COMPUTING edge

> Blockchain
> Cryptocurrency
> Digital Transformation
> Internet of Things

JANUARY 2020

www.computer.org

IEEE

IEEE COMPUTER SOCIETY

COMPUTING edge

IEEE COMPUTER SOCIETY computer.org • +1 714 821 8380

Printed with inks containing soy and/or vegetable oils

IEEE

IEEE COMPUTER SOCIETY

STAFF

Editor
Cathy Martin

Publications Operations Project Specialist
Christine Anthony

Production & Design
Carmen Flores-Garvey

Publications Portfolio Managers
Carrie Clark, Kimberly Sperka

Publisher
Robin Baldwin

Senior Advertising Coordinator
Debbie Sims

Circulation: ComputingEdge (ISSN 2469-7087) is published monthly by the IEEE Computer Society. IEEE Headquarters, Three Park Avenue, 17th Floor, New York, NY 10016-5997; IEEE Computer Society Publications Office, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720; voice +1 714 821 8380; fax +1 714 821 4010; IEEE Computer Society Headquarters, 2001 L Street NW, Suite 700, Washington, DC 20036.

Postmaster: Send address changes to ComputingEdge-IEEE Membership Processing Dept., 445 Hoes Lane, Piscataway, NJ 08855. Periodicals Postage Paid at New York, New York, and at additional mailing offices. Printed in USA.

Editorial: Unless otherwise stated, bylined articles, as well as product and service descriptions, reflect the author's or firm's opinion. Inclusion in ComputingEdge does not necessarily constitute endorsement by the IEEE or the Computer Society. All submissions are subject to editing for style, clarity, and space.

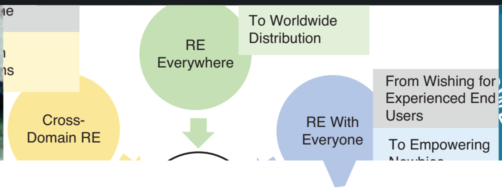
Unsubscribe: If you no longer wish to receive this ComputingEdge mailing, please email IEEE Computer Society Customer Service at help@computer.org and type "unsubscribe ComputingEdge" in your subject line.

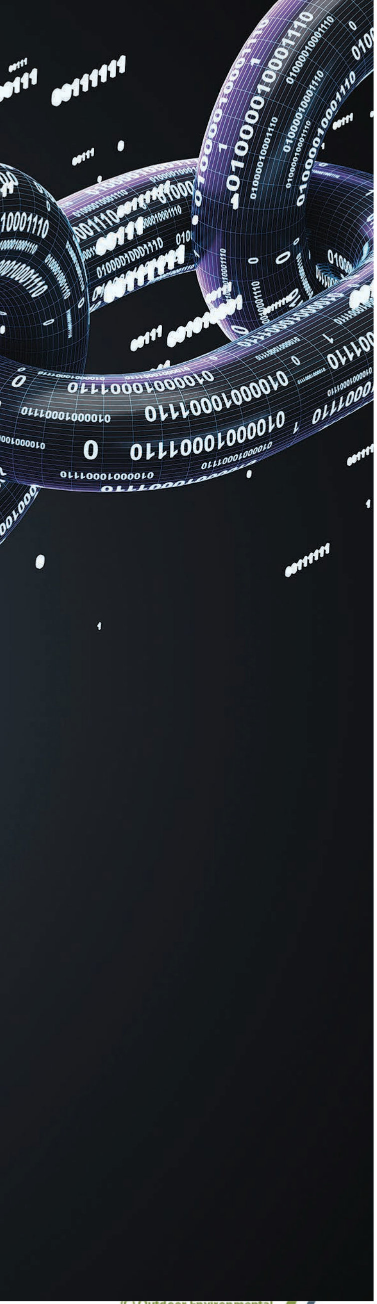
IEEE prohibits discrimination, harassment, and bullying. For more information, visit www.ieee.org/web/aboutus/whatis/policies/p9-26.html.

IEEE Computer Society Magazine Editors in Chief

JANUARY 2020 • VOLUME 6, NUMBER 1

COMPUTING edge





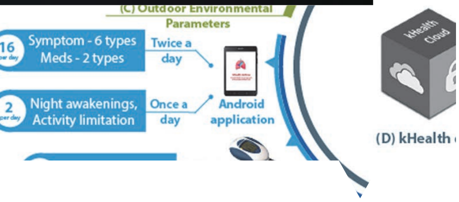
Blockchain

Cryptocurrency

Digital Transformation

Internet of Things

Departments



46

Extending Patient-Chatbot Experience with Internet of Things and Background Knowledge

Magazine Roundup

The IEEE Computer Society's lineup of 12 peer-reviewed technical magazines covers cutting-edge topics ranging from software design and computer graphics to Internet computing and security, from scientific applications and machine intelligence to visualization and microchip design. Here are highlights from recent issues.

Computer

Log My Dog: Perceived Impact of Dog Activity Tracking

The pet industry is catching up in the wearables market, and pet activity and location trackers are increasingly worn by our furry friends. The authors of this article from the September 2019 issue of *Computer* report on an empirical study investigating the user perceptions of a popular dog activity tracker. Results show that these trackers have a positive impact on owners' motivation to increase their physical activities with their dogs.

Computing in Science & Engineering

Scalable Reactive Molecular Dynamics Simulations for Computational Synthesis

Reactive molecular dynamics (MD) simulation is a powerful research tool for describing chemical reactions. The authors of this article from the September/October 2019 issue of *Computing in Science & Engineering* eliminate the speed-limiting charge iteration in MD with a novel extended-Lagrangian scheme. The extended-Lagrangian reactive MD (XRMD) code drastically improves energy conservation while substantially reducing time-to-solution. Furthermore, the authors introduce a new polarizable charge equilibration (PQEq) model to accurately predict atomic charges and polarization.

IEEE Annals of the History of Computing

Founding and Growing Adobe Systems, Inc.

Founded in 1982, Adobe Systems heralded several of the technological innovations necessary to precipitate the emergence of desktop publishing as well as many features of modern office computing, digital media, and graphic arts. In this article from the July–September 2019

2469-7087/20 © 2020 IEEE

issue of *IEEE Annals of the History of Computing*, Adobe founders Charles Geschke and John Warnock cover their professional history, the conception of Adobe Systems, and its growth. They also explain the technology behind the advances in computer printing, electronic file transfer, and digital art and photography. Adobe, its products, and its engineers played a key role in these developments, which enabled desktop publishing and the publishing revolution.

IEEE Computer Graphics and Applications

Uncertainty-Aware Visualization for Analyzing Heterogeneous Wildfire Detections

There is growing interest in using data science techniques to characterize and predict natural disasters and extreme weather events. Such techniques merge noisy data gathered in the real world, from sources such as satellite detections, with algorithms that strongly depend on the noise, resolution, and uncertainty in these data. In this article from the September/October 2019 issue of *IEEE Computer Graphics and Applications*, the authors present a visualization approach for interpolating multiresolution, uncertain satellite detections of wildfires into intuitive visual representations. They use extrinsic, intrinsic, coincident, and adjacent uncertainty representations as appropriate for understanding the information at each stage. To demonstrate their approach, the authors use their framework to tune two algorithms for characterizing satellite detections of wildfires.

IEEE Intelligent Systems

Using Social Media to Detect Socio-Economic Disaster Recovery

There has been growing interest in harnessing artificial intelligence (AI) to improve situational awareness for disaster management. As a first step toward investigating the possibility of developing an AI-based method for detecting socio-economic recovery, this article from the May/June 2019 issue of *IEEE Intelligent Systems* studies the correlations between public sentiment on social media and socio-economic recovery activities as reflected in market data. The result shows multiple correlations between sentiment on social media and the socio-economic recovery activities involved in restarting daily routines. Conventional socio-economic recovery indicators, such as governmental statistical data, have a significant time lag before publishing. Using public sentiment on social media instead can improve situational awareness in recovery operations.

IEEE Internet Computing

Energy-Efficient Analytics for Geographically Distributed Big Data

Big data analytics on geographically distributed datasets (across data centers or clusters) has been attracting increased interest in both academia and industry, posing significant complications for system and algorithm design. In this article from the May/June 2019 issue of *IEEE Internet Computing*, the authors present a dynamic global manager selection algorithm to minimize energy consumption cost by fully exploiting the system diversities in geography and variation over time. The algorithm makes real-time decisions based on measurable system parameters through stochastic optimization methods, while achieving performance balance between energy cost and latency. Extensive trace-driven simulations verify the effectiveness and efficiency of the proposed algorithm. The authors also highlight several potential research directions that remain open and require future elaborations in analyzing geo-distributed big data.

IEEE Micro

Accelerating Image-Sensor-Based Deep-Learning Applications

In this article from the September/October 2019 issue of *IEEE Micro*, the authors review two inference accelerators that exploit value properties in deep neural networks (DNNs): Diffy and Tactical. Diffy targets spatially correlated activations in computational imaging DNNs. Tactical targets sparse neural networks using a low-overhead hardware/software weight-skipping front-end. The authors combine these accelerators into Di-Tactical to boost benefits for both scene understanding workloads and computational imaging tasks.

IEEE MultiMedia

A 3D Scene Management Method Based on the Triangular Mesh for Large-Scale Web3D Scenes

Real-time rendering of large-scale Web3D scenes was difficult to implement in virtual-reality systems and geographic information systems (GIS) in the past because of the technical constraints in CPU, memory, and network bandwidth. In this article from the July–September 2019 issue of *IEEE MultiMedia*, a model management strategy is proposed based on triangular meshes, in which neighborhood buildings are considered as nodes and connected. Each node in the mesh has a set of level-of-detail (LOD) models, including high-, medium-, and low-precision models. Besides a model file, the high-precision LOD of the node can be a subtriangular mesh. The 3D models in a complex scene can be flexibly managed with some nested triangular meshes. According to the experimental results, the proposed method effectively achieves the progressive downloading, dynamic loading, and real-time display for a large-scale 3D scene. Its performance is better than the traditional methods.

IEEE Pervasive Computing

A Conversational Robot to Conduct Therapeutic Interventions for Dementia

Verbal communication is an essential component of effective nonpharmacological interventions for people with dementia (PwD). The authors of this article from the April–June 2019 issue of *IEEE Pervasive Computing* describe Eva, a conversational robot developed to conduct therapeutic interventions for PwD. A previously reported study conducted with Eva using a Wizard-of-Oz approach proved that it successfully engaged PwD with the sessions. This article reports improvements to Eva that allow the robot to guide the therapy sessions without human intervention and findings from its deployment in a geriatric residence. These improvements include the automatic generation of a therapy script tailored to the profile and preferences of the participants, expectations about the type and length of responses by participants to certain queries, and strategies to recover from communication breakdowns. A user study with five PwD shows that when acting in fully autonomous mode, Eva is as effective in engaging participants in the therapy as with the Wizard-of-Oz condition, and that communication breakdowns are adequately resolved.

IEEE Security & Privacy

Stealing, Spying, and Abusing: Consequences of Attacks on Internet of Things Devices

The authors of this article from the September/October 2019 issue of *IEEE Security & Privacy* studied the security practices of a diverse set of Internet of Things (IoT) devices with different architectures. They found vulnerabilities that can be exploited to launch novel attacks. The real-world implications of IoT attacks show the risks associated with these new technologies and can help us articulate the need for better security practices.

IEEE Software

Perceptions of Gender Diversity's Impact on Mood in Software Development Teams

Gender inequality persists in IT teams. The authors of this article from the September/October 2019 issue of *IEEE Software* examine how gender composition affects the workplace atmosphere. They discuss the problem of gender discrimination and consider methods to reduce inequality.

IT Professional

Toward a Blockchain-Enabled Crowdsourcing Platform

Crowdsourcing has been pursued as a way to leverage the power of the crowd for many purposes in diverse sectors, including collecting information, aggregating funds, and gathering employees. Data integrity and nonrepudiation are of utmost importance in these systems and are currently not guaranteed. Blockchain technology has been proven to improve on these aspects. In this article from the September/October 2019 issue of *IT Professional*, the authors investigate the benefits that the adoption of blockchain technology can bring in crowdsourcing systems. To this end, they provide examples of real-life crowdsourcing use cases and explore the benefits of using blockchain, mainly as a database.

EDITOR'S NOTE

Blockchain to the Rescue

Many tough problems facing business, government, and individuals could be solved through indelible ledgers. Transparent, secure transaction records could help improve trust and efficiency in everything from payments to voting. Enter blockchain-based systems. This issue of *ComputingEdge* explores what makes blockchain such a powerful technology with the potential to transform numerous industries.

"On the Origins and Variations of Blockchain Technologies," from *IEEE Security & Privacy*, provides a history of blockchain going back to David Chaum's 1979 vault system. The authors describe the foundational elements of the technology and compare the properties of diverse blockchain systems. *IT Professional*'s "BLOCKCHAIN" discusses the technology's growing popularity with businesses and other organizations.

The first modern blockchain was implemented in the cryptocurrency bitcoin, and cryptocurrency remains the most common application of blockchain technology. *IEEE Internet Computing*'s "A Service-Oriented Perspective on Blockchain Smart Contracts" examines the underlying technology used in cryptocurrency platforms like Bitcoin and Ethereum. *Computer*'s "Cryptocurrencies: Transparency versus Privacy" warns that cybercriminals are sometimes able to expose the identity of cryptocurrency users despite pseudonyms and concealed IP addresses.

Business professionals need to understand a variety of new technologies—not just blockchain—in order to compete. "Skills and Competencies for Digital Transformation," from *IT Professional*, provides an overview of the high-tech tools that companies should consider implementing. *IEEE Software*'s "Ubiquitous Requirements Engineering: A Paradigm Shift that Affects Everyone," describes the evolving role of software engineering in digital transformation, particularly in addressing the needs of diverse users.

The Internet of Things (IoT) is one of the crucial technologies that modern businesses need to employ. In "The IoT and Digital Transformation: Toward the Data-Driven Enterprise," from *IEEE Pervasive Computing*, the authors propose a process for companies that want to adopt IoT solutions. Healthcare is among the industries that can benefit from the IoT, as shown in *IEEE Intelligent Systems*' "Extending Patient-Chatbot Experience with Internet of Things and Background Knowledge: Case Studies with Healthcare Applications."

2469-7087/20 © 2020 IEEE

On the Origins and Variations of Blockchain Technologies

Alan T. Sherman, Farid Javani, Haibin Zhang, and Enis Golaszewski | University of Maryland, Baltimore County

We explore the origins of blockchain technologies to better understand the enduring needs they address. We identify the five key elements of a blockchain, show the embodiments of these elements, and examine how these elements come together to yield important properties in selected systems. To facilitate comparing the many variations of blockchains, we also describe the four crucial roles of common blockchain participants. Our historical exploration highlights the 1979 work of David Chaum, whose vault system embodies many of the elements of blockchains.

Understanding Blockchains

With myriad blockchain distributed ledger systems in existence, more than 550 associated patent applications under review, and much associated hype, it can be difficult to make sense of these systems, their properties, and how they compare. Through exploring the origins of these technologies, including David Chaum's 1979 vault system, we provide insights and a clear and useful way to think about blockchains. Our historical perspective distills important ideas, identifies enduring needs, and shows how changing technologies can satisfy those needs. This perspective will help people understand where blockchains came from, whether they are important, and if they will persist. (For a complete list of references, see A. Sherman et al.)[1]

Elements of Blockchains

Blockchains provide a mechanism through which mutually distrustful remote parties (nodes) can reach consensus on the state of a ledger of information. To trace the origins of these technologies, we start by identifying their essential elements informally. A blockchain is a distributed ledger comprising blocks (records) of information, including information about transactions between two or more parties. The blocks are cryptographically *linked* to create an immutable ledger. Nodes may append information to the ledger through invoking transactions. An *access policy* determines who may read the information. A *control policy* determines who may participate in the evolution of the blockchain and how new blocks may potentially be appended to the blockchain. A *consensus policy* determines which state of the blockchain is valid, resolving disputes should conflicting possible continuations appear.

As explained by Cachin and Vukolic,[2] a range of control policies is possible, including permissioned, consortium, private, and permissionless blockchains. In a permissioned blockchain, a body identifies and controls who may update state and issue transactions. A private blockchain is a permissioned blockchain controlled by one organization. A consortium blockchain is a permissioned blockchain involving a group of organizations. In a permissionless blockchain, anyone may potentially append new blocks, with the consensus policy (e.g., a majority of participants) determining which continuation is valid.

Blockchains achieve consensus and control (and, in particular, prevent double spending) in part through applying protocols and establishing high costs (both economic and computational) to modify the ledger. Typically, permissioned systems run faster than permissionless systems do because their control and consensus strategies depend on faster fault-tolerant protocols[3] rather than on time-consuming cryptographic proofs of work (PoWs), and they usually involve fewer nodes. Gencer et al. show that permissionless blockchains (such as Bitcoin and Ethereum) are much more centralized than many people assume: 20 mining pools control 90% of the computing power.

Some blockchains additionally support the idea of smart contracts, which execute terms of agreements between parties, possibly without human intervention. These agreements might be embodied as arbitrary computer programs including conditional statements.

Digital Object Identifier 10.1109/MSEC.2019.2893730
Date of publication: 20 March 2019

2469-7087/20 © 2020 IEEE

Embodiments of the Elements

Although the seminal paper on Bitcoin appeared in 2008 (with the mysterious author Satoshi Nakamoto),[4] most of the underlying technological ideas had arisen many years earlier. A blockchain is a type of distributed database, an idea that goes back to at least the 1970s (e.g., Wong[11]). More generally, the idea of record keeping goes back millennia, including to ancient Mesopotamia. Kanare describes proper methods for scientific logging, including the idea of preserving all transaction records, in addition to the history of any modifications to the collected data—ideas that are found in many systems (e.g., Hyperledger Fabric).

The idea of immutably chaining blocks of information with a cryptographic hash function appears in the 1979 dissertation of Ralph Merkle at Stanford, in which Merkle explains how information can be linked in a tree structure now known as a *Merkle hash tree*. A linear chain is a special case of a tree, and a tree provides a more efficient way of chaining information than does a linear chain. Subsequently, in 1990, Haber and Stornetta applied these ideas to time-stamp documents, creating the company Surety in 1994. These prior works, however, do not include other elements and techniques of blockchain.

To prevent an adversary from unduly influencing the consensus process, many permissionless systems require that new blocks include a proof of computational work. Nakamoto's paper cites Back's[5] 2002 effective construction from Hashcash. In 1992, Dwork and Naor proposed proof of computation to combat junk mail. The idea and a construction underlying PoW, however, may be seen in an initial form in 1974 in Merkle's puzzles,[6] which Merkle proposed to implement public-key cryptography. Bitcoin was the first to use PoW for both mining and achieving consensus.

PoW aims, in part, to defend against Sybil attacks, in which adversaries attempt to forge multiple identities and use those forged identities to influence the consensus process. With PoW, however, a node's influence on the consensus process is proportional to its computational power: forging multiple identities that share the adversary's given computational power does not help. To adapt to varying amounts of available computational resources, PoW systems dynamically throttle the difficulty of the PoW problem to achieve a certain target rate at which the problems are solved.

