhibited through social media services, such as posting experiences, questions and feedbacks about services, could bring novel insight to better understand and leverage both traditional and emerging services. For example, the records of using services to build service mashups posted on public programming forums can be leveraged to recommend interesting services to other users and build new service mashups. As another example, user behavior patterns can be detected from event logs recorded by social media or e-business platforms and used to discover latent knowledge for business process mining. Furthermore, emerging services may bring novel QoS features that go beyond the traditional ones, such as reliability, availability, and response time. Domain-specific quality features that reflect users' interests in choosing and composing services could be extracted through social media services that capture user personal judgement [10].

## 3.3 Crowdsourcing-Based Reputation

Future research on crowdsourcing-based reputation of services should target the following directions:

**Quality of crowdsourcing**. Future studies should focus on how monetary or other interesting factors affect the quality of crowdsourcing data for choosing services. *Social studies should be carried out to survey the impact of these interest factors on the reliability of crowdsourcing and the scope of the crowdsourcing contributors.* The three factors (*ambiguity, context-dependency,* and *individual preference*) that influence the quality of crowdsourcing data should be studied. For the ambiguity issue, future research should focus on *how trust assessment questions and scales as well as human computer interactions should be designed to precisely capture users' trust-related perceptions*, through the collaboration with cognitive science and human-computer interaction research. The difference between context-dependency and individual preference is that the former relies more on the objective factors, e.g., time, location, and social relations, and the latter relies more on the subjective factors, e.g., personal experience. These two kinds of factors, e.g., social relations and personal preference, may influence each other simultaneously. Future research should target *how to model the interrelation between the two groups of factors and how they can be integrated to predict their impact on the quality of crowdsourcing data.*

**Trustworthiness of crowdsourcing contributors**. Future research should focus on *studying the context-dependent features of contributors' trustworthiness.* The credibility of a crowd member (i.e., a human user or another service) determines how much other service consumers may trust its reported ratings regarding the services it has invoked. This allows us to differentiate between service trust and feedback trust. For instance, a service that is not trustworthy as a service provider may be trustworthy as a judge of the behavior of other service providers, and vice versa. *Trade-off strategies for selecting users with different cost and trustworthy levels for crowdsourcing* should also be explored. Selecting the most appropriate crowdsourcing contributors or crowd workers will require services that interact and combine information from contributors, service providers, and third-party sources, such as job listings and social media [8].

## 3.4 Internet of Things

In traditional service computing, a service composition focuses on finding an effective combination of component services [19]. Recent work has suggested that a service composition in the IoT needs to find an effective combination of both component services and devices [11]. In emerging IoT architectures based on service composition, there is a need to find an effective combination of component services, data services supported by cloud platform services, and devices. Cloud component services are an integral part of the IoT, because they are needed to manage device Web representations, contexts, and related data processing services anywhere and without the need to rely on a computing center.

Complementary to the current work on service discovery and integration, an important and novel direction in IoT research lies in the area of device discovery and integration. Existing results based on a combination of Semantic Sensor Networks (SSN) and OpenIoT [18] provide a device layer architecture and related functionality for IoT device discovery and integration. SSN defines an ontology that is used to describe IoT things and to find devices from the attributes of the data they produce. Nevertheless, as IoT things are diverse in functionality and access methods, it is infeasible for a single uniform ontology to cover the highly heterogeneous space of things. Information retrieval and text mining techniques can be leveraged to improve the accuracy of device discovery.

Integration of IoT things can be greatly facilitated by discovering the correlations among things. However, correlations among IoT things are usually difficult to detect. Unlike human social networks, where people are well connected, things often have limited explicit interconnections. An interesting direction is multi-hop connections, which leverage human-to-things interactions to correlate IoT things. Graph-based approaches and machine-learning techniques can help discover hidden interesting relationships among IoT things, and thus suggest interesting and novel integration patterns.

# 4. CONCLUSION

Service computing has a bright future supporting the tremendous advances in emerging areas of computing such as Mobile Computing, Cloud Computing, Big Data, Social Computing, and beyond. We make the case in this Manifesto that the potential of service computing is far greater than what has been achieved so far. We lay down a path forward to take service computing to new heights of innovation. To forge ahead, we make the important statement that, for the service computing paradigm to succeed, it needs to be decoupled from the technology of the day. The challenges may be difficult but the payoffs are great and there is no reason why an ambitious research agenda would not produce enormous benefits for computer science and society.

# ACKNOWLEDGMENTS

This research was made possible by LP120200305, DP150100149, and DP160103595 grants from Australian Research Council and NPRP 7-481-1-088 and NPRP 9-224-1-049 grants from the Qatar National Research Fund (a member of The Qatar Foundation). The statements made herein are solely the responsibility of the authors.

# 5. REFERENCES

[1] Special issue: Services science. *Communication of the ACM*, 49(7), 2006.