Permissioned blockchains can be modeled using the concept of (Byzantine fault-tolerant) state machine replication, a notion proposed in 1978 by Lamport and, later, concisely formalized by Schneider. State machine replication specifies what are the transactions and in what order they are processed, even in the presence of (Byzantine) faults and unreliable communications.[3] Thereby, to achieve a strong form of transaction consensus, many permissioned systems build on the ideas from the 1998 Paxos protocol of Lamport[7] (which deals only with crash failures) and from the 2002 Practical Byzantine Fault Tolerance protocol of Castro and Liskov. Nakamoto observed that the permissionless Bitcoin system realizes Byzantine agreement in open networks.

Arguably, many of the elements of blockchains are embodied in David Chaum's 1979 vault system,[8] described in his 1982 dissertation[9] at Berkeley, including detailed specifications. Chaum describes the design of a distributed computer system that can be established, maintained, and trusted by mutually suspicious groups. It is a public record-keeping system with group membership consistency and private transaction computations that protects individual privacy through physical security. The building blocks of this system include physically secure vaults, existing cryptographic primitives (symmetric and asymmetric encryption, cryptographic hash functions, and digital signatures), and a new primitive introduced by Chaum—threshold secret sharing.[8] Chaum's 1982 work went largely unnoticed, apparently because he never made any effort to publish it in a conference or journal, instead pursuing different approaches to achieving individual privacy.

In Chaum's system, each vault signs, records, and broadcasts each transaction it processes. Chaum states, "Because the aggregate includes COMPRESSED_HISTORY, the [cryptographic] checksum is actually 'chained' through the entire history of consensus states."[9] He further says, "Nodes remember and will provide all messages they have output—each vault saves all it has signed, up to some limit, and will supply any saved thing on request; only dead vaults can cause loss of recently signed things."[9]

Chaum's system embodies a mechanism for achieving membership consistency: "Among other things, the algorithms must provide a kind of synchronization and agreement among nodes about allowing new nodes into the network, removing nodes from the network, and the status of nodes once in the network."[9] The system also embodies a weak form of transaction consensus, albeit vaguely described and apparently not supporting concurrent client requests: "If the output of one particular processor module is used as the output for the entire vault, the other processors must be able to compare their output to its output, and have time to stop the output on its way through the isolation devices."[9] The consensus algorithm involves majority vote of nodes based on observed signed messages entering and leaving vaults.

Chaum created his vaults system before the emergence of the terms *permissioned* and *permissionless* blockchains, and his system does not neatly fall into either of these discrete categories. In Chaum's system, each node identifies itself uniquely by posting a public key, authenticated by level 2 trustees. For this reason, some people may consider Chaum's system a permissioned blockchain.

This narrow view, however, diminishes the fact that each node can be authorized in a public ceremony independently from any trustee. During this ceremony, vaults are assembled from bins of parts, which the public (not necessarily nodes) can inspect and test—a procedure that inspired Chaum to coin the more limited phrase *cut and choose*. Regardless of whether one views some configurations of Chaum's vaults as permissionless systems, the trust bestowed through the public ceremony creates a system whose trust model is the antithesis of that of a private (permissioned) blockchain. For these reasons, we consider Chaum's system publicly permissioned.

Chaum assumes, essentially, a best-effort broadcast model, and he does not provide mechanisms for achieving consensus with unreliable communications—technologies that subsequently have been developed and applied in modern permissioned systems. Chaum's dissertation does not include the ideas of PoW, dynamic throttling of work difficulty, and explicit smart contracts (though Chaum's vaults support arbitrary distributed private computation).

Unlike in most blockchain systems, nodes in Chaum's system hold secret values, which necessitates a more complex mechanism for restarting after failures. Using what Chaum calls *partial keys*, any vault can back up its state securely by encrypting it with a key and then escrowing this key using what we now call *threshold secret sharing*. After reading Chaum's February 1979 technical report[8] that describes partial keys, Adi Shamir published an elegant alternate method for secret sharing in November 1979.

Chaum also notes that pseudonyms can play an important role in effecting anonymity: "Another use allows an individual to correspond with a record keeping organization under a unique pseudonym which appears in a roster of acceptable clients."[9] To enable private transactions for blockchains, engineers are exploring the application of trusted execution environments, continuing an approach fundamental in Chaum's vaults.

Table 1. A timeline of selected discoveries in cryptography and blockchain technology.

1970	James Ellis, public-key cryptography discovered at Government Communications Headquarters (GCHQ) in secret
1973	Clifford Cocks, RSA cryptosystem discovered at GCHQ in secret
1974	Ralph Merkle, cryptographic puzzles (paper published in 1978)
1976	Diffie and Hellman, public-key cryptography discovered at Stanford
1977	Rivest, Shamir, and Adleman, RSA cryptosystem invented at the Massachusetts Institute of Technology
1979	David Chaum, vaults and secret sharing (dissertation in 1982)
1982	Lamport, Shostak, and Pease, Byzantine Generals Problem
1992	Dwork and Naor, combating junk mail
2002	Adam Bach, Hashcash
2008	Satoshi Nakamoto, Bitcoin
2017	Wright and Savanah, nChain European patent application (issued in 2018)

In 1994, Szabo[10] coined the term *smart contract*, but the idea of systematically applying rules to execute the terms of an agreement has a long history in trading systems. For example, in 1949, with a system involving ticker tapes and humans applying rules, Future, Inc. generated buy and sell orders for commodities. Recently, so-called hybrid blockchains have emerged, which combine Byzantine fault-tolerant state machine replication with defenses against Sybil attacks—for example, PeerCensus, ByzCoin, Solidus, Hybrid Consensus, Elastico, OmniLedger, and RapidChain.

Also, Hyperledger (an umbrella project involving Fabric, a system for permissioned blockchains) and Ethereum (a platform for public blockchains) have joined forces. Recently, researchers have applied game theory to model and analyze the behaviors of players and mining pools in blockchain-based digital currencies (see Dhamal and Lewenberg). Table 1 chronicles some of the important cryptographic discoveries underlying blockchain technologies. For example, in 2018, the European Patent Office issued the first patent on blockchain—a method for enforcing smart contracts.

Comparison of Selected Blockchain Systems

To illustrate how the elements come together in actual blockchain systems, we compare a few selected systems, including Chaum's vaults, Bitcoin, Dash, Corda, and Hyperledger Fabric, chosen for diversity. Table 2 describes how each of these systems carries out the four crucial participant roles of any blockchain defined ahead. For more context, Table 3 characterizes a few important properties of these systems and of one additional system—Ethereum.

In his vault system, Chaum[9] identifies four crucial participant roles of any blockchain, which we call *watchers*, *doers*, *executives*, and *czars*. The watchers passively observe and check the state of the ledger. The doers (level 1 trustees) carry out actions, including serving state. The executives (level 2 trustees) sign (or otherwise attest to) the blocks. The czars (level 3 trustees) change the executives and their policies. Chaum refers to these participants as *bodies*,[9] leaving it unclear whether they could be algorithms.

Although most systems do not explicitly specify these roles, all systems embody them, though with varying nuances. For example, many people naively think of Bitcoin as a fully distributed system free of any centralized control, but, in fact, Bitcoin's core developers—as is true for all distributed systems—carry out the role of czars, changing the underlying software that implements policy. Despite these significant powers, the control structure is still more distributed (anyone can potentially become a core developer) than for a permissioned system controlled entirely by a prespecified entity. In Bitcoin, in each round, the winning miner (a doer) becomes an executive for that round. It is instructive to understand how each blockchain system allocates the four participant roles.

Table 3 illustrates some of the possible variations of blockchains, including varying control and consensus policies as well as different types of smart contracts. Whereas most blockchain systems maintain a single chain, Corda supports multiple independent chains, per node or among subsets of nodes. Similarly, Chaum's system also supports multiple chains. While most blockchains require each node to maintain the same state, Corda's and Chaum's systems do not.

Conflicts and Challenges

Because blockchain technologies address enduring needs for permanent, indelible, and trusted ledgers, they will likely be around in various forms for a long time. There are, however, some troubling fundamental conflicts that have not been solved. These conflicts include tensions between the following pairs of potentially dissonant concerns: privacy and indelibility, anonymity and accountability, stability and alternative future continuations, and current engineering choices and long-term security. For example, recent European privacy laws grant individuals the right to demand that their personal data be erased from most repositories (the right to be forgotten). Satisfying this erasure requirement is highly problematic for indelible blockchains, especially for ones whose nodes lack physical security.

An attraction of blockchains is their promise of stability enforced through consensus, yet sometimes the nodes cannot agree, resulting in a fork and associated possible splits in the continuations of the chain. In a hard fork, level 3 trustees issue a significant change in the rules that is incompatible with the old rules. In a

Table 2. Alignment of participant roles across five blockchain systems.

Role	**Chaum, 1982** *A flexible system based on vaults*	**Bitcoin, 2008** *A permissionless system using PoW*	**Dash, 2014** *A system that speeds up Bitcoin with a masternode network*	**Corda, 2016** *A permissioned system with smart contracts*	**Hyperledger Fabric, 2016** *A permissioned system with smart contracts*
Watchers Passively check state	Any computer online[9]	Nodes (distinct from full nodes)	Any computer online	Nodes	Peers
Doers Carry out actions, including serving state	Level 1 trustee	Full nodes	Miners	Nodes	Peers
Executives Sign blocks (or otherwise attest to them)	Level 2 trustee (promoted from level 1 by czars)[9]	Winning miner (promoted from doers each round)	Winning masternode (promoted by an algorithm from the masternode network, which anyone may join for 1,000 Dash)	Nodes (each node is an executive for its Corda blocks, called *states*)	Endorsing peers
Czars Change executives and their policies	Level 3 trustee[9]	Core developers	Quorum of masternodes	Permissioning service	Endorsement policies

Table 3. Three properties of several distributed ledger systems.

System	Permissioned?	Basis of Consensus	Smart Contracts
Chaum, 1982	Permissioned, with option for publicly permissioned	Weak consensus; does not handle concurrent client requests	Private arbitrary distributed computation
Bitcoin, 2008	Permissionless	PoW	Conditional payment and limited smart contracts through scripts
Dash, 2014	Combination	Proof of stake	No
Ethereum, 2014	Permissionless	PoW	Yes, nonprivate Turing complete objects
Hyperledger Fabric, 2015	Permissioned	Based on state machine replication	Yes, off-chain
Corda, 2016	Permissioned	Based on state machine replication	Yes (set of functions), including explicit links to human language

soft fork, there is a less severe change in the rules for which the old system recognizes valid blocks created by the new system (but not necessarily vice versa).

Security engineers must commit to particular security parameters, hash functions, and digital signatures methods.

No such choice can remain computationally secure forever in the face of evolving computer technology, including quantum computers and other technologies not yet invented. The hopeful permanence of blockchains is dissonant with the limited-time security of today's engineering choices.

Additional challenges facing blockchains include the huge amounts of energy spent on blockchain computations (especially PoW), the high rates at which ledgers grow, and the associated increases in transaction latency and processing time (Bitcoin's ledger is currently more than 184 GB).

As of September 2018, the hash rate for Bitcoin exceeded 50 million TH/s, consuming more than 73 TWh of power per day, more than the amount consumed by Switzerland. These hashes were attempts to solve cryptographic puzzles of no intrinsic value (finding an input that, when hashed, produces a certain number of leading zeroes), and almost all of these computations went unused. Attempts, such as Primecoin and others, to replace cryptographic hash puzzles with useful work (e.g., finding certain types of prime integers) are challenging because it is very hard to find useful problems that have assured difficulty and whose level of difficulty can be dynamically throttled. Some researchers are exploring alternatives to PoW, such as proof of space, proof of stake, and proof of elapsed time.

To understand blockchain systems, it is helpful to view them in terms of how the watchers, doers, executives, and czars carry out their functions under the guidance of the access, control, and consensus policies. This systematic abstract view helps focus attention on crucial elements and facilitates a balanced comparison of systems. Blockchains address many longstanding inherent needs for indelible ledgers, from financial transactions to property records and supply chains. With powerful existing cryptographic techniques, a wide set of available variations, and a large amount of resources allocated to these technologies, blockchains hold significant potential. ■

Acknowledgments

We thank Dan Lee, Linda Oliva, and Konstantinos Patsourakos for their helpful comments. Alan T. Sherman was supported in part by the National Science Foundation under Scholarship for Service grant 1241576.

References

1. A. Sherman, F. Javani, H. Zhang, and E. Golaszewski, On the origins and variations of blockchain technologies. 2018. [Online]. Available: http://arxiv.org/abs/1810.06130
2. C. Cachin and M Vukolic, "Blockchain consensus protocols in the wild," in *Proc. 31st Int. Symp. Distributed Computing*, 2017, vol. 1, pp. 1–16.
3. L. Lamport, R. Shostak, and M. Pease, "The Byzantine generals problem," *ACM Trans. Programming Languages Syst.*, vol. 4, no. 3, pp. 382–401, 1982. [Online]. Available: https://dl.acm.org/citation.cfm?doid=357172.357176
4. S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Bitcoin, 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf
5. A. Back, "Hashcash: A denial of service counter-measure," Hashcash, 2002. [Online]. Available: http://www.hashcash.org/papers/hashcash.pdf

This article originally appeared in
IEEE Security & Privacy, *vol. 17, no. 1, 2019.*

6. R. C. Merkle, “Secure communications over insecure channels,” *Commun. ACM*, vol. 21, no. 4, pp. 294–299, 1978. [Online]. Available: https://dl.acm.org/citation.cfm?doid=359460.359473
7. L. Lamport, “The part-time parliament,” *ACM Trans. Comput. Syst.*, vol. 16, no. 2, pp. 133–169, 1998. [Online]. Available: https://dl.acm.org/citation.cfm?doid=279227.279229
8. D. L. Chaum, “Computer systems established, maintained, and trusted by mutually suspicious groups,” Elect. Eng. Res. Lab., Univ. California, Berkeley, Tech. Memo. UCB/ERL/M79/10, 1979.
9. D. L. Chaum, “Computer systems established, maintained and trusted by mutually suspicious groups,” Ph.D. dissertation, Dept. Comput. Sci., Univ. California, Berkeley, 1982.
10. N. Szabo, “Smart contracts,” 1994. [Online]. Available: http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html
11. E. Wong, “Retrieving dispersed data from SDD-1: A system for distributed databases,” in *Proc. 2nd Berkeley Workshop Distributed Data Management and Comput. Networks*, May 1977, pp. 217–235.

Alan T. Sherman is a professor of computer science at the University of Maryland, Baltimore County. His research interests include secure voting, applied cryptography, and cybersecurity education. He is a Senior Member of the IEEE. Contact him at sherman@umbc.edu.

Farid Javani is a Ph.D. student at the University of Maryland, Baltimore County. Contact him at javani1@umbc.edu.

Haibin Zhang is an assistant professor in the Department of Computer Science and Electrical Engineering at the University of Maryland, Baltimore County. Haibin received a Ph.D. from the University of California, Davis, in 2001. His research interests include distributed computing and secure blockchains. Contact him at hbzhang@umbc.edu.

Enis Golaszewski is a Ph.D. student at the University of Maryland, Baltimore County. Contact him at golaszewski@umbc.edu.

Digital Object Identifier 10.1109/MSEC.2019.2900896

IEEE COMPUTER SOCIETY

PURPOSE: The IEEE Computer Society is the world's largest association of computing professionals and is the leading provider of technical information in the field.

MEMBERSHIP: Members receive the monthly magazine *Computer*, discounts, and opportunities to serve (all activities are led by volunteer members). Membership is open to all IEEE members, affiliate society members, and others interested in the computer field.

COMPUTER SOCIETY WEBSITE: www.computer.org

OMBUDSMAN: Direct unresolved complaints to ombudsman@computer.org.

CHAPTERS: Regular and student chapters worldwide provide the opportunity to interact with colleagues, hear technical experts, and serve the local professional community.

AVAILABLE INFORMATION: To check membership status, report an address change, or obtain more information on any of the following, email Customer Service at help@computer.org or call +1 714 821 8380 (international) or our toll-free number, +1 800 272 6657 (US):

- Membership applications
- Publications catalog
- Draft standards and order forms
- Technical committee list
- Technical committee application
- Chapter start-up procedures
- Student scholarship information
- Volunteer leaders/staff directory
- IEEE senior member grade application (requires 10 years practice and significant performance in five of those 10)

PUBLICATIONS AND ACTIVITIES

***Computer*:** The flagship publication of the IEEE Computer Society, *Computer* publishes peer-reviewed technical content that covers all aspects of computer science, computer engineering, technology, and applications.

Periodicals: The society publishes 12 magazines and 18 journals. Refer to membership application or request information as noted above.

Conference Proceedings & Books: Conference Publishing Services publishes more than 275 titles every year.

Standards Working Groups: More than 150 groups produce IEEE standards used throughout the world.

Technical Committees: TCs provide professional interaction in more than 30 technical areas and directly influence computer engineering conferences and publications.

Conferences/Education: The society holds about 200 conferences each year and sponsors many educational activities, including computing science accreditation.

Certifications: The society offers three software developer credentials. For more information, visit www.computer.org/certification.

BOARD OF GOVERNORS MEETING

22 – 23 January: Costa Mesa, California

IEEE

revised 19 December 2019

EXECUTIVE COMMITTEE

President: Leila De Floriani
President-Elect: Forrest Shull
Past President: Cecilia Metra
First VP: Riccardo Mariani; **Second VP:** Sy-Yen Kuo
Secretary: Dimitrios Serpanos; **Treasurer:** David Lomet
VP, Membership & Geographic Activities: Yervant Zorian
VP, Professional & Educational Activities: Sy-Yen Kuo
VP, Publications: Fabrizio Lombardi
VP, Standards Activities: Riccardo Mariani
VP, Technical & Conference Activities: William D. Gropp
2019–2020 IEEE Division VIII Director: Elizabeth L. Burd
2020-2021 IEEE Division V Director: Thomas M. Conte
2020 IEEE Division VIII Director-Elect: Christina M. Schober

BOARD OF GOVERNORS

Term Expiring 2020: Andy T. Chen, John D. Johnson, Sy-Yen Kuo, David Lomet, Dimitrios Serpanos, Hayato Yamana
Term Expiring 2021: M. Brian Blake, Fred Douglis, Carlos E. Jimenez-Gomez, Ramalatha Marimuthu, Erik Jan Marinissen, Kunio Uchiyama
Term Expiring 2022: Nils Aschenbruck, Ernesto Cuadros-Vargas, David S. Ebert, William Gropp, Grace Lewis, Stefano Zanero

EXECUTIVE STAFF

Executive Director: Melissa A. Russell
Director, Governance & Associate Executive Director: Anne Marie Kelly
Director, Finance & Accounting: Sunny Hwang
Director, Information Technology & Services: Sumit Kacker
Director, Marketing & Sales: Michelle Tubb
Director, Membership Development: Eric Berkowitz

COMPUTER SOCIETY OFFICES

Washington, D.C.: 2001 L St., Ste. 700, Washington, D.C. 20036-4928; **Phone:** +1 202 371 0101; **Fax:** +1 202 728 9614; **Email:** help@computer.org
Los Alamitos: 10662 Los Vaqueros Cir., Los Alamitos, CA 90720; **Phone:** +1 714 821 8380; **Email:** help@computer.org

MEMBERSHIP & PUBLICATION ORDERS

Phone: +1 800 678 4333; Fax: +1 714 821 4641; Email: help@computer.org

IEEE BOARD OF DIRECTORS

President: Toshio Fukuda
President-Elect: Susan K. "Kathy" Land
Past President: José M.F. Moura
Secretary: Kathleen A. Kramer
Treasurer: Joseph V. Lillie
Director & President, IEEE-USA: Katherine J. Duncan
Director & President, Standards Association: Robert S. Fish
Director & VP, Educational Activities: Stephen Phillips
Director & VP, Membership & Geographic Activities: Maike Luiken
Director & VP, Publication Services & Products: Tapan Sarkar
Director & VP, Technical Activities: Kazuhiro Kosuge

Department

BLOCKCHAIN

George Strawn
National Academies of Sciences, Engineering, and Medicine

This article originally appeared in IT Professional, *vol. 21, no. 1, 2019.*

"YABB," YET ANOTHER book on blockchain (to modify the acronym YACC—yet another complier compiler), *Life After Google*, was published in July of 2018 by the prolific author George Gilder. (Amazon currently lists 75 books on the topic.) As Gilder lays out in the book, he believes that the client–server model of current internet usage will be succeeded by a peer-to-peer (P2P) model employing blockchain: the technology that enabled the cryptocurrency bitcoin a decade ago. Another YABB is IBM's short *Blockchain for Dummies* (https://public.dhe.ibm.com/common/ssi/ecm/xi/en/xim12354usen/ibm-blockchain_second-edition_final_XIM12354USEN.pdf), which is available for free download and has the more modest goal of showing how *blockchain for business ledgers* is available now for practical use. In this paper, I will review some related characteristics of the Internet, of P2P, and of blockchain. Then, I will describe how blockchain is "ready for business use," and finally, I will comment on its potential impact on 21st century employment and business.