[2] E. Al-Masri and Q. H. Mahmoud. Investigating web services on the world wide web. In *Proceedings of the 17th international conference on World Wide Web (WWW 2008)*, pages 795–804. ACM, 2008.

[3] B. M. Brian and N. M. F. Knowledge discovery in services (kds): Aggregating software services to discover enterprise mashups. *IEEE Transactions on Knowledge and Data Engineering*, 23(6):889–901, 2011.

[4] H. Chesbrough and J. Spohrer. A research manifesto for services science. *Communications of the ACM*, 49(7):35–40, 2006.

[5] A. Doan, R. Ramakrishnan, and A. Halevy. Crowdsourcing systems on the worldwide web. *Communication of the ACM*, 54(4):86–96, 2011.

[6] M. Fricke. The knowledge pyramid: A critique of the dikw hierarchy. *Journal of Information Science*, 35(2):131–142, 2009.

[7] M. N. Huhns and M. P. Singh. Service-oriented computing: Key concepts and principles. *IEEE Internet Computing*, 9(1):75–81, 2005.

[8] J. Jarrett, L. Ferreira da Silva, L. Mello, S. Andere, G. Cruz, and M. Blake. Self-generating a labor force for crowdsourcing - is worker confidence a predictor of quality? In *Proceedings of the Third IEEE Workshop on Hot Topics in Web Systems and Technologies*, pages 85–90. IEEE, 2015.

[9] A. L. Lemos, F. Daniel, and B. Benatallah. Web service composition: A survey of techniques and tools. *ACM Comuting Surveys*, 48(3):1–41, 2015.

[10] X. Liu, A. Kale, J. Wasani, C. Ding, and Q. Yu. Extracting, ranking, and evaluating quality features of web services through user review sentiment analysis. In *Proceedings of the 2015 IEEE International Conference on Web Services (ICWS 2015)*, pages 153–160. IEEE Computer Society, 2015.

[11] A. G. Neiat, A. Bouguettaya, T. Sellis, and H. Dong. Failure-proof spatio-temporal composition of sensor cloud services. In *Proceedings of 12th International Conference on Service-Oriented Computing (ICSOC 2014)*, pages 368–377. Springer, 2014.

[12] S. Nepal, C. Paris, and A. Bouguettaya. Trusting the social web: Issues and challenges. *World Wide Web Journal*, 18:1–7, 2015.

[13] T. Noor, Q. Z. Sheng, A. Ngu, and S. Dustdar. Analysis of web-scale cloud services. *IEEE Internet Computing*, 18(4):55–61, 2014.

[14] A. Pal. Internet of things: Making the hype a reality. *IT Professional*, 17(3):2–4, 2015.

[15] M. Papazoglou and W. Van den Heuvel. Service oriented architectures: approaches, technologies and research issues. *The VLDB Journal*, 16(3):389–415, 2007.

[16] M. P. Papazoglou and D. Georgakopoulos. Service-oriented computing. *Communications of the ACM*, 46(10):24–28, 2003.

[17] D. Raggett. The web of things: Challenges and opportunities. *IEEE Computer*, 49(5):26–32, 2015.

[18] M. Serrano, Q. Hoan Nguyen Mau, P. Danh Le, M. Hauswirth, J. Soldatos, N. Kefalakis, P. Jayaraman, and A. Zaslavsky. Defining the stack for service delivery models and interoperability in the internet of things: A practical case with openiot-vdk. *IEEE Journal on Selected Areas in Communications*, 33(4):676–689, 2015.

[19] Q. Z. Sheng, X. Qiao, A. V. Vasilakos, C. Szabo, S. Bourne, and X. Xu. Web services composition: A decade's overview. *Information Sciences*, 280:218–238, 2014.

[20] M. Singh. Norms as a basis for governing sociotechnical systems. *ACM Transactions on Intelligent Systems*, 5(1):1–23, 2013.

[21] W. Tan, M. Blake, I. Saleh, and S. Dustdar. Social-network-sourced big data analytics. *IEEE Internet Computing*, 17(5):62–69, 2013.

[22] X. Xu, Q. Z. Sheng, L.-J. Zhang, Y. Fan, and S. Dustdar. From big data to big service. *IEEE Computer*, 48(7):80–83, 2015.

[23] P. Yolum and M. Singh. Flexible protocol specification and execution. In *Procedings of the First International Joint Conference on Autonomous Agents and MultiAgent Systems*, pages 527–534, 2002.

[24] Q. Yu, X. Liu, A. Bouguettaya, and B. Medjahed. Deploying and managing web services: Issues,

solutions, and directions. *The VLDB Journal*, 17(3):537–572, 2008.

[25] Q. Yu and A. Bouguettaya. Framework for web service query algebra and optimization. *ACM Transactions on the Web*, 2(1):1–35, 2008.