THE INTERNET, P2P, AND BLOCKCHAIN

One of the goals of the Internet architecture was to minimize points of failure. The switching centers that characterized the telephone network were such points of failure. If a switching center were to be destroyed, the telephones in that area would become disconnected from the network. (Cold War worries about a nuclear arrack made this a major concern.) By switching packets instead of circuits and by distributing the switching function to every Internet router, that goal was achieved.

On the other hand, very little thought was given to other dimensions of network security, and so, we have been playing Internet security catch-up ever since. (One might say that it is an ill wind that blows no good, since Internet security jobs are in plentiful supply.) But as increasingly more functions of society are transferred to the Internet, its lack of security has become a major societal problem. Gilder and others believe that this lack of security is the Achilles heel of today's Internet, and the reason that P2P architecture and secure blockchain technology will supersede it.

The original Internet architecture was in fact P2P. This simply means that any internet node could both provide services to other nodes and/or ask them to provide services. (For more depth, see https://en.m.wikipedia.org/wiki/Peer-to-peer.) For example, the file transfer protocol was/is bidirectional: Any node can send and/or receive files. As the Internet matured in the 1990s and 2000s, important services arose that were unidirectional, for example, Google searches, Amazon purchases, and Facebook friends. These services were provided by nodes that came to be called servers, and nodes that utilized those services were called clients.

The client–server model is subject to various security breaches (of course, so is P2P). For example, distributed denial-of-service attacks flood a server with so many requests for service that it

Digital Object Identifier 10.1109/MITP.2018.2879244
Date of current version 26 February 2019.

2469-7087/20 © 2020 IEEE

shuts down. Also, most servers require usernames and passwords from clients (and perhaps credit card numbers). So, many users (me included) have hundreds of usernames and passwords, which I am supposed to remember and never write down. (This situation seems to me to provide more liability protection for the server than security for the client.) And security breaches of servers are legion, yielding crooks millions of usernames, passwords, and credit card numbers. Such hacks are of increasing importance as online banking and other significant transactions are conducted online.

Blockchain technology is simply a distributed ledger on a P2P network whose transactions cannot be erased or altered (see https://en.m.wikipedia.org/wiki/Blockchain for implementation details). As new transactions occur and are verified, they are copied onto all copies of the ledger. It has been said that blockchain/P2P might to for transactions what the Internet/Web did for Information.

BLOCKCHAIN FOR BUSINESS TRANSACTIONS

Speaking of transactions, they are the business activity that blockchain is ready to facilitate, according to IBM and others. Companies record transactions in ledgers, and traditionally, each company keeps its own ledger. Blockchain technology enables a single, shared ledger for all the companies engaging in related transactions. Moreover, this shared ledger has several pleasing security characteristics. First, it is copied onto the computers of all participating companies, making loss of data extremely unlikely. Second, once a transaction has been "agreed to," it cannot be changed or deleted. This provides a new level of "technological trust" that has traditionally been provided by trusted third parties. Regarding how a transaction is agreed to, the business use just described requires only a simple vote by the companies involved rather than an expensive "mining" activity as in the bitcoin application.

USE CASES, EMPLOYMENT, AND OTHER IMPLICATIONS

Chapter four of *Blockchain for Dummies* describes a plethora of transaction/ledger use cases the fall within the sphere of blockchain. These use cases occur in a wide range of areas: financial services, multinational policy management, government, supply chain management, and health care. Since these use cases replace (or at least reduce) the need for trusted third party oversight, that reduction in employment is obvious. Less obvious is the fact that many of these use cases can contain "smart contracts" that automate various follow-on functions once a transaction has been completed (e.g., automatic payment once a shipment has been received). Thus, the need for fewer manual steps may extend well beyond transaction management.

Perhaps even more important is that these use cases typically take a significant amount of time to complete. The use of blockchain could cut days and weeks to hours and minutes, and since time is money, use of this technology could be doubly cost-saving. Of course, in addition to increasing efficiency, effectiveness could be improved as well. As Chapter Three of Dummies explains, blockchain can reduce business network (information, interaction, and innovation) frictions in a number of ways. Share ledgers, P2P transactions, and smart contracts are at the center of this business innovation.

In a recent blockchain report (https://public.dhe.ibm.com/common/ssi/ecm/gb/en/gbe03835usen/gbe03835usen-00_GBE03835USEN.pdf) drawing on 2965 conversations with C-suite executives, IBM reported the following industry statistics: Over one third of organizations across all industries and regions are already considering or are actively engaged with blockchain, and 66% of early adopters—or explorers—intend to adopt a new platform business model that breaks the boundaries of traditional market exchanges. It would seem this train is moving.

WHAT IS NEXT?

Blockchain burst on the scene as the bitcoin technology only ten years ago. It was the result of innovative software, and hardware innovations followed as the computationally expensive mining confirmation of transactions was optimized. On the other hand, 3-D printing (a.k.a. additive engineering) resulted from hardware innovations over 30 years ago and is now also moving (Amazon lists 100 books and gadgets on the subject). As Moore's and related laws continue to lower the cost and raise the performance of IT systems, predicting what is next requires matching new price points, innovative thinking, and society's needs and desires. It is an exciting ride with far to go!

12.19
Computer
Technology Predictions
2022
2021
9
2020

Spotlight

A Service-Oriented Perspective on Blockchain Smart Contracts

Florian Daniel
Politecnico di Milano

Luca Guida
Politecnico di Milano

***Abstract*—Smart contracts turn blockchains into distributed computing platforms. This paper studies whether smart contracts as implemented by a state-of-the-art blockchain technology may serve as a component technology for a computing paradigm like service-oriented computing in the blockchain, in order to foster reuse and increase cost-effectiveness.**

■ **A BLOCKCHAIN IS** a shared, distributed ledger, that is, a log of transactions that provides for persistency and verifiability of transactions.[1] A *transaction* is a cryptographically signed instruction constructed by a user of the blockchain,[2] for example, the transfer of cryptocurrency from one account to another. Transactions are grouped into blocks, linked and secured using cryptographic hashes. A *consensus protocol* enables the nodes of the blockchain network to create trust in the state of the log and makes blockchains inherently resistant to tampering.[3] Thanks to these properties, blockchain technology is able to eliminate the need for a middleman from the management of transactions, such as a bank in the transfer of money.

Digital Object Identifier 10.1109/MIC.2018.2890624
Date of current version 6 March 2019.

Next to logging transactions, blockchain platforms support the execution of pieces of code, so-called *smart contracts*,[4,5] able to perform computations inside the blockchain. For example, a smart contract may be used to automatically release a given amount of cryptocurrency upon the satisfaction of a condition agreed on by two partners. If we put multiple smart contracts (and partners) into communication, we turn the blockchain into a proper distributed computing platform.[6] This makes the technology appealing to application scenarios that ask for code execution that is reliable, verifiable, and transactional.

For example, Xu *et al.*[7] propose the use of smart contracts as software connectors for reliable, decentralized data sharing, while Weber *et al.*[8] propose the integration of multiple smart contracts for distributed business process execution. The first example aims to support data providers in publishing data sets and data consumers in finding and selecting data sets; using cryptocurrency, data providers are automatically paid according to the value of the provided data, establishing an open, blockchain-based marketplace for data. The second example generates smart contracts starting from a BPMN choreography diagram and puts them into direct communication; the idea is to enable the execution of business processes even among potentially untrusted partners. The common ingredients of both examples are smart contracts and verifiable transactions.

Developing applications that integrate multiple smart contracts is however not easy, and today's predominant ad-hoc development practice will not be able to scale and be sustainable in the long term. In fact, Atzei *et al.*[9] show that already today even simple smart contracts are often affected by a variety of security vulnerabilities. Nikolić *et al.*[10] show that several of the smart contracts deployed on Ethereum either "lock funds indefinitely, leak them carelessly to arbitrary users, or can be killed by anyone." Singh and Chopra[11] go beyond implementation aspects and discuss existing sociotechnical limitations of smart contracts, such as lack of control, lack of understanding, and lack of social meaning.

We argue that future blockchain applications ask for abstractions, methods, and instruments that help developers to cope with complexity, such as those proposed by service-oriented computing (SOC). In fact, the characteristics of the described data sharing scenario directly map to those of SOC (service provider, service consumer, service broker), yet smart contracts still lack equivalent support for description, discovery, and the specification of nonfunctional properties. Similarly, the business process scenario resembles very much that of service-based business processes, yet the smart contracts generated in the scenario are tailored to specific tasks and partner interactions and are not directly applicable in processes with different partners and/or choreography needs. That is, while they present significant opportunities for reuse, they do not yet explore them.

In the following, we thus look at smart contracts from an SOC perspective and study their suitability as elementary pieces for a blockchain-based, distributed computing paradigm. The assumption is that principled reuse not only helps to lower complexity but also increases correctness by design.

BLOCKCHAIN AND SMART CONTRACTS

Next to Bitcoin, several alternative platforms have emerged over the last few years. Besides the *type of cryptocurrency* adopted as incentive mechanism, these platforms distinguish themselves by few key properties.

The *access policy* tells who can participate in the blockchain network. *Public* blockchains allow anyone to join and to access the information stored in the blockchain via the Internet; *private* blockchains are restricted to private networks and selected nodes only.

The *validation policy* tells who among the nodes can participate in consensus creation and deploy smart contracts. *Permissionless* blockchains allow every node to perform both; *permissioned* blockchains limit these capabilities to special nodes only, e.g., qualified through direct invitation.

The *consensus protocol* specifies how trust is created among participants: *Proof of work* (e.g., adopted by Bitcoin) requires nodes, so-called miners, to invest significant hashing power to create trust. *Proof of stake* (Cardano) requires nodes to prove ownership of sufficient cryptocurrency to establish trust. *Byzantine Fault Tolerance* uses replication to establish trust in the state of the network, even if faced with failing network nodes. Variants are redundant BFT (Hyperledger Indy) and practical BFT (Quorum), which aim at increased redundancy and speed, respectively. Other notable consensus protocols are *proof of elapsed time* (Hyperledger Sawtooth), *proof of importance* (NEM), *proof of state* (Universa Blockchain Protocol), *Raft-based consensus* (Quorum), *stream-processing ordering services* (Hyperledger Fabric), and *Tempo* (Radix DLT).

The choice of the consensus protocol affects the *transaction processing time* (time till a transaction is added to a block) and the *transaction rate* (number of transactions processed per second). These properties and the access and validation policies determine a blockchain's ability to support different distributed computing scenarios.

Table 1. Core characteristics of four example blockchain platforms.

	Bitcoin	Ethereum	Hyperledger Fabric	Corda
Cryptocurrency	Bitcoin (BTC)	Ethereum (ETH)	No built-in currency	No built-in currency
Access policy	Public	Public	Private	Private
Validation policy	Permissionless	Permissionless	Permissioned	Permissioned
Consensus protocol	Proof of work	Proof of work (proof of stake under review*)	Voting-based algorithm (Apache Kafka)	Validity consensus, Uniqueness consensus
Transaction processing time (average)	~ 10 minutes	~ 15 seconds	Almost instantaneous	Almost instantaneous
Max transaction rate	~ 7 TPS	~ 20 TPS	3,500 + TPS	~ 170 TPS
Smart contract language	Bitcoin Script, high-level languages (BALZaC, BitML) compilable to Bitcoin native transactions	Solidity, Serpent, lowlevel Lisp-like language (LLL), Mutan	Go	JVM programming languages like Kotlin, Java
Turing completeness	No	Yes	Yes	Yes

[Online]. Available: https://cryptoslate.com/ethereums-proof-of-stake-protocol-in-review/

As for the implementation of smart contracts, each platform typically supports one or more *programming languages*. Some support general-purpose languages, such as C, C++, C#, F#, Go, Java, JavaScript, Kotlin, Objective-C, PHP, Python, Rust, and Visual Basic .Net. Others propose platform-specific languages, such as Bitcoin Script or Ethereum Solidity. The former are Turing complete, the latter not necessarily (e.g., Bitcoin Script is not).

In Table 1, we summarize these characteristics for four platforms: Bitcoin (bitcoin.org), the first blockchain platform; Ethereum (ethereum.org), the platform that first introduced Turing-complete smart contracts; Hyperledger Fabric (hyperledger.org/projects/fabric), a private, permissioned platform hosted by the Linux Foundation and supported by more than 200 industry leaders; and Corda (corda.net), a private, permissioned platform by a consortium of more than 200 financial institutions and technology firms with a focus on interoperability. These platforms represent an opportunistic selection (far from exhaustive) based on our own knowledge and the goal of communicating some of the diversity that characterizes current blockchain technology.

SERVICE ORIENTATION

Service orientation is commonly associated with the binomial SOAP/WSDL or the REST architectural style. Smart contracts use neither of these, so we fall back to the generic definition by Alonso *et al.*[12] who define services as "components that can be integrated into more complex distributed applications." In order to compare different web service technologies, Lagares Lemos *et al.*[13] distinguish services by their type, interaction style, interaction protocol, data format, and descriptor. We discuss these characteristics next for smart contracts, in order to enable identifying analogies and differences between the proposed service-oriented interpretation of smart contracts and traditional web service technologies. We specifically focus on Ethereum as such is currently the most used blockchain platform for smart contract development.

Contract Type

Components encapsulate data to be fetched and visualized or integrated and/or application logic to be interacted with. What the component delivers is a function of the type of the component. For smart contracts we can distinguish the following contract types:

- *Generic contracts* implement application logic, e.g., for deposit management, that can be invoked by blockchain clients or by other contracts; in general, this type of contract is stateful in that it maintains application state across interactions.
- *Libraries* implement one or more functions, e.g., a math library, that are meant for reuse by other contracts; libraries do not store internal variables and are stateless.
- *Data contracts* provide data storage services inside the blockchain, e.g., a client references manager, that are meant for use by other contracts; by design, they are stateful.
- *Oracles* deliver data services from the outside of the blockchain to the inside of the blockchain, e.g., currency conversation rates. Contracts cannot make calls outside the blockchain, as outside dependencies may prevent verifiability (conversion rates change over time). If data from the outside is needed, it can be pushed by clients to oracles using transactions; these then allow other contracts to query for the data.

Interaction Style

Integrating a component into a composite application usually does not only involve a one-shot query or call. It may be necessary to interact with the component multiple times and to establish some form of conversation with it. For smart contracts we have:

- *Pull* interactions enable a client or contract to initiate an interaction and to invoke a contract that otherwise would be passive; for instance, a client may invoke a contract to withdraw a deposit.
- *Push* interactions enable the contract to become active and to initiate an interaction with clients or other contracts; for instance, a contract may invoke a data contract to obtain a list of accounts to send cryptocurrency to.
- *Business-protocol*-based interactions support patterns that may involve multiple interactions and multiple clients or contracts; the protocol specifies the order of interactions and the roles of the involved parties.

As running smart contracts costs money, contracts are activated only in response to explicit invocations. A contract or a group of interacting contracts is thus always triggered by a client transaction, and independent, active behaviors are typically not supported.

Interaction Protocol

This tells how a component implements its interactions. Conventional web services use message-oriented protocols such as SOAP or HTTP, while all major programming languages also support RPC-like interactions (Remote Procedure Calls). Ethereum uses a message-based protocol supporting the following interaction features:

- *Transactions* are used by blockchain clients (the users of the blockchain) to create new contracts or to invoke existing contracts; once validated, which consumes cryptocurrency, transactions are added to the blockchain and remain publicly accessible.
- *Events* enable a contract to push information to the outside world in response to a transaction invoking the contract; when the transaction is added to the blockchain, also the event becomes publicly accessible.
- *Calls* (so-called *message calls*) are used by contracts to interact with each other in a fashion that uses different state spaces for each contract for isolation; calls are executed locally to each blockchain node and do not consume cryptocurrency.
- *Delegate calls* are used by contracts to invoke libraries in a fashion where functions are executed in one, the caller's, state space; delegate calls too are node-local and do not consume cryptocurrency.

If an interaction originates from a blockchain client, it uses JSON-RPC or is enacted using the command line; if it originates from a smart contract, the message is exchanged via RPC. Transactions contain a set of predefined parameters: the number of transactions sent by the sender, the amount of cryptocurrency the sender is willing to pay for consumed resources (so-called *gas*), the maximum consumable amount of gas, the address of the recipient, the amount of cryptocurrency to be transferred, possible signatures of the sender, and either the code of the contract to be created or input data to be processed. Events contain, among others, one or more topics that allow clients to search for and subscribe to events and a data field. Calls contain the sender and receiver addresses, a possible value and data; calls may return a value.

Data Format

The data format determines how exchanged data is formatted. Message-oriented interaction protocols typically support self-describing document formats like XML and JSON; RPC-oriented protocols enable the exchange of native data structures, such as Java or JavaScript objects, using an internal, binary format hidden to developers.

Data in Ethereum transactions and events is encoded using the Application Binary Interface (ABI), which specifies how functions are called and data are formatted. Clients either serialize data in a binary format on their own, e.g., when using the command line or by using a suitable library function, e.g., the function toPayload of the library web3.js. Values are encoded in sequential order and according to their data types and are not self-describing. In order to allow the receiver to identify which function is called, the sequence of values is preceded by 4 B of a Keccak-256 hash of the respective function signature. This allows everybody to parse the binary formatted data.

Data in message/delegate calls between contracts is exchanged by passing variables, masking the underlying ABI formatting.

Description

The final aspect of components is component description, which enables discovery and selection. For web services, description languages such as WSDL and WADL and semantics-oriented languages such as OWL-S, WSDL-S, and WSMO are used to describe service endpoints, operations, and data formats.

The construct that gets closest to a description of Ethereum smart contracts is the so-called “ABI in JSON” interface description produced by the Solidity compiler during compilation, as exemplified by the following lines of code:

```
[{
  “type”: “function”,
  “inputs”: [{“name”: “username”, “type”: “string”},
             {“name”: “password”, “type”: “string”}],
  “name”: “create_user”,
  “outputs”: [{“name”: “success”, “type”: “bool”}]
}, {
  “type”: “event”,
  “inputs”: [{“name”: “username”, “type”: “string”,
            “indexed”: true},
             {“name”: “count”, “type”: “uint256”,
             “indexed”: false}],
  “name”: “user_created”
}]
```

The description specifies one function (create_user) and one event (user_created), along with their inputs and outputs. The inputs of the event are their publicly accessible arguments stored in the blockchain; indexed arguments are searchable. What this description does not include is the name of the contract, its address, the network/chain ID if the contract is deployed on a test network, and non-functional properties (e.g., the cost of invoking the function). These are essential for search and discovery. Also, Ethereum does not come with a registry for smart contracts, although contract metadata (containing the ABI in JSON description) can be stored in Swarm, a redundant and decentralized store of Ethereum’s public record.

STATE OF TECHNOLOGY

In Table 2, we summarize how these SOA characteristics are manifest (or not) in the four platforms we introduced earlier.

As expected, Bitcoin is the most limited platform in terms of features supported when it comes to smart contracts. In fact, it was born as support for its homonymous cryptocurrency and less to support generic computations.

Table 2. SOC perspective on selected smart contract technologies.

	Bitcoin	Ethereum	Hyperledger Fabric	Corda
Contract type	Contracts, oracles	Contracts, libraries, data contracts, oracles	Contracts (chaincode), data contracts	Contracts, libraries, oracles
Interaction style	Pull interactions	Pull and push interactions, business protocols	Pull and push interactions	Pull and push interactions, business protocols
Interaction protocol	Transactions	Transactions, events, message calls, delegate calls	Transactions, calls (limited to contracts on same node and channel), events—exposes REST APIs toward these	Transactions, inter-node messages (so-called flows), scheduled invocations of contracts
Data format	Binary payloads in transactions	Binary payloads in transactions and events, Solidity data types in message/ delegate calls	Binary or JSON formatted key-value pairs	Any type of the contract language, zip attachments referenced using hashes
Description	No contract description	Contract metadata (JSON) to be published on a public storage platform (e.g., Swarm)	Chaincode metadata with interfaces, endpoints, and interaction schemas	No contract description

Ethereum, on the other hand, is the most complete platform, with Hyperledger Fabric and Corda providing comparable features.

In terms of contract types, all platforms support oracles, except Hyperledger Fabric for which so-called "gateway services" are still under discussion (as of June 2018). Reusable code libraries are supported only by Ethereum and Corda. It is important to note that contracts generally encapsulate application logic; data contracts are typically very limited in their storage capacity, as storing data on the blockchain may incur significant costs.

All platforms except Bitcoin support pull and push interactions; Bitcoin features only client-originated pull transactions. Looking at the interaction protocols, Ethereum, Hyperledger Fabric, and Corda support transactions, calls between contracts, and events; Bitcoin has only transactions.

Payload data is binary formatted in Bitcoin and Ethereum transactions and events, while Ethereum message/delegate calls pass native Solidity data structures. Hyperledger Fabric structures data as key-value pairs in binary and/or JSON format. Corda, in addition to generic Kotlin/Java data objects, also supports transactions with generic attachments; attachments are zipped and hash referenced.

As for the description of smart contracts for search and reuse, support is very limited. Only Ethereum and Hyperledger Fabric provide basic metadata describing a contract's interface (operations and arguments), but we are far from a common description format let alone a registry for the discovery of contracts.

DISCUSSION AND OUTLOOK

By now, there is a general consensus that the impact of blockchain will go far beyond cryptocurrencies, possibly with disruptive effects on distributed application development.[14] The key enabler for this impact is smart contracts able to support a new kind of distributed computing.[6]

While the number and types of platforms for smart contracts are constantly growing—this paper studies four of them, dozens of others have emerged—the resulting technological landscape is getting increasingly intricate and heterogeneous.

Yet, this paper shows that from an application point of view the conceptual underpinnings of this new landscape are more integrated than one would expect and that smart contracts, to some extent, may indeed be interpreted as elementary pieces, that is, services, of a blockchain-based, SOC paradigm. The paper, however, also shows that we are still far from a smart contract model that sees interoperability and reusability as beneficial features, as instead we are used to in the context of SOC.

In order to enable service orientation in blockchain and to unleash the full power of smart contracts, several challenges need to be faced, among which we mention:

- *Search, discovery and reuse*: It is striking that so little attention has been paid so far to enable developers to reuse already deployed contracts, especially if we consider that deploying a new contract is typically more cost-intensive then just invoking an already deployed one. Suitable abstract descriptors and searchable registries are badly needed.
- *Cost awareness*: Smart contracts natively incorporate the concept of resource consumption and cost of invocations. It is crucial that smart contracts be able to properly communicate and negotiate these kinds of service levels, enabling a natural pay-per-invocation model.
- *Performance*: Libraries and data contracts are executed locally inside each node and have thus negligible response times; oracles and generic contracts, which may require transaction processing, may lead to higher, unpredictable response times. The challenge is improving performance in terms of transaction rates and processing times.
- *Interoperability and standardization*: Today, platforms concentrate on own technologies as distinguishing feature, which is understandable. This, however, slows down integration, which eventually will nevertheless be needed. The challenge is agreeing on shared interaction styles and protocols as well as data formats and, of course, authentication and certification mechanisms. A particular challenge is cross-blockchain integration.
- *Composition*: Finally, in order to be able to exploit the full power of smart contracts (and to collectively save resources and money) it is necessary to conceive and implement composition solutions able to abstract away from technicalities and to provide developers with instruments and infrastructures that enhance productivity effectively.

In short, what we envision is an evolution from today's technology silos to an abstract, reuse-oriented contract ecosystem able to preserve the guarantees proper of blockchain technology.

REFERENCES

1. S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008, [Online]. Available: https://bitcoin.org/en/bitcoin-paper
2. G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," Ethereum Project Yellow Paper, vol. 151, pp. 1–32, 2014.
3. D. Mingxiao, M. Xiaofeng, Z. Zhe, W. Xiangwei, and C. Qijun, "A review on consensus algorithm of blockchain," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, 2017, pp. 2567–2572.
4. N. Szabo, "Smart contracts: Building blocks for digital markets," *EXTROPY: J. Transhumanist Thought*, 1996, Art. no. 16.
5. R. M. Parizi and A. Dehghantanha, "Smart contract programming languages on blockchains: An empirical evaluation of usability and security," in *Proc. Int. Conf. Blockchain*, 2018, pp. 75–91.
6. B. Dickson, "How blockchain can create the world's biggest supercomputer," TechCrunch, Dec. 2016.
7. X. Xu *et al.*, "The blockchain as a software connector," in *Proc. 13th Work. IEEE/IFIP Conf. Softw. Archit.*, 2016, pp. 182–191.
8. I. Weber, X. Xu, R. Riveret, G. Governatori, A. Ponomarev, and J. Mendling, "Untrusted business process monitoring and execution using blockchain," In *Proc. Int. Conf. Bus. Process Manage.*, 2016, pp. 329–347.
9. N. Atzei, M. Bartoletti, and T. Cimoli, "A survey of attacks on Ethereum smart contracts (sok)," in *Principles of Security and Trust*. Berlin, Germany: Springer, 2017, pp. 164–186.

10. I. Nikolić, A. Kolluri, I. Sergey, P. Saxena, and A. Hobor, “Finding the greedy, prodigal, and suicidal contracts at scale,” arXiv:1802.06038, 2018.
11. M. P. Singh and A. K. Chopra, “Violable contracts and governance for blockchain applications,” arXiv:1801.02672, 2018.
12. G. Alonso, F. Casati, H. Kuno, and V. Machiraju, *Web Services*. Berlin, Germany: Springer, 2004.
13. A. Lagares Lemos, F. Daniel, and B. Benatallah, “Web service composition: A survey of techniques and tools,” *ACM Comput. Surveys*, vol. 48, no. 3, Feb. 2016, Art. no. 33.
14. L. Mearian, “What is blockchain? The most disruptive tech in decades.” IDG Commun., Inc., Framingham, MA, USA, May 2018.

Florian Daniel is an Associate Professor with Politecnico di Milano, Milan, Italy. His research interests include service-oriented computing, blockchain, business process management, and data science. He received the Ph.D. degree in information technology from Politecnico di Milano. Contact him at florian.daniel@polimi.it.

Luca Guida is a graduate of Politecnico di Milano, Milan, Italy. His research interests include blockchain, service-oriented computing, and spatial data analysis. He received the master’s degree (*cum laude*) in computer science and engineering from Politecnico di Milano and Alta Scuola Politecnica, Milan, Italy. Contact him at luca.guida@mail.polimi.it.

This article originally appeared in
IEEE Internet Computing, *vol. 23, no. 1, 2019.*

ADVERTISER INFORMATION

Advertising Coordinator

Debbie Sims
Email: dsims@computer.org
Phone: +1 714-816-2138 | Fax: +1 714-821-4010

Advertising Sales Contacts

Mid-Atlantic US:
Dawn Scoda
Email: dscoda@computer.org
Phone: +1 732-772-0160
Cell: +1 732-685-6068 | Fax: +1 732-772-0164

Southwest US, California:
Mike Hughes
Email: mikehughes@computer.org
Cell: +1 805-208-5882

Northeast, Europe, the Middle East and Africa:
David Schissler
Email: d.schissler@computer.org
Phone: +1 508-394-4026

Central US, Northwest US, Southeast US, Asia/Pacific:
Eric Kincaid
Email: e.kincaid@computer.org
Phone: +1 214-553-8513 | Fax: +1 888-886-8599
Cell: +1 214-673-3742

Midwest US:
Dave Jones
Email: djones@computer.org
Phone: +1 708-442-5633 Fax: +1 888-886-8599
Cell: +1 708-624-9901

Jobs Board (West Coast and Asia), Classified Line Ads

Heather Bounadies
Email: hbuonadies@computer.org
Phone: +1 623-233-6575

Jobs Board (East Coast and Europe), SE Radio Podcast

Marie Thompson
Email: marie.thompson@computer.org
Phone: +1 714-813-5094

COMPSAC 2020

Madrid, Spain
July 13-17, 2020

COMPSAC is the IEEE Computer Society Signature Conference on Computers, Software and Applications. It is a major international forum for academia, industry, and government to discuss research results and advancements, emerging challenges, and future trends in computer and software technologies and applications. The theme of COMPSAC 2020 is "Driving Intelligent Transformation of the Digital World".

Staying relevant in a constantly evolving digital landscape is a challenge faced by researchers, developers, and producers in virtually every industry and area of study. Once limited to software-enabled devices, the ubiquity of digitally-enabled systems makes this challenge a universal issue. Furthermore, as relevance fuels change, many influencers will offer solutions that benefit their own priorities. Fortunately, history has shown that the building blocks of digital change are forged by those conducting foundational research and development of digital systems and human interactions. Artificial Intelligence is not new, but is much more utilized in everyday computing now that data and processing resources are more economically viable, hence widely available. The opportunity to drive the use of this powerful tool in transforming the digital world is yours. Will your results help define the path ahead, or will you relegate those decisions to those with different priorities for utilizing intelligence in digital systems? COMPSAC has been and continues to be a highly respected venue for the dissemination of key research on computer and software systems and applications, and has influenced fundamental developments in these fields for over 40 years. COMPSAC 2020 is your opportunity to add your mark to this ongoing journey, and we highly encourage your submission!

COMPSAC 2020, organized as a tightly integrated union of symposia, will focus on technical aspects of issues relevant to intelligent transformation of the digital world. The technical program will include keynote addresses, research papers, industrial case studies, fast abstracts, a doctoral symposium, poster sessions, and workshops and tutorials on emerging and important topics related to the conference theme. Highlights of the conference will include plenary and specialized panels that will address the technical challenges facing researchers and practitioners who are driving fundamental changes in intelligent systems and applications. Panels will also address cultural and societal challenges for a society whose members must continue to learn to live, work, and play in the environments the technologies produce.

Authors are invited to submit original, unpublished research work, as well as industrial practice reports. Simultaneous submission to other publication venues is not permitted except as highlighted in the COMPSAC 2020 J1C2 & C1J2 program. All submissions must adhere to IEEE Publishing Policies, and will be vetted through the IEEE CrossCheck portal. Further info is available at www.compsac.org.

Organizers
Standing Committee Chair: Sorel Reisman (California State University, USA)
Steering Committee Chair: Sheikh Iqbal Ahamed (Marquette University, USA)
General Chairs: Mohammad Zulkernine (Queen's University, Canada), Edmundo Tovar (Universidad Politécnica de Madrid, Spain), Hironori Kasahara (Waseda University, Japan)
Program Chairs in Chief: W. K. Chan (City University, Hong Kong), Bill Claycomb (Carnegie Mellon University, USA), Hiroki Takakura (National Institute of Informatics, Japan)
Workshop Chairs: Ji-Jiang Yang (Tsinghua University, USA), Yuuichi Teranishi (National Institute of Information and Communications Technology, Japan), Dave Towey (University of Nottingham Ningbo China, China), Sergio Segura (University of Seville, Spain)
Local Chairs: Sergio Martin (UNED, Spain), Manuel Castro (UNED, Spain)

Important Dates
Workshops proposals due: 15 November 2019
Workshops acceptance notification: 15 December 2019
Main conference papers due: 20 January 2020
Paper notification: 3 April 2020
Workshop papers due: 9 April 2020
Workshop paper notifications: 1 May 2020
Camera-ready and registration due: 15 May 2020

IEEE COMPUTER SOCIETY

Photo: King Felipe III in Major Square, Madrid
Photo credit: Iria Castro - Photographer (Instagram @iriacastrophoto)

EDITOR JEFFREY VOAS
NIST; j.voas@ieee.org

Cryptocurrencies: Transparency Versus Privacy

Nir Kshetri, University of North Carolina at Greensboro

Cryptocurrencies can have significant privacy costs. A motivated adversary has available a range of actions to identify the actual user associated with a cryptocurrency account. By taking appropriate measures, cryptocurrency users can minimize privacy violations and reduce the risk of privacy breaches.

Transparency is a major factor that is driving the use of blockchain-based applications such as cryptocurrencies. A major question becomes whether transparency provides reasonable privacy protection. For instance, many firms in the financial sector do not like the fact that blockchain's transparent nature gives other users access to the details of conducted transactions.

Let's begin with cryptocurrencies. It is important to note that cryptocurrencies possess built-in mechanisms that provide reasonable levels of privacy to users. To make the costs of transparency less severe to privacy, Bitcoin and other cryptocurrencies employ pseudonymity. Users can conduct transactions with one another without disclosing any information related to their identity.

Concealing the Internet Protocol (IP) addresses of users is another mechanism that provides protection to cryptocurrency user privacy. For example, in the Bitcoin network, correspondence cannot be established between transactions and IP addresses. Bitcoin users are connected to a peer-to-peer (P2P) network. Data continue to flow among the devices connected to the P2P network until everyone has the information related to a transaction. No one, except for the originator, knows who initiated the transaction.[1]

CONSEQUENCES OF PRIVACY VIOLATIONS IN THE CRYPTO-WORLD

Individuals and organizations are likely to suffer more severe consequences from cases of privacy violation if they engage in illegal behaviors using cryptocurrencies (compared with other transaction models). For example, if someone is caught in a crime, the cryptocurrency account can be linked

Digital Object Identifier 10.1109/MC.2018.2876182
Date of publication: 15 January 2019

to any crime committed by that person in the past. Privacy breaches are likely to lead to more severe criminal consequences, referred as an *amplified technical impact*.[2]

Zcash transactions have two types of addresses: transparent and shielded.

Privacy is important for citizens and businesses. If an individual uses Bitcoin to pay for certain goods or services, the party with whom the transaction is being made can know exactly how much money the individual has. This may increase the threat to personal safety. A supplier that has received a payment from a business would know how much money the business has. Knowing fund availability and customer price sensitivity could affect future negotiations. Finally, if online businesses have information about a consumer's spending patterns, they could predict the highest price that the consumer could pay. The business could then use price tampering to increase profits.[3]

UNWANTED PERSONAL INFORMATION LEAKS

There are various sources of unwanted personal information leaks in transactions involving cryptocurrencies. For instance, while Bitcoin transactions are difficult to track, they are not completely anonymous. All transactions are recorded in a permanent public ledger. After the Bitcoins are moved from that address, financial movements can be traced. Users can be traced through IP addresses and money flows. A team of researchers studied 130 major merchants that allow Bitcoin transactions. They found that at least 53 of the merchants leaked payment information to at least 40 third parties. While most of the information leaked was intentional and used for advertising and analytics, some merchant websites also leaked precise blockchain transaction information to trackers.[4]

Blockchain ledgers are searchable and, hence, can be used to track transactions.[5] If a leak involves the amount and time of the purchase, a motivated adversary can convert the purchase amount into Bitcoins using the exchange rate at that time. Then, a blockchain can be searched for a transaction of that amount and at that time. This gives away the user's Bitcoin address. Any other purchases made using that address are now easier to trace.[4]

Sometimes, an act of carelessness on the part of the user may decrease privacy. This happened to Ross Ulbricht, who created the online black market Silk Road, best known as a platform for selling illegal drugs. When Ulbricht looked for help to expand the Silk Road business, he used the same pseudonym that he had adopted previously to post announcements on illegal drug discussion forums. This made him an FBI suspect. The FBI tracked his IP address to an Internet café in San Francisco and caught Ulbricht as he was logging in to Silk Road as an administrator.[1]

Another privacy problem occurs when users of cryptocurrencies such as Bitcoin reuse addresses. By doing so, they publicly disclose information about past financial transactions, and this can compromise their privacy. The transparency and immutability features of cryptocurrencies like Bitcoin make it possible to track every transaction involving a given address. Even if a person has engaged in careful processes to hide his or her identity, once a link has been established between a person's identity and a Bitcoin address, all past transactions made by the owner of the Bitcoin address will be associated to the owner's identity.

CRYPTOCURRENCIES HAVE DIFFERENT LEVELS OF PRIVACY

Well-known cryptocurrencies such as Bitcoin have not been able to meet all privacy needs of users. As mentioned, financial firms are concerned that blockchain's ledger allows other users to access the details of transactions already conducted. In response to these demands, some cryptocurrencies provide users with higher levels of privacy protection.

Blockchain is still in early-stage development, and various alternative models and forms of cryptocurrencies are evolving along with it. For instance, to make blockchain more appealing to financial institutions, the cryptocurrency Zcash, which was launched in October 2016, has promised transactional privacy.[6] It employs cryptography to enhance user privacy.

Zcash transactions can be made transparent, like those of Bitcoin, or shielded through a zero-knowledge proof. Zcash transactions have two types of addresses: transparent and shielded. In transparent addresses, as is the case for Bitcoin, the monetary amount of the transaction as well as information about the receiver and the sender appears in the blockchain. On the other hand, if a shielded address is used, the address is "obscured" on the public ledger. Also, if both the sender and the receiver use shielded addresses, the transaction amount is encrypted.

Users of shielded addresses constitute a small proportion of Zcash adopters. In early 2017, shielded addresses accounted for about 0.8% of Zcash transactions.[7] That proportion is predicted to increase to 4% by mid-2018.[8]

A relatively low adoption rate of shielded addresses might be due to the additional time and computational resources required. Shielded addresses require a more computationally intensive process. To use Zcash's privacy features, users may need 4 GB or more of RAM (tinyurl.com/y9dtj3dh). With 4 GB of RAM, operations were reported to take as long as 2 min to complete.[9] Therefore, most exchanges and wallets support only transparent Zcash transactions.[8]

Likewise, Monero focuses on privacy and untraceability by hiding the transaction's sender, receiver, and monetary amount. To achieve this, Monero mixes Monero "coins" with other forms of payments. This makes it nearly impossible to link a transaction to any particular identity or previous transaction from the same source if only Monero's blockchain is searched.[10]

Despite higher levels of user privacy from Monero and Zcash, these cryptocurrencies have not yet achieved higher popularity. For instance, as of mid-July 2018, market capitalization of Monero and Zcash was about $2 billion and $816 million, respectively, compared with Bitcoin's $115 billion and Ethereum's $48 billion (coinmarketcap.com/).

REGULATORY AND LAW ENFORCEMENT RESPONSES

Regulatory and law enforcement agencies are now focusing on illegal activities associated with cryptocurrencies. Law enforcement agencies are concerned with the anonymity features of cryptocurrencies. At a congressional hearing, former assistant US attorney Kathryn Haun noted that, when regulators issue subpoenas requesting documents relating individual identities to illicit activities at cryptocurrency exchanges, subpoenas may return information such as "Mickey Mouse" living at "123 Main Street" (tinyurl.com/y8g2x23c).

Academic researchers and blockchain intelligence companies are using advances in computer science, economics, and forensics to help law enforcement. Law enforcement agencies now have access to advanced techniques to track illegal activities that employ cryptocurrencies. Elliptic, a blockchain intelligence company, uses artificial intelligence to scan and analyze the Bitcoin network to identify suspicious transactions. It can trace transactions to individuals and groups. Elliptic's services are used by online exchanges and law enforcement to detect money laundering (bit.ly/1T3SBwc).

The higher levels of privacy offered by cryptocurrencies such as Monero and Zcash concern regulators who are focused on money laundering. A cybercrime expert at the European Union's law enforcement agency, Europol, noted that criminals have begun shifting away from Bitcoin to cryptocurrencies with higher levels of privacy (tinyurl.com/yat9hucw). In recent years, regulators have increased their focus on cryptocurrencies with higher degrees of nontraceability. In June 2018, in testimony before the House of Representatives Committee on Financial Services Subcommittee on Terrorism and Illicit Finance, an official of the US Secret Service recommended better regulation of less traceable cryptocurrencies to prevent illegal activities from benefiting from nontraceable coins (tinyurl.com/ycot283t).

Cryptocurrencies' transparency and immutability features come with a privacy cost. Adversaries can use a range of actions to identify the actual user associated with a specific cryptocurrency account.

It is important for cryptocurrency users to be aware that their privacy can be compromised. Users need to take precautions to minimize privacy violations and mitigate the risk of privacy breaches. Users should refrain from reusing identities in both their noncryptocurrency and cryptocurrency worlds. Likewise, by reusing cryptocurrency addresses, users are more likely to publicly disclose personal information. Higher levels of privacy require generating a new address for each transaction.

ACKNOWLEDGMENTS

I thank Jeff Voas for numerous edits and suggestions on previous versions of this article.

REFERENCES

1. J. Bohannon. (2016, Mar. 9). Why criminals can't hide behind Bitcoin. Science. [Online]. Available: http://www.sciencemag.org/news/2016/03/why-criminals-cant-hide-behind-bitcoin
2. ISACA, Generating value from big data analytics, ISACA, Schaumburg, IL, White Paper, 2014. Available: http://www.isaca.org/Knowledge-Center/Research/ResearchDeliverables/Pages/Generating-Value-From-Big-Data-Analytics.aspx
3. draglet.com. (2018). What is Monero? Everything you need to know. *Draglet*. [Online]. Available: https://www.draglet.com/what-is-monero/
4. technologyreview.com. (2017, Aug. 23). Bitcoin transactions aren't as anonymous as everyone hoped. *Technology Review*. [Online]. Available: https://www.technologyreview.com/s/608716/bitcoin-transactions-arent-as-anonymous-as-everyone-hoped/
5. E. Aldaz-Carroll and E. Aldaz-Carroll. (2018, Feb. 1). Can cryptocurrencies and blockchain help fight corruption? *Brookings*. [Online]. Available: https://www.brookings.edu/blog/future-development/2018/02/01/can-cryptocurrencies-and-blockchain-help-fight-corruption/
6. L. Clozel. (2016). How Zcash tries to balance privacy, transparency in Blockchain. *American Banker*. [Online]. Available: http://www.americanbanker.com/news/law-regulation/how-zcash-tries-to-balance-privacy-transparency-in-blockchain-1092198-1.html
7. A. Hertig. (2017, Jan. 13). Hardly anyone seems to be using Zcash's anonymity features. *Coin Desk*. [Online]. Available: https://www.coindesk.com/hardly-anyone-is-using-zcashs-anonymity-features-but-we-couldnt-tell-if-they-were/
8. B. Penny. (2018, May 3). What is ZEC? Introduction to Zcash: Blockchains

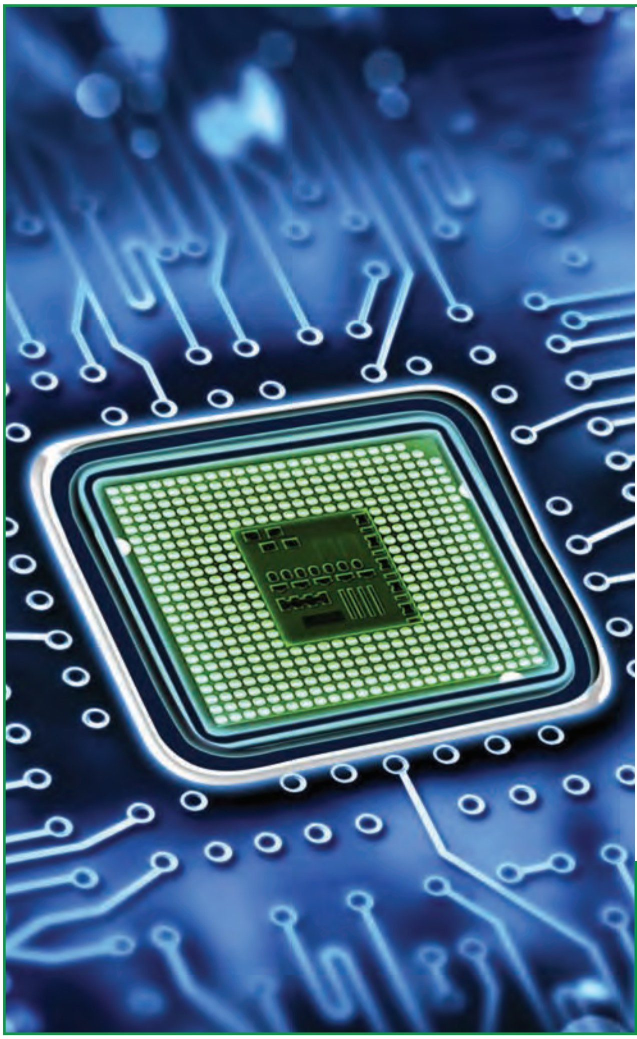
can cause zzzzz, but pure currency cryptos can really push boundaries. *Crypto Briefing*. [Online]. Available: https://cryptobriefing.com/what-is-zec-introduction-to-zcash/

9. P. Peterson. (2016, Oct. 19). User expectations at Sprout Pt. 2: Software usability and hardware requirements. *Zcash*. [Online]. Available: https://blog.z.cash/software-usability-and-hardware-requirements/

10. A. Greenberg. (2018, Mar. 27). The dark web's favorite currency is less untraceable than it seems. *Wired*. [Online]. Available: https://www.wired.com/story/monero-privacy/

NIR KSHETRI is a professor of management in the Bryan School of Business and Economics at the University of North Carolina at Greensboro. Contact him at nbkshetr@uncg.edu.

This article originally appeared in Computer, *vol. 51, no. 11, 2018.*



COLUMN: LIFE IN THE C-SUITE

Skills and Competencies for Digital Transformation

Stephen J. Andriole
Villanova University School of Business

Editor:
Stephen J. Andriole,
Villanova University School of Business;
steve@andriole.com

Digital transformation requires a special set of skills and competencies, such as BPM, robotic process automation, cloud computing, emerging technology, agile program management, cybersecurity, and effective internal and external communications skills.

CIOs, CTOs, CMOs, COOs, CFOs, and CEOs need teams with the right skills and competencies, especially ones that enable digital transformation. At the very least, teams need skills and competencies in the following areas.

BUSINESS ANALYSIS MODELING, SIMULATION, AND AUTOMATION

The requirement here includes knowledge of business process modeling/management (BPM), robotic process automation (RPA), requirements identification, modeling and validation, and, of course, digital transformation itself. It also assumes the ability to model, simulate and measure existing and future business processes and whole new business models. This area also assumes knowledge of, and experience with, requirements matching with external vendor capabilities and specific transformation programs and projects.

EMERGING TECHNOLOGIES, ESPECIALLY DISRUPTIVE TECHNOLOGIES

This requirement includes knowledge of emerging technologies that might—and should—disrupt the business rules, processes, and models of specific vertical industries and companies. The requirement also assumes competency in competitive technology intelligence. It assumes wide and deep knowledge of, and experience with, the adoption of disruptive technology. Of special importance are emerging/disruptive technologies like virtual/augmented reality, automated reasoning, distributed ledger technology, cashless payment systems, real-time statistical/augmented analytics, simulation/gaming technology, location-based technology, and disruptive interface technologies like intelligent speech and facial recognition, among others.

2469-7087/20 © 2020 IEEE

STATISTICAL AND AUGMENTED ANALYTICS

The requirement here includes knowledge of structured and unstructured descriptive, explanatory, and predictive analytics. It also includes knowledge of, and experience with, the major open source analytics platforms like Hadoop and Spark, among others. It focuses on data science, data representation, deep learning, simulation, and displays. The requirement also includes knowledge of augmented analytics, which, according to the Gartner Group, is "an approach that automates insights using machine learning and natural-language generation, (and) marks the next wave of disruption in the data and analytics market."[1]

CLOUD COMPUTING

The requirement here includes knowledge of all flavors of cloud delivery, including all of the service models that cloud computing provides, such as infrastructure (IaaS), software (SaaS), platform (PaaS), security (SaaS), mobility (MaaS), analytics (AaaS), blockchain (BaaS), and even learning (LaaS). It is also about knowledge of, and experience with, alternative cloud delivery architectures, cloud service level agreements (SLAs), cloud performance metrics, and cloud virtualization (especially container) technologies.

PERFORMANCE METRICS

The requirement here is on operational, delivery, organization, and financial metrics, including metrics around online cloud application performance, cloud application availability, delivery incidents, SLA adherence, project performance (especially satisfaction), personnel performance reviews, budgeting, and resource costs. Knowledge and experience here also refer to the tools available to track, measure, and report technology performance metrics.

REMOTE, AGILE PROJECT, AND PROGRAM MANAGEMENT

This requirement includes knowledge of project and program management tools, techniques, and best practices. It assumes knowledge of, and experience with, project and program management of small and large-scale technology projects and familiarity with the array of tools available to the professional project and program managers. This assumes the ability to manage projects and programs cost-effectively, within task-defined timelines, and remotely. It also assumes agility.

COMPETITIVE VENDOR MANAGEMENT

This requirement includes knowledge of technology vendor management best (and worst) practices. This assumes knowledge of, and experience with, the development of requests for information, requests for proposals, and requests for quotes, including automated tools to develop and compare these documents. This also assumes the development of detailed SLAs and the management tools for measuring SLA compliance and performance. Communications and negotiation skills are also part of this skills/competencies area.

DIGITAL SECURITY AND SECURITY MANAGEMENT

The requirement here includes knowledge of the variety of current and emerging security technologies including blockchain technology, multifactor authentication, application isolation, intelligent/automated security tools, mobile application wrapper technology, detection technologies, IaaS/SaaS device security technologies, automated testing, and pervasive/IoT security technologies, among others. The security requirement also includes knowledge of security challenges and processes, including security policies and the adoption of best practices, compliance with industry standards (such as ISO27002), regulatory compliance (such as with

GDPR), vulnerability assessment/remediation, penetration testing, incident response, network and systems monitoring, forensic analysis, security awareness and training, backup, and recovery, among others. The focus should be on audit-approved security-as-a-service, not on in-house security delivery core competencies.

INTERNAL AND EXTERNAL COMMUNICATIONS SKILLS

This requirement includes experience writing reports and creating presentations that are easily understood and therefore actionable. The key to communication is purposeful brevity: Is the team capable of such (written and oral) communication? Communications should also be customized to specific audiences, such as executives, boards, internal auditors, sales and marketing professionals, and customers, among others. Making presentations that are easily understood and therefore actionable is an essential skill. Verbal communications should also be customized to specific audiences, such as executives, boards, internal auditors, sales and marketing professionals, and customers, among others. The requirement also includes experience presenting to outside constituencies and stakeholders, especially vendors, external auditors, customers, and professional organizations. Senior members of the technology team must be "presentable" to a wide external audience. As the company's business-technology representatives—and as one of the principal spokespersons for digital transformation—the senior team (especially the CIO, CTO, CMO, COO, CFO, and CEO) must all be superb presenters.

FILLING THE GAPS

These skills and competencies should be used to assess digital transformation capabilities, which involves an objective workforce assessment of the business-technology team. If gaps exist—as they likely will—CIOs, CTOs, CMOs, COOs, CFOs, and CEOs must react accordingly. Digital transformation is complicated yet potentially extremely impactful, especially when transformation leverages emerging and disruptive technology, but skills and competencies gaps must be addressed.

> If a company is not a disruptor, it is disruptable. Digital transformation thus becomes a survival tactic and a long-term strategy.

CIOs, CTOs, CMOs, COOs, CFOs, and CEOs have three options: repair, rent, or replace. The repair option is often a good one: retrain and retool the willing keepers. Rethink how many full-time permanent technology professionals are necessary: rent the others as consultants, contractors, and long-term vendors. Unfortunately, companies may also have to replace some members of the business-technology team. While this is always difficult, unsalvageable talent threatens competitiveness.

Digital transformation is challenging—and continuously necessary. This is not the first time we have heeded the call to "re-engineer," and it will not be the last. "Digital transformation" is today's unique call to action. It is unique today because of the trajectory of digital technology and the impact that current, emerging, and disruptive technology has had on business processes and whole new business models. Industries and companies now live in fear of disruption because of what is happened to the travel, delivery, transportation, insurance, and retail industries. The real estate, banking, and election industries are next—and with a vengeance.

Said differently, if a company is not a disruptor, it is disruptable. Digital transformation thus becomes a survival tactic and a long-term strategy. CIOs, CTOs, CMOs, COOs, CFOs, and CEOs have little choice. They must identify the skills and competencies necessary to remain competitive. The list will change over time, and sometimes very quickly—disruptively. C-Suiters must educate, re-educate, train, retrain, replace, and rent the necessary skills and competencies quickly, effectively, and continuously to assure competitiveness through digital transformation.

REFERENCE

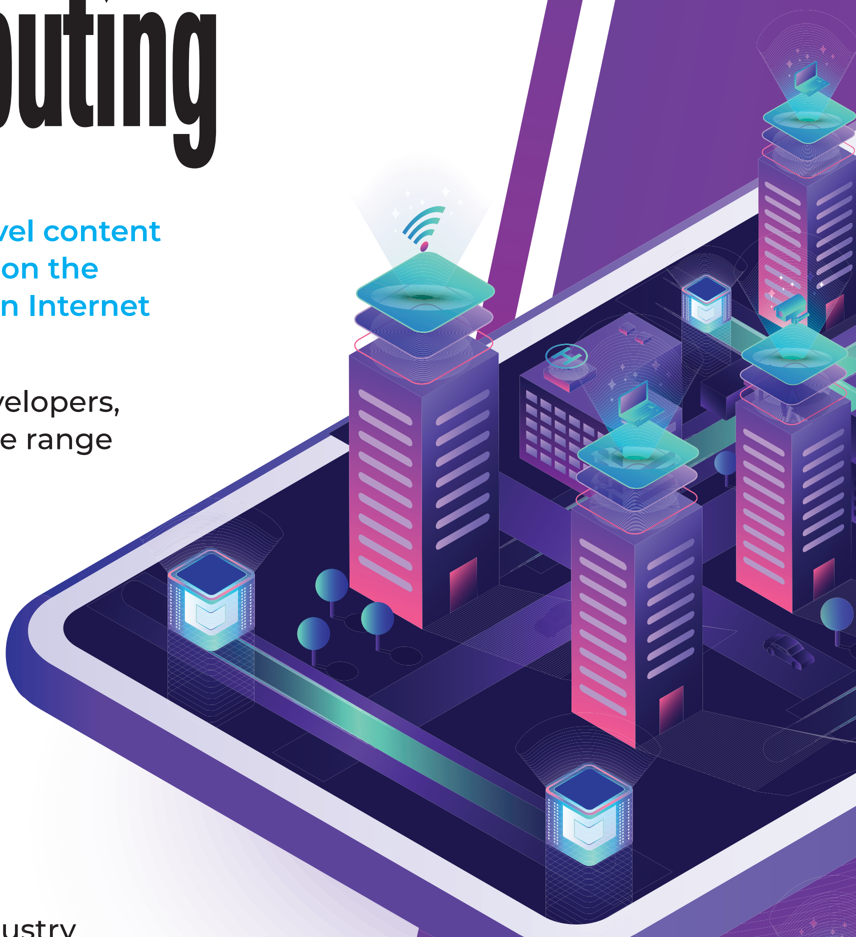
1. “Augmented analytics is the future of data and analytics,” *Gartner*, Jul. 2017. Available at: https://www.gartner.com/doc/3773164/augmented-analytics-future-data-analytics

ABOUT THE AUTHOR

Stephen J. Andriole is the Thomas G. Labrecque Professor of Business Technology with the Villanova School of Business, Villanova University, where he teaches courses in strategic technology and innovation and entrepreneurialism. Contact him at steve@andriole.com.

This article originally appeared in IT Professional, *vol. 20, no. 6, 2018.*

Rejuvenating Binary Executables ▪ Visual Privacy Protection ▪ Communications Jamming
IEEE SECURITY & PRIVACY
Building Dependability, Reliability, and Trust
SOFTWARE EVERYWHERE
January/February 2016 Vol. 14, No. 1
Policing Privacy ▪ Dynamic Cloud Certification ▪ Security for High-Risk Users
IEEE SECURITY & PRIVACY
IEEE Symposium on Security and Privacy
March/April 2016 Vol. 14, No. 2
Smart TVs ▪ Code Obfuscation ▪ The Future of Trust
IEEE SECURITY & PRIVACY
ECONOMICS OF CYBERSECURITY PART 2
May/June 2016 Vol. 14, No. 3
S & P
IEEE
IEEE computer society CELEBRATING 70 YEARS
Reliability Society



IEEE Internet Computing
VOLUME 22, NUMBER 2
MARCH/APRIL 2018
IEEE Internet Computing
IoT-Enhanced Human Experience
IEEE
IEEE Internet Computing
VOLUME 22, NUMBER 3
MAY/JUNE 2018
IEEE Internet Computing
JULY/AUGUST 2018
tion of Rack-Scale Systems

REQUIREMENTS

Editor: **Sarah Gregory**
Intel Corporation
sarah.c.gregory@ieee.org

Ubiquitous Requirements Engineering

A Paradigm Shift That Affects Everyone

Karina Villela, Eduard C. Groen, and Joerg Doerr

IN RECENT YEARS, we have witnessed profound changes in business and society. The use of digital technologies has brought about disruptive changes in every domain, changes that are widely known as the "digital transformation." Systems are growing increasingly interconnected and complex with cyberphysical systems even sensing and actuating in the physical world. Typical computer, tablet, and smartphone users include anyone from children to the elderly. A single software product can now easily reach audiences of millions with unprecedented opportunities to obtain feedback.

The techniques that have so far proven crucial for eliciting requirements do not hold up to the paradigm shifts that have taken place. Consequently, we argue that requirements engineering (RE) will have to evolve in several dimensions and thereby become ubiquitous.[1] Our view, as shown in Figure 1, consists of six dimensions of ubiquity in RE. For each dimension, we have identified the transformation (colored rectangle) required to overcome the critical barrier posed by the status quo (gray rectangle) for the way RE is performed in the digital transformation era.

You have probably noticed a shift in how your company is doing business. Some companies experience stronger dependency on other companies or have the actual need to cocreate an ecosystem with other companies. You may also have found that your product's end users have changed or that they have changed the way in which they communicate with you or others about your product. Perhaps you have worked on projects in which you needed to elicit requirements from stakeholders in unusual ways. Any of these observations may be an indication that your business is in need of ubiquitous RE.

In this department, we discuss four of the six transformations toward ubiquitous RE, combining "open RE" and "cross-domain RE" due to their strong synergy. These transformations have a great impact in industry and may have imminent implications for your work practice. We will begin each section by describing what has changed in the world and how RE needs to adapt. Then we will paint a picture of how RE could function from the perspective of a requirements engineer before we discuss the hurdles that still need to be overcome.

Cross-Domain and Open RE to Shape Software Ecosystems

In all domains, we see a rapidly growing demand by companies to form partnerships with other companies in software ecosystems to offer innovative digital solutions and thereby expand their business.[2] Through orchestrated cooperation, partners from different business

Digital Object Identifier 10.1109/MS.2018.2883876
Date of publication: 22 February 2019

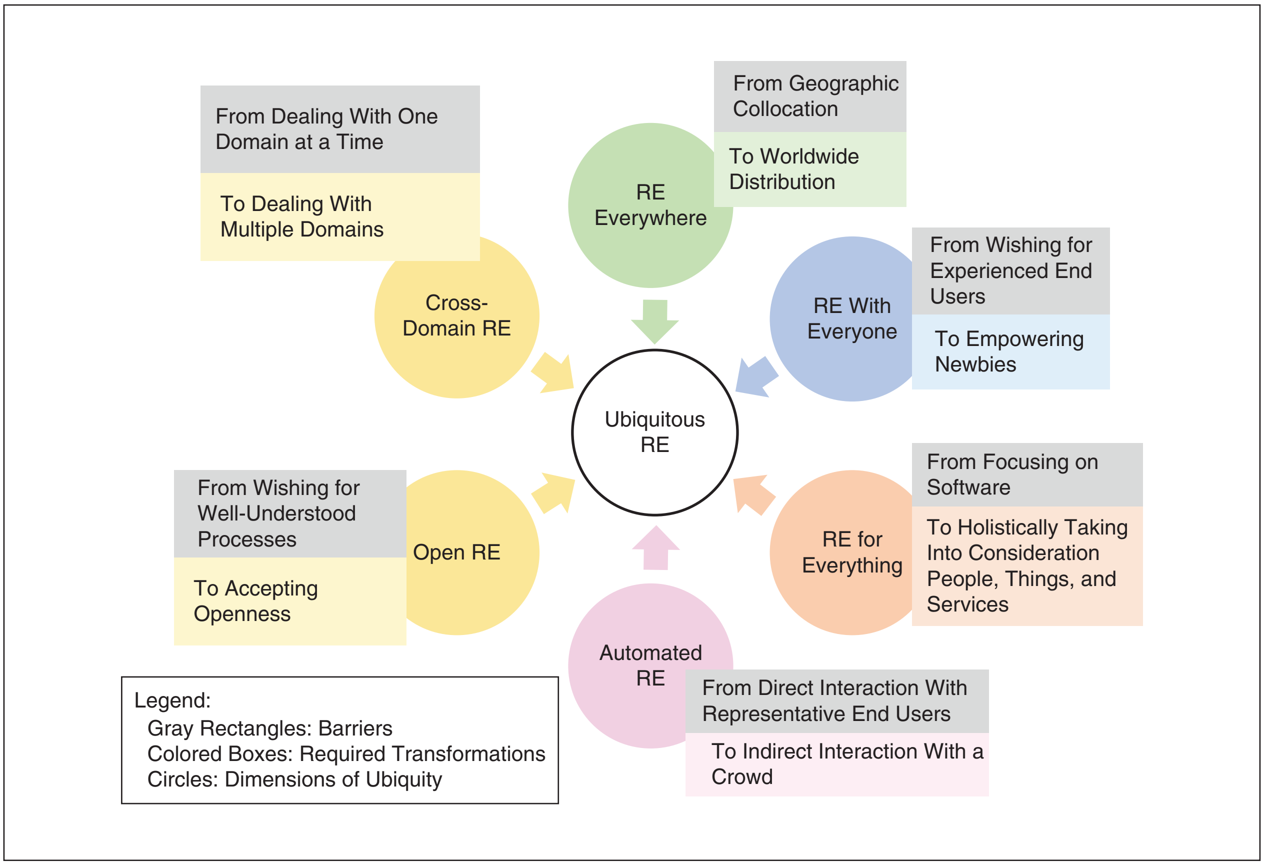

From Dealing With One Domain at a Time
To Dealing With Multiple Domains
Cross-Domain RE
From Geographic Collocation
To Worldwide Distribution
RE Everywhere
From Wishing for Experienced End Users
To Empowering Newbies
RE With Everyone
Ubiquitous RE
From Wishing for Well-Understood Processes
To Accepting Openness
Open RE
From Focusing on Software
To Holistically Taking Into Consideration People, Things, and Services
RE for Everything
Automated RE
From Direct Interaction With Representative End Users
To Indirect Interaction With a Crowd
Legend:
Gray Rectangles: Barriers
Colored Boxes: Required Transformations
Circles: Dimensions of Ubiquity

FIGURE 1. The six dimensions of RE ubiquity.[1]

sectors and different domains can provide high-level services that go far beyond their current and individual offerings. New business models and processes arise in scenarios where business and technical solutions influence each other and therefore must be shaped at the same time. A good example is the agricultural domain, which is being influenced by technologies based on the Internet of Things and big data and where the interplay of farming equipment manufacturers, chemical industry, insurance companies, and farm management providers creates new cross-domain software ecosystems. For planned and even for existing software ecosystems, several partners might still be unknown, or at least their contributions to the ecosystem may still be unclear.

Requirements engineers have used glossaries, domain concepts, and domain-relevant processes to familiarize themselves with new business domains and facilitate a shared understanding among project stakeholders. Some have been switching among domains rather than specializing in one domain or subdomain. These practices can help with the shaping of cross-domain ecosystems, but they do not suffice; requirements engineers must be capable of fostering connections among several business domains. In a similar way, adopting an incremental life cycle or an agile development approach can help but does not suffice when dealing with the inherent openness of software ecosystems. Requirement engineers must be capable of fostering the simultaneous shaping of business and software and be able to deal with uncertainty. The skills of requirements engineers need to shift from being able to elicit and represent knowledge and requirements obtained from domain experts to being able to connect businesses and propose requirements to domain experts. In this sense, a requirement engineer acts instead as a business transformer.[3]

Cross-domain and open RE starts with the identification of key ecosystem partners, which are concrete organizations interested in being part of the software ecosystem. A requirements engineer is part of an ecosystem leadership team, which also includes an integration architect, an experience designer, and others. This team seeks to make the design of the software ecosystem as tangible as possible. As a business transformer, the requirements engineer plans workshops where the key partners play with alternative physical representations of the ecosystem to find out how business flows can take better advantage of the assets of the ecosystem partners and ensure benefits to the overall ecosystem. Based on their knowledge of all involved domains, the requirements engineers also make assumptions and invent requirements, as the domain experts do not know yet how to transform their business models into an innovative software ecosystem. Due to the inherent complexity, requirements engineers always select a small subset of the ecosystem as the scope and test their assumptions and proposed requirements in short feedback cycles, either at the conceptual level or by using simulation or prototypes. During the whole process, they align the key ecosystem partners in several dimensions: social, business, technical, and legal.

Currently, only a few methods and tools support the shaping of a software ecosystem (see Villela et al.,[1] Section VI.B, for a review). To provide some guidance to requirements engineers, we are designing a framework of decisions that needs to be made to shape a planned ecosystem together with a workflow of activities that indicate the time for making those decisions.[4] However, we additionally see the need for 1) techniques that support the ideation of the ecosystem business and the performance of quick validation rounds with key ecosystem partners, 2) models and visualizations to provide different ecosystem views on different levels and from different perspectives, and 3) means to support a continuous change process based on runtime monitoring of emergent behavior.

Automated RE to Exploit User Feedback

Software is a commodity for virtually everyone. There is hardly any business area that is devoid of any software support whatsoever. On the flip side of software having become this widespread in both business-to-consumer and business-to-business settings, it has become hard to involve the enormous pool of stakeholders, let alone elicit requirements from a representative subset to build software that meets all users' expectations and needs. This is especially true for a heterogeneous user base, whose requirements are likely to be even more divergent. Moreover, companies have to deal with increasingly diverse, complex, and large software systems, while the demand for fast innovation calls for short feedback loops.

Traditional requirements elicitation techniques, such as interviews or focus groups, have scalability problems. They stretch the limitation of resources when performed with more than a few dozen people and if they need to be performed continuously to keep up with the competition. Besides, they are typically best suited for collocated settings. Approaches for dealing with large crowds of users typically make RE scalable by using new communication mechanisms and (big data) analytics. We introduced the paradigm of crowd-based requirements engineering (CrowdRE), which involves automated gathering and analysis of user feedback, as well as the use of motivational techniques to boost the generation of user feedback.[5] Together, these approaches address typical problems experienced in RE, including engaging a high number of stakeholders, prioritizing requirements reliably, and refining coarse-grained requirements.

When automation is employed in RE, the role of the requirements engineer resembles that of a data analyst. No direct interaction with end users takes place to elicit requirements. Rather, the requirements engineer gets to see the results of automated analyses conducted over user feedback, which should produce information about requirements, and can then make decisions accordingly. A company can obtain such data from all their communication channels (e.g., social media, review sites, bug trackers, and customer relationship management systems) and from the software product itself (e.g., log data, built-in feedback mechanisms). Because such user feedback has been shown to be a fruitful source of opinions and requirements, text mining and usage mining approaches automatically extract requirements and relevant information from such data. To improve the interpretability and validity of the results, requirements engineers could employ crowdsourcing techniques to manually assess user feedback (e.g., rating or annotating sentences or validating analysis results).

CrowdRE is gaining traction; practitioners are interested in the topic, and the body of research on automated user feedback is growing. However, mining techniques and classification algorithms have only been adapted to RE recently and need to be further refined to provide reliable results without requiring much additional manual work. What makes automatic analyses especially difficult is the inherent ambiguity of unstructured user feedback. Moreover, companies still neglect most of the communication channels they have in use, while research has focused on public communication channels. Thus, there is potential to assess a greater spectrum of feedback channels, including feedback about competitor products. CrowdRE's ultimate goal is not only to identify requirements-relevant expressions within user feedback but also to suggest written requirements and perform quality checks on those quasi-requirements.

ABOUT THE AUTHORS

KARINA VILLELA is a senior researcher at the Fraunhofer Institute for Experimental Software Engineering IESE, where she leads the requirements engineering team. Her research interests include the trend towards ubiquitous requirements engineering, software ecosystems, and variation management. Villela received a Ph.D. in computer science from the Federal University of Rio de Janeiro. Contact her at karina.villela@iese.fraunhofer.de.

EDUARD C. GROEN is a researcher at the Fraunhofer Institute for Experimental Software Engineering IESE. His research interest is deriving requirements from natural-language texts through CrowdRE. Groen received an M.S. in psychology, with a specialization in engineering psychology, from the University of Twente and is pursuing his Ph.D. in computer science at Utrecht University. Contact him at eduard.groen@iese.fraunhofer.de.

JOERG DOERR is the head of the Information Systems division at the Fraunhofer Institute for Experimental Software Engineering IESE and a lecturer at the University of Kaiserslautern. His research interest is software engineering for information systems, focusing on requirements engineering, especially nonfunctional requirements. Doerr received a Ph.D. in computer science from the University of Kaiserslautern. He is a member of the German Informatics Society. Contact him at joerg.doerr@iese.fraunhofer.de.

> Now that digital transformation impacts society as a whole, digital solutions affect everyone.

RE With Everyone to Support the Expression of Needs or Wishes

Software was traditionally developed for users who were familiar with computers or whose tasks would be supported by the software. Now that digital transformation impacts society as a whole, digital solutions affect everyone.[6] Solutions designed to address societal issues, for example, in smart cities or smart rural areas, are intended to be used by people with different interests, skills, and backgrounds. This includes elderly people who have no special technological affinity and may be hard to reach due to their fear of having digital solutions forced upon them. Other settings require inclusive approaches—for example, when designing solutions for people with mental or social impairments (e.g., severe forms of autism). RE traditionally relies on techniques that assume stakeholders are able to express and reflect on their requirements, mostly verbally, or can recognize that a particular solution (e.g., a prototypical implementation) meets their needs. Even more recent techniques, such as design thinking, require end users who are intrinsically motivated and possess collaborative skills. However, to ensure the expected societal or social impact of a digital solution, requirements engineers are increasingly faced with the challenge of engaging end users and understanding their needs based on their interests, skills, and backgrounds.

With the demand for inclusive approaches, requirements engineers need to carefully plan their RE approach to ensure that the RE methods fit the end users' characteristics. To do so, existing RE methods need to be characterized according to aspects that are relevant for actively engaging end users in RE activities, such as duration, frequency, location, and degree of interactivity. Requirements engineers can then characterize end users according to aspects, such as their domain knowledge, attitude toward IT, overall motivation, and temporal availability and select RE methods that fit the characteristics of a specific group of end users. Their choices may lead to new methods being introduced, or to existing ones being employed or adapted.

Involving end users as "cocreators" of a digital solution can help increase participation and acceptance by specific groups. However, it is necessary to offer informal settings in which they can feel comfortable collaborating. The so-called Living Lab approach[7] might provide solutions to this challenge, but the motivation to actively participate in RE or to even show up still remains a challenge. The incorporation of gamification principles and factors that enhance motivation, such as external stimuli or incentives, social interaction, and the assignment of tasks and responsibilities, needs to be investigated. There are only a few studies that report on how they applied cocreation and approaches such as Living Labs in settings with high social impact. Such experiences need to be shared so others can learn about which method is suitable for a particular stakeholder group. Overall, requirements engineers need to make an effort to see eye to eye with the stakeholders on a social level, whether by talking to villagers at the market or engaging in one-on-one sessions with a mentally impaired person under therapeutic guidance.

Overall, requirements engineers need to make an effort to see eye to eye with the stakeholders on a social level.

Working Together

To respond to the challenging demands of digital transformation, RE will have to become ubiquitous in several dimensions, with the role of the requirements engineer remaining central to the success of software products. To achieve this goal, further applied research will be needed to address the practical implications and actual needs of industry and society. Practitioners are invited to embrace the need for change and to provide insights into what they can contribute as well as what their needs are. Only through close collaboration among research, society, and industry can the hurdles that currently still prevent true RE ubiquity be overcome.

References

1. K. Villela et al., "Towards ubiquitous RE: A perspective on requirements engineering in the era of digital transformation," in *Proc. 26th IEEE Int. Requirements Engineering Conf. (RE 18)*, pp. 205–216.
2. S. Jansen, S. Brinkkemper, and M. A. Cusumano, *Software Ecosystems: Analyzing and Managing Business Networks in the Software Industry*, Cheltenham, U.K.: Edward Elgar Publishing, 2013.
3. S. Hess, J. Knodel, M. Naab, and M. Trapp, "Engineering roles for constructing ecosystems," in *Proc. 10th European Conference on Software Architecture Workshops*, 2016, pp. 24–28.
4. K. Villela, S. Kedlaya, and J. Dörr, "An approach to requirements engineering for software ecosystems," in *Proc. Requirements Engineering: Foundation for Software Quality* (Essen 2019), to be published.
5. E. C. Groen et al., "The crowd in requirements engineering: The landscape and challenges," *IEEE Softw.*, vol. 34, no. 2, pp. 44–52.
6. C. Ncube and S.-L. Lim, "On systems of systems engineering: A Requirements engineering perspective and research agenda," in *Proc. 26th IEEE Int. Requirements Engineering. Conf. (RE 18)*, pp. 112–123.
7. J. Salminen, S. Konsti-Laakso, M. Pallot, B. Trousse, and B. Senach, "Evaluating user involvement within living labs through the use of a domain landscape," in *Proc. 17th Int. Conf. Concurrent Enterprising*, 2011, pp. 1–10.

This article originally appeared in IEEE Software, *vol. 36, no. 2, 2019.*

DEPARTMENT: IOT NEWS

The IoT and Digital Transformation: Toward the Data-Driven Enterprise

Alexander A. Pflaum
Fraunhofer Center for Applied Research on Supply Chain Services SCS and Otto-Friedrich University of Bamberg

Philipp Gölzer
Fraunhofer Center for Applied Research on Supply Chain Services SCS

Editor:
Florian Michahelles
florian.michahelles@siemens.com

Internet of Things (IoT) technologies are transforming the focus of business processes from physical products to data-driven services. The authors propose a reference process for digital transformation of the company that goes beyond traditional technology-driven approaches that solely focus on the identification, specification, and implementation of IoT solutions to also include a strategy-driven approach that takes into account complementary technologies and innovations, considers potential barriers to digital transformation, and develops suitable countermeasures.

Internet of Things (IoT) technologies have been with us for a while. During the last two decades, many researchers have made successful advances in smart products, communication protocols and systems, middleware and integration platforms, architectures, and applications. Scientific journals as well as management magazines profile cutting-edge IoT developments. However, less attention has been given to the IoT's economic impact.[1] This article aims to reduce this gap and give some recommendations.

THE IOT: FROM SMART PRODUCTS TO DATA-DRIVEN SERVICES

IoT applications are sometimes called cyber-physical systems (CPSs). While each term has different contexts of use, we use them interchangeably here. From an "end product" point of view, both concepts feature physical goods with powerful embedded microelectronic systems that have their own identity, can sense environmental parameters, determine their position, process data,

2469-7087/20 © 2020 IEEE

make their own decisions, and communicate and cooperate with the environment directly or via an "Internet of Services."[2]

There is a large variety of IoT applications. Smart toothbrushes help keep your teeth and gums healthy. Smart shipping containers monitor transportation processes and protect valuable items from theft and damage. Smart machines constantly monitor their status and request maintenance before a costly breakdown. Smart vehicles such as automated guided vehicles (AGVs)—robots that autonomously move materials in a warehouse—communicate and coordinate production supplies efficiently. Although "smart" products are at the heart of IoT applications, in most cases the full applications require complementary innovations: smart products and CPSs are combined with other technologies such as cloud and mobile computing, digital social networks, and data analytics. The key insight from a management perspective is that the source of innovation does not lie within a single technology; it is the fusion of different technologies that drives innovative IoT solutions. Apps are orchestrated by combining micro services on digital platforms in the cloud and downloaded onto smartphones and other smart products. These, in turn, create data and deliver it to the web and, vice versa, use data provided by the web for their own purposes. The integration of different technologies leads to a new system enabling innovative and formerly unthinkable data-driven services.

TOWARD THE DATA-DRIVEN ENTERPRISE: A CHALLENGE FOR MANAGEMENT

From an innovation management perspective, the main goal behind implementing IoT solutions is the transformation of the traditional product-oriented enterprise into its data-driven counterpart. Eventually, the comprehensive implementation of IoT solutions equates to the digitization of the company. CPSs enhance the granularity and the quality of a firm's data pool. Once translated into knowledge, data enables new service offerings and creates new turnover potential. The question is how to monetize this potential. The activities of digitization pioneers reveal two different strategies. On one hand, a company can turn a physical product into its smart equivalent, embed it into a smart service, develop a suitable business model, and sell it to the market to make additional money. On the other hand, the same company can use smart products from the market to optimize its own production processes and make them more efficient and agile. The Germany-based company Schaeffler, for instance, follows both strategies. It offers smart ball bearings that can monitor temperature, vibration, and lubrication. These smart ball bearings are then integrated into machine tools. The machine tools are in turn used to produce the smart bearings themselves. The underlying process of digital transformation is complex and has to be carefully managed.

A STRUCTURED PROCESS FOR DIGITAL TRANSFORMATION

Digital transformation in industry has typically followed a technology-driven "bottom-up" approach. First, innovation teams within companies are coached in all aspects of digitization technologies and their applications. These teams then identify potential use cases and work on the corresponding solution specifications. Next, the use cases are assessed with regard to both technical feasibility and the economic benefits to the company and are ranked within an implementation roadmap. This approach requires a profound understanding of digitization technologies as well as the firm's processes. If not all knowledge is available inside the company, external experts can be involved.

One might assume that during subsequent implementation there would be no major barriers to overcome. Unfortunately, experience suggests the process is surprisingly challenging. Internal IT departments, for instance, are often too focused on keeping operational systems running and have problems with the innovative character of CPS-based solutions. Use cases are frequently not implemented because cost–benefit analyses are difficult and investment in the necessary IT infrastructure is considered too expensive. Data scientists and other specialists are not commonly available in a company, and the competition for such talent is fierce—hence, quickly finding qualified staff is difficult. Finally, the maturity of digitization technologies is often either over- or understated.

Digital transformation affects a firm's strategy, its offerings, the IT infrastructure, the way to collaborate with partners, its organizational structure, overall process organization, and core competences, as well as the overall company culture at the time.[3] The potential for things going wrong is therefore high. Consequently, the bottom-up approach alone is insufficient to successfully transform a company.

Equally important is a strategy-driven or "top-down" approach that helps to avoid the difficulties mentioned above, as well as to speed up the process. Here, the company first develops a strategic vision of the data-driven version of the enterprise to identify potential barriers and to initiate countermeasures. For this, instruments are needed to identify, structure, and handle upcoming barriers that might arise during the transformation process. For example, maturity models measuring the degree of digital transformation[3] are commonly used as a tool to determine the firm's position in the transformation process and to indicate potential problems and corresponding countermeasures.

Ultimately, the top-down and the bottom-up approaches must be combined to solve the digital transformation problem. To accomplish this, we propose the four-step iterative process shown in Figure 1.

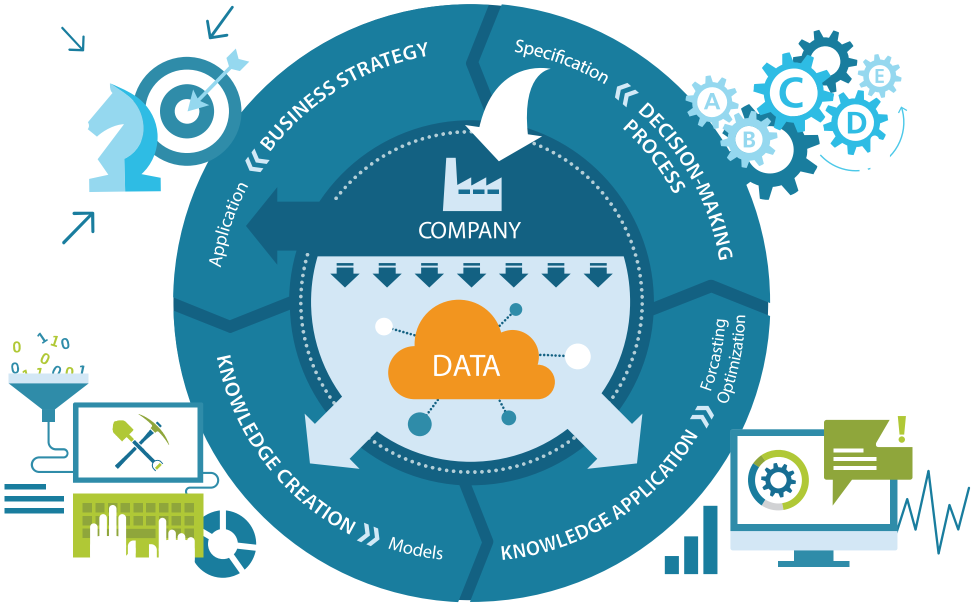


Figure 1. Proposed four-step reference process for digital transformation of companies. (Source: Fraunhofer Center for Applied Research on Supply Chain Services SCS)

The process starts with a Business Strategy step—the creation of a strategic business vision for the data-driven enterprise. This vision must then be broken down into business initiatives and a set of data-driven use cases that support business strategies and goals. Setting the correct framework conditions of the IT infrastructure, partnering network, organization, human resources, innovation culture, and so on require a digital maturity assessment to avoid friction losses during implementation itself. Potential use cases must then be evaluated and prioritized as to business value, data accessibility, and implementation feasibility. The most promising use case can then be implemented ("application").

In the next step of the digital transformation process, Knowledge Creation, a data model is first developed that includes all information necessary to solve the core problem behind the use case ("model"). This model is then populated with data coming from different sources inside and outside the company.

Next, in the Knowledge Application step, companies use AI techniques to get new insights and knowledge from this data and to derive use case–specific solutions ("forecasting, optimization"). Methods and algorithms used in this context usually come from statistics, mathematics, and machine learning and must be integrated into a technical solution corresponding to the given use case. Implementation of models in applications such as a standard procurement system (SPS), a manufacturing execution system (MES), and enterprise resource planning (ERP) can create new business value for the firm.

In the last step, Decision-Making Process, company management must decide how to integrate the data-driven solution into the organizational decision processes ("specification"). The vision, as well as the roadmap, can then be revised and the framework conditions adapted before the next use case is selected. Each implementation gets one step closer to the vision of the data-driven enterprise.

EFFECTS ON THE BUSINESS MODEL

Our own extensive consultancy experience as well as discussions with numerous industry experts have taught us that the realization of smart products and the related digital transformation of a company will fundamentally change the company's business model.[4]

The value proposition of a data-driven enterprise differs significantly: whereas previously the firm offered only a "dumb" physical product, it is now creating value from data and selling data embedded into smart services.[5] The physical product recedes into the background, and the company stops being a traditional manufacturer and starts becoming a service provider.[6–7] The market side of the business model is also subject to significant changes. The product is no longer sold as an investment good but as a service. The payment model changes from a one-off payment to a continuous cash flow based on as-a-service concepts. The market is growing because, thanks to the pay-as-a-service model, even small and medium-size companies can now afford the formerly too-expensive good. The customer is now much more involved in the development of services, as well as in the value-creation process, thus fundamentally changing the character of a company's relationship with its customers. Even the resource side of the business model looks different now: the key activity is turning data into value. Digital platforms[8] are needed to handle the data created and used by smart products. Cost structures are changing because the firm has to establish comprehensive service processes. Additionally, organizational units focusing on the firm's digital transformation have to be set up. And, finally, cooperation models are changing. The company has to recognize that the traditional buyer–seller relationships are disappearing and that it is part of a complex business ecosystem where companies are partners and largely cooperate at eye level.[9]

> The realization of smart products and the related digital transformation of a company will fundamentally change the company's business model.

CONCLUSION

Smart products, which are at the heart of the IoT, will drive the future digital transformation of companies and radically change their business model. The implementation of smart products and corresponding data-driven services must be carefully managed due to its game-changing character. Based on our own experience with consultancy projects as well as discussions with digitization experts, we developed an iterative reference process for digital transformation that is currently being used and evaluated in various industry and research projects carried out by the

Fraunhofer Center for Applied Research on Supply Chain Services SCS in Nuremberg, Germany. Of course, depending on a given company's situation, not all of our recommendations and process steps might be necessary.

Beyond developing the process itself, we have also gained three important insights. First, it is essential that companies first create a vision for their own data-driven enterprise and then align their goals with the process's various activities. Second, digital transformation is a race against time; it is necessary to establish a digitization department as well as a supporting business ecosystem in order to move fast, avoid mistakes, and be efficient. Third, we believe that fundamental change within individual enterprises as well as the industry at large can only occur with an open innovation culture, requiring new types of skills in both data science and service system engineering.

REFERENCES

1. M.E. Porter and J.E. Heppelmann, "How Smart, Connected Products are Transforming Competition," *Harvard Business Rev.*, vol. 92, no. 11, 2014, pp. 64–88.
2. C. Klötzer and A. Pflaum, "Cyber-Physical Systems as the Technical Foundation for Problem Solutions in Manufacturing, Logistics and Supply Chain Management," *Proc. 5th Int'l Conf. Internet of Things* (IoT 15), 2015; doi.org/10.1109/IOT.2015.7356543.
3. C. Klötzer and A. Pflaum, "Toward the Development of a Maturity Model for Digitalization within the Manufacturing Industry's Supply Chain," *Proc. 50th Hawaii Int'l Conf. System Sciences* (HICSS 17), 2017, pp. 4210–4219.
4. A. Osterwalder and Y. Pigneur, *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers*, Wiley, 2010.
5. H. Kagermann, "Change through Digitization—Value Creation in the Age of Industry 4.0," *Management of Permanent Change*, Springer, 2017.
6. V. Eloranta and T. Turunen, "Seeking Competitive Advantage with Service Infusion: A Systematic Literature Review," *J. Service Management*, vol. 26, no. 3, 2015, pp. 394–425.
7. H. Gebauer, E. Fleisch, and T. Friedli, "Overcoming the Service Paradox in Manufacturing Companies," *European Management J.*, vol. 23, no. 1, 2005, pp. 14–26.
8. M.W. Van Alstyne, G.G. Parker, and S.P. Choundary, "Pipelines, Platforms, and the New Rules of Strategy," *Harvard Business Rev.*, vol. 94, no. 4, 2016, pp. 54–60.
9. M. Papert and A. Pflaum, "Development of an Ecosystem Model for the Realization of Internet of Things (IoT) Services in Supply Chain Management—A Grounded Theory Study," *Electronic Markets*, vol. 27, no. 2, 2017, pp. 175–189.
10. C. Frankenberger, T. Weiblen, and O. Gassmann, "Network Configuration, Customer Centricity, and Performance of Open Business Models: A Solution Provider Perspective," *Industrial Marketing Management*, vol. 42, no. 5, 2013, pp. 671–682.

ABOUT THE AUTHORS

Alexander A. Pflaum is director of the Fraunhofer Center for Applied Research on Supply Chain Services SCS and a professor of supply-chain management at Otto-Friedrich University of Bamberg. Contact him at alexander.pflaum@scs.fraunhofer.de.

Philipp Gölzer is director of the Machine Learning and Optimization Group at the Fraunhofer Center for Applied Research on Supply Chain Services SCS. Contact him at philipp.goelzer@scs.fraunhofer.de.

This article originally appeared in
IEEE Pervasive Computing, *vol. 17, no. 1, 2018.*

Department: Internet-of-Things
Editor: Amit Sheth, amit@knoesis.org

Extending Patient-Chatbot Experience with Internet-of-Things and Background Knowledge: Case Studies with Healthcare Applications

Amit Sheth
Kno.e.sis-Wright State University

Saeedeh Shekarpour
University of Dayton

Hong Yung Yip
Kno.e.sis-Wright State University

The transition towards *personalized health management* requires public awareness about management strategies of self-monitoring, self-appraisal, and self-management, eventually paving a way to more timely interventions and higher quality patient–clinician interactions.[1] A key enabler is patient generated health data, fueled in good part by the growth in wearable devices including smart watches and other Internet-of-Things (IoT) for health-tracking (http://bit.ly/smart-wearables). These tracking devices provide "low-level" monitoring signals indicating health conditions such as sleep apnea and heart rhythm disorder. However, to make more sense of IoT data, it is imperative that we develop cognitive approaches where they mine, interlink, and abstract diverse IoT data. These cognitive approaches often needs to keep the user closely engaged to acquire more information, to obtain feedback, to collect verbal health conditions, and to provide intervention and management actions.

The chatbot technology was initially introduced as an artificial conversational agent to simulate conversations with a user using voice or text interactions (http://bit.ly/chatbot-communication).[2] Its market is projected to reach $1.23 billion by 2025 (http://bit.ly/chatbot-market). If this technology is equipped with

Digital Object Identifier 10.1109/MIS.2019.2905748
Date of current version 19 September 2019.

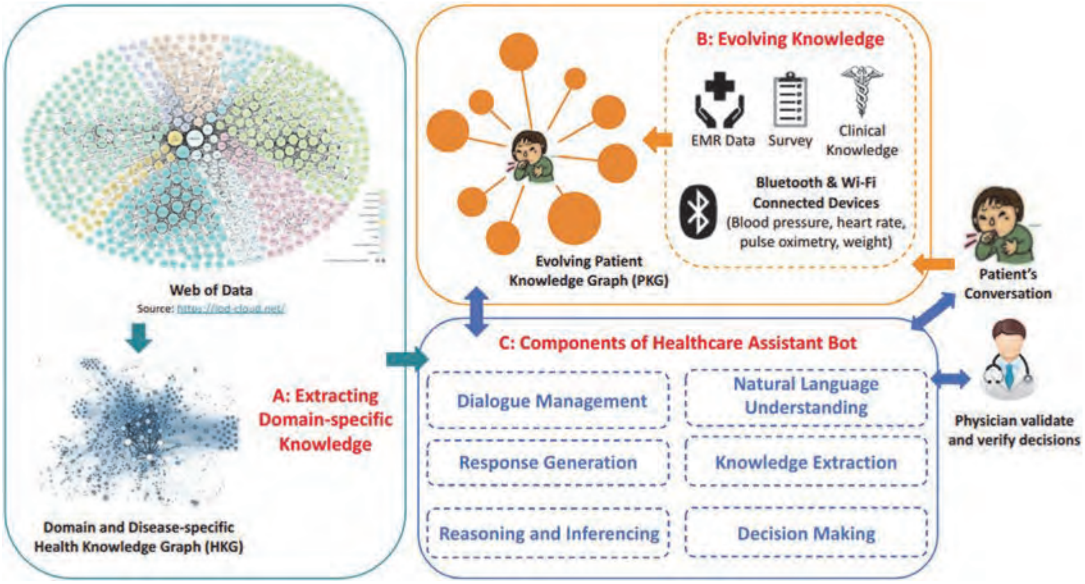

Web of Data
Source: https://lod-cloud.net/
Domain and Disease-specific Health Knowledge Graph (HKG)
A: Extracting Domain-specific Knowledge
B: Evolving Knowledge
EMR Data
Survey
Clinical Knowledge
Bluetooth & Wi-Fi Connected Devices (Blood pressure, heart rate, pulse oximetry, weight)
Evolving Patient Knowledge Graph (PKG)
Patient's Conversation
C: Components of Healthcare Assistant Bot
Dialogue Management
Natural Language Understanding
Response Generation
Knowledge Extraction
Reasoning and Inferencing
Decision Making
Physician validate and verify decisions

Figure 1. A healthcare assistant bot interacts with the patient via various conversational interfaces (voice, text, and visual) to disseminate information and provide recommendation (validated by physician). The core functionalities of the chatbot (Component C in the blue box) are extended with a background HKG (Component A in the green box) and an evolving PHKG (Component B in the orange box).

cognitive capabilities and additionally fed by continuous stream of IoT data, it can accelerate the use of personalized health management applications with improved clinical outcomes. Recently, the coalition of knowledge representation and machine learning has been the center of attention towards a more explainable cognitive computing.[3,4] For a specific domain such as healthcare, the chatbot technology will require advanced cognitive capabilities relying on the representation of background medical knowledge (context) and specific health conditions of patients (personalized knowledge). The incorporation of data collected from IoT and mobile computing (which are often personalized data) into chatbot technology will enable constant tracking of a patient's health condition. Furthermore, it will demonstrate the advancement of current conversational AI capabilities for managing and mining conversations to collect evidence about patients and generate personalized and contextualized inference complemented by knowledge extracted from multiple sources.

In this article, we share our perspective on how the contemporary chatbot technology can be extended towards a more intelligent, engaging, context-aware, and personalized agent. Furthermore, we underline the importance of contextualization, personalization, and abstraction[1] with the use of domain-specific as well as patient-specific knowledge, and present examples of three healthcare applications.

CONTEXTUAL HEALTH KNOWLEDGE GRAPH AND EVOLVING PERSONALIZED KNOWLEDGE GRAPH

A knowledge graph is a structured representation of all the involving concepts, relations, and entities of a given domain. One large public knowledge has been Web of Data that surpasses 149 billion facts collected from 9960 data sets of diverse domains (observed on October 28, 2018, at http://stats.lod2.eu/). AI technologies can take advantage of these big interlinked knowledge. In the following, we first present the motivations and then discuss the two key challenges faced by current health systems. We describe how to augment existing health strategies by extending patient-chatbot experience that relies on three types of input knowledge (see Figure 1): (i) a background *Health Knowledge Graph (HKG)* (see Figure 1A) that comprises of domain and disease-specific knowledge which may be manually developed or extracted from Web of Data that includes a rich source of structured medical and life science data, (ii) an evolving *Patient Health Knowledge Graph (PHKG)* (see Figure 1B) that incorporates Patient Generated Health Data (PGHD) from sensors and IoT devices and structured knowledge extracted from a patient's Electronic Medical Record (EMR) as well as environmental data (e.g., pollen, air quality) from public web services. The PHKG continues to grow by expanding *informative pieces* of knowledge from continuous patient interactions with the chatbot and (iii) is refined by *healthcare provider's feedback* (see Figure 1C) on predictions and analytics.

CURRENT HEALTHCARE CHALLENGES AND PROPOSED SOLUTIONS

Contextualization and Personalization of Patient's Data. The first challenge for developing personal health agent is the need to contextualize and personalize healthcare treatments and decisions. Current healthcare system lacks contextual and personalized knowledge about its patients[3] due to the limited patient–physician time spent during clinical visits, the patient's ability to recall prior events, and clinic-centric system that captures only a part of relevant patient data. Contextual factors in this instance refer to a more in-depth health management and clinical protocol knowledge that a physician may utilize, whereas personalized factors include a patient's health history, data capturing patients health condition (e.g., a lab or BMI), ongoing activities, and lifestyle choices. A survey presented in the article by Linder *et al*.[5] reports several notable barriers to the effective use of clinical decision support systems during patient visits, including physician losing direct eye contact with patients, falling behind schedule, inability to type quickly enough, and feeling that using the computer in front of the patient is rude. It concludes that EMRs have mixed effectiveness for supporting decision-making of physicians since exploring them is not reasonably agile to derive effective knowledge.[4] These factors can potentially lead to missing patients' data and likely to affect other healthcare professionals who utilize these data. On the bright side, patients are increasingly using technology (e.g., wearables) and using mobile applications to generate what is termed PGHD. Incorporation of such data in better health management is likely to become more important, and chatbots can further make it easier to collect some of the patient data such as symptoms or how a patient feels.

Contemporary implementations of chatbot technologies do not understand conversation narrative and demonstrate very limited cognitive capabilities and commonsense reasoning. Handling these limitations for a broad domain might take years, but in a specific domain such as health care, and even narrower applications, such as a specific disease, these limitations can be alleviated by extending the chatbot technology with *domain and disease-specific health background knowledge* (i.e., contextual and personalized knowledge). There are publicly available generic knowledge graphs (e.g., DBpedia and Freebase) as well as healthcare-specific knowledge source, e.g., unified medical language system, PubMed, systematized nomenclature of medicine-clinical term, and International classification of diseases. Chatbot technology can acquire a context-aware (i.e., patient's context), domain-specific (i.e., health domain) knowledge graph (extracted and integrated from external sources such as Web of Data) termed *HKG*. The HKG can be updated and synchronized by the evolution of Web of Data or relevant knowledge sources. HKG provides essential facts (background knowledge) that are necessary for *response generation, reasoning, and inference* components of chatbot engine. The other obstacle to have a holistic overview of a patient's circumstance is the lack of a *unified* and *semantic-based* approach for *publishing and integrating an individual patient's data*. This gap hinders the health care system to provide a comprehensive history and insight about patients. To tackle this deficiency, we propose to publish a knowledge graph out of anonymized patient data that is collected from various sources (knowledge collected from EMR, IoTs devices, and external web services). PHKG further integrates knowledge extracted from previous *conversations of patients with chatbot.* To sum up, having two background knowledge graphs (see Figure 1) to feed the core chatbot engine will enhance reasoning and prediction in support of improving health decision making.

LIMITED PATIENT HEALTH DATA DUE TO EPISODIC VISITS AND TIME CONSTRAINTS

The American Academy of Family Physicians (AAFP) defines *primary care* as promoting effective communication and encouraging the role of the patients as partner in healthcare. During clinical visit, the primary care physician assumes the primary contact of patients for diagnosing a wide range of illnesses and injuries, counselling, and education as well as initiating preventative care. They are also responsible for making referrals to specialists according to the patient's condition. This is a task of significant responsibility since a patient may endure prolonged suffering in case of a wrong referral. However, with increasing societal demand to healthcare resources, a significant percentage of physicians reported that they ran out of consultation time to converse and accurately diagnose the root cause of patients' conditions (http://bit.ly/clinical-challenges). Consequently, some patients are being deprived of education about their health conditions, causes, available treatments, and education (such as on lifestyle changes). This indicates a worrisome gap in *collecting, managing and analyzing patient's health data* as well as a proper mechanism for *educating, advising, and referring* patients.

Mobile devices and IoTs are increasingly prevalent with overall improved technology literacy among populations. They can hence be leveraged for continuous real-time tracking of patient health signals. These signals can help in bridging the information gap between each hospital visit and providing just-in-time adaptive interventions.[6] For example, a joint project between Kno.e.sis and Dayton Children's Hospital has developed knowledge-enabled semantic multi-sensory approach for personalized pediatric asthma management (kHealth, http://bit.ly/kHealth-Asthma).[7] The kHealth-Asthma kit represented in Figure 2 consists of an Android application that asks contextual questionnaire (tailored to specific conditions of the user) to capture symptoms and medication usage. It also uses IoT and Web Services to collect patient's and patient relevant relevant data including (a) physiological data captured via Fitbit (activity and sleep) and Peak Flow meter (PEF/FEV1 values); (b) indoor environmental data (particulate matter, volatile organic compound, CO2, humidity, and temperature) using Foobot, an indoor air quality monitor; (c) outdoor allergens and air quality recorded using web services (ozone, pollen, and air quality); and (d) selected data semi-automatically (human validation with strict anonymization) extracted from patient's clinical notes (from EMRs). A total of 110 evaluations in this 150+ planned completed pediatric asthma patient cohort study have been completed, each lasting one or three months of participation. A compliance rate of 89% (defined as over 75% of data requiring active patient participation) shows the user acceptance of such a technology. The total number of data points collected per patient per day is up to 1852 over 29 types of parameters. All data are anonymized and securely backup on the Kno.e.sis cloud. These data are integrated together using a visualization and analysis platform, kHealthDash (http://bit.ly/kHealthDash).

ARCHITECTURE OVERVIEW OF A HEALTH CHATBOT

Content, user interface, and user feedback are three major components that go hand-in-hand in creating a positive user experience which is a critical for defining the relationship a user has with a chatbot. Having the chatbot's core functionalities extended with HKG and PHKG help contextualize and personalize conversations. However, without an equally strong frontend communication system to (a) receive user input and (b) articulate smart responses by (c) making intelligent inferences and prediction, user interest and experience may decline and diminish over time (http://bit.ly/why-chatbots-fail). The six core components of the chatbot (see Figure 1C) each represents a research problem: conversation management, natural language (narrative) understanding, response generation, knowledge extraction and discovery, reasoning and inference engine, and prediction module. The following are the proposed extensions to the current state-of-the-art approaches to improve patient experience in using a chatbot.

(a) *Receive and understand user input:* A chatbot should be sufficiently dynamic to communicate with patients via multiple input and output modalities including voice, text, and smart display. The chatbot should provide feedback to the user and affirm its

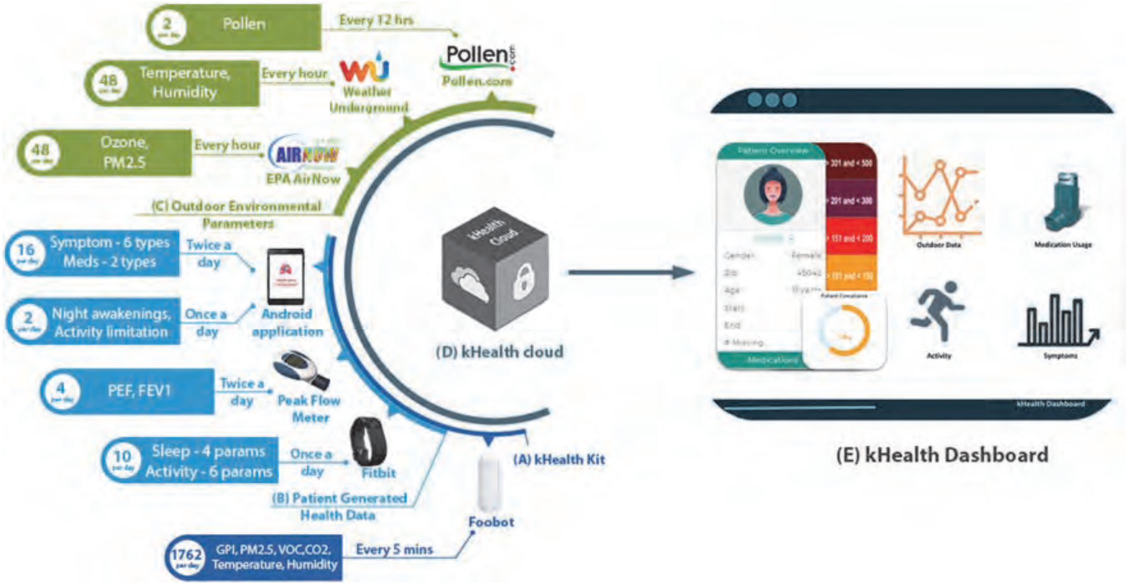


Figure 2. The kHealth framework with kHealth-Asthma kit, kHealth cloud (D), and kHealth Dashboard (E), showing the frequency of data collection, the number of parameters collected, and the total number of data points collected per day per patient (shown in dark blue). The kHealth kit components that are given to patients which collects PGHD are shown in light blue and the outdoor environmental parameters with their sources are shown in green. All data are anonymized and associated with respective randomly assigned patient IDs.

understanding to avoid conflict and knowledge mis-representation.

(b) *Generating smart responses:* The responses articulated by the chatbot are reasoned from the underlying HKG and PHKG to guarantee domain-specificity and contextualization as well as personalization aspects. The "smart" is attributed by the following components:

(i) *Comprehensible and concise.* Conciseness and comprehensibility of answers profoundly matter as a slight flaw could compromise reliability.

(ii) *Context-awareness and coherence.* The chatbot should consider the patient's context in terms of space and time in addition to the input provided. For example, if an asthmatic patient asks for the weather condition, a generic answer would be "Today is fairly sunny" versus a personalized answer with respect to the patient's disease "Today is fairly sunny. However, the ragweed pollen is a little high which does not look too good for your health. Do remain indoor as much as possible." The latter illustrates context awareness.

(iii) *Dynamicity and evolution.* The more the patient interacts with the chatbot, the more knowledge it discovers about the patient. In addition, knowledge evolves over time and they should be reflected on the knowledge bases (HKG and PHKG).

(iv) *Balancing response granularity and volume.* The complexity of traversing the graphs followed by reasoning and formulating a response, either by visualization or verbalization, increases dramatically with the volume of data. Retrieving and balancing an optimum amount of data, yet sufficient for a reasonable response is critical to communicate timely and effectively.

(c) *Inference, reasoning, and prediction.* As knowledge evolves, both HKG and PHKG should be continuously updated to infer new insights (http://bit.ly/PHKG-evolution). The prediction module relies on both new and historical knowledge about the patient in order to infer, reason, and make a reasonable recommendation to assist the patient for self-management and self-appraisal. The predictions are also continuously presented to the corresponding physicians to create situational awareness, and in case of an emergency condition, the physician can be notified immediately to intervene.

CASE STUDIES WITH HEALTHCARE APPLICATIONS

The first use case is major depressive disorder. Depression is highly prevalent in the U.S. with estimated prevalence rates of 10.5% affecting millions of U.S. adults (http://bit.ly/major-depression). Successful early identification and intervention, albeit challenging, can lead to positive health and behavioral improvements. A routine screening for depression by a clinician involves administering a Patient Health Questionnaire (PHQ-9, http://bit.ly/PHQ-9) which relies heavily on patient's ability to recall events that occurred over the span of last two weeks. Instead, a chatbot can directly converse with patient to collect relevant data on a continuous basis in real-time, or as an added option, a patient can consent the chatbot to use his social media conversations to indirectly assess some of the components of PHQ-9 assessment and directly converse with a patient for the remaining information needed for an assessment. The patient's PHKG can represent patient's past encounters and behavioral manifestations (optionally on social media) over a substantial period of time for a more accurate prognosis. In addition, a contextualized chatbot with domain knowledge can understand slang terms that are commonly used in social media such as "bupe" which refers to its medical term "buprenorphine." This allows a viable entry for a chatbot to deliver tailored psychotherapy based on *Cognitive-behavioral therapy*[8] and initiate the need for treatment intervention conforming to medical protocols.

The second use case is asthma. More than 20.4 million people in the U.S. are diagnosed with asthma in 2016 and asthma-related healthcare costs alone are around $50 billion a year (http://bit.ly/asthma-facts). In an attempt to bridge the informational gap between episodic patient–doctor visits, a chatbot can combine active and passive sensing using a variety of low-cost sensors and IoTs for continuous monitoring and collection of multimodal data. These longitudinal measurements can then be leveraged and transformed into practical and actionable information for both patient and health care provider. Specifically, the patient can access information with regards to his/her asthma control level based on symptoms, severity, and triggers for self-monitoring, all at the convenience by conversing with the chatbot.

The third prominent use case is elderly care. With improved healthcare services and amenities, elder residents are becoming one of the fastest growing cohort.[9] These older adults however are at the highest risk for developing chronic diseases such as heart failure and chronic obstructive pulmonary disease. As the technology matures, a chatbot with consented access to patient–doctor profile and social information can be delegated to match patient–doctor preferences, organize telehealth sessions, and schedule appointments by looking up the doctor's calendar. Extending with IoTs such as pill's bottle sensor, the chatbot can be made smarter to remind and nudge patient of timely medication intake as well as adherence to clinician prescribed management plan. By incorporating background geospatial and gazetteers knowledge sources, it is also feasible to coordinate and arrange transportation service for elderly with physical disabilities and transportation barriers, especially in congested cities.

In conclusion, while the chatbot technology is not new, we discussed how its potential can be extended with IoTs and knowledge graphs. We further illustrated the possible health services a chatbot can intervene using three disease-specific use cases. To sum up, the chatbot technology can (i) be empowered with multisensory capabilities through IoTs and sensors, (ii) provide contextualized and personalized reasoning capabilities grounding with domain-specific knowledge, and (iii) assist situations requiring high cognitive load. These diverse potentials hold prodigious promising for a close future of promoted healthcare approach.

REFERENCES

1. A. Sheth, U. Jaimini, and H. Y. Yip, "How will the internet of things enable augmented personalized health?," *IEEE Intell. Syst.*, vol. 33, no. 1, pp. 89–97, Jan./Feb. 2018.

2. L. Pichponreay, J. H. Kim, C. H. Choi, K. H. Lee, and W. S. Cho, “Smart answering chatbot based on OCR and overgenerating transformations and ranking,” in *Proc. 8th Int. Conf. Ubiquitous Future Netw.*, pp. 1002–1005, Jul. 2016.
3. A. Holzinger, C. Biemann, C. S. Pattichis, and D. B. Kell, “What do we need to build explainable AI systems for the medical domain?,” arXiv:1712.09923, 2017.
4. W. Samek, T. Wiegand, and K. R. Müller, “Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models,” arXiv:1708.08296, 2017.
5. J. A. Linder, J. L. Schnipper, R. Tsurikova, A. J. Melnikas, L. A. Volk, and B. Middleton, “Barriers to electronic health record use during patient visits,” in *Proc. AMIA Annu. Symp.*, 2006, vol. 2006, pp. 499–503.
6. I. Nahum-Shani *et al.*, “Just-in-time adaptive interventions (JITAIs) in mobile health: Key components and design principles for ongoing health behavior support,” *Ann. Behav. Med.*, vol. 52, no. 6, pp. 446–462, 2017.
7. U. Jaimini, K. Thirunarayan, M. Kalra, R. Venkataraman, D. Kadariya, and A. Sheth, “How is my child’s asthma?” Digital phenotype and actionable insights for pediatric asthma, JMIR Pediatrics and Parenting, vol. 1, no. 2, 2018, Art. no. e11988.
8. K. K. Fitzpatrick, A. Darcy, and M. Vierhile, “Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (Woebot): A randomized controlled trial,” *JMIR Mental Health*, vol. 4, no. 2, 2017.
9. A. E. Schneider, N. Ralph, C. Olson, A. M. Flatley, and L. Thorpe, “Predictors of senior center use among older adults in New York City public housing,” *J. Urban Health*, vol. 91, no. 6, pp. 1033–1047, 2014.

Amit Sheth works on semantic-cognitive-perceptual computing and knowledge-enhanced learning with application in healthcare and social good. He is a fellow of IEEE, AAAI, and AAAS. He is the corresponding author of this article and can be reached at amit.shet@wright.edu and http://knoeiss.org/amit.

Hong Yung (Joey) Yip is currently working toward the Ph.D. degree on topics in knowledge graph, deep learning, and conversational AI. Contact him at joey@knoesis.org.

Saeedeh Shekarpour is an Assistant Professor of computer science with the University of Dayton, Dayton, OH, USA. She works on knowledge graphs and cognitive computing in question answering and chatbot technologies. Contact her at sshekarpour1@udayton.edu.

This article originally appeared in
IEEE Intelligent Systems, *vol. 34, no. 4, 2019.*

Assistant Professor of Computer Engineering

The University of Southern Mississippi

The School of Computing Sciences and Computer Engineering in the College of Arts and Sciences at the University of Southern Mississippi is seeking applications for one tenure-track, assistant professor position in the field of Computer Engineering with a start date of fall 2020. The position will be based at the university's main campus in Hattiesburg.

Candidates must have a Ph.D. in Computer Engineering or a closely related field, and be able to demonstrate the ability to develop a successful research program and participate effectively in the development and teaching of the Computer Engineering curriculum. Applicants with a wide range of interests in Computer Engineering are encouraged to apply while areas of expertise related to Internet of Things, rapid prototyping, embedded systems, system on chip, cloud computing and, especially, cybersecurity will be given priority consideration.

Applications must include: CV; cover letter; brief statement of teaching philosophy; description of research interests; and at least three references. The position will remain open until filled.

The University of Southern Mississippi is a public, Doctoral University with Very High Research Activity. The new Computer Engineering program started in 2017 and is in a rapid phase of expanding research and education activities and offers excellent opportunities for interdisciplinary and industrial collaborations.

All are encouraged to visit the university and the School's websites starting from http://www.usm.edu/computing-sciences-computer-engineering/ for general information, and potential applicants may contact the Search Committee Chair, Dr. Amer Dawoud, amer.dawoud@usm.edu for specific inquiries (Job Req # 1252).

The University of Southern Mississippi is an equal employment opportunity employer. All qualified applicants will receive consideration for employment without regard to race, color, religion, gender, national origin, age, disability or veteran status.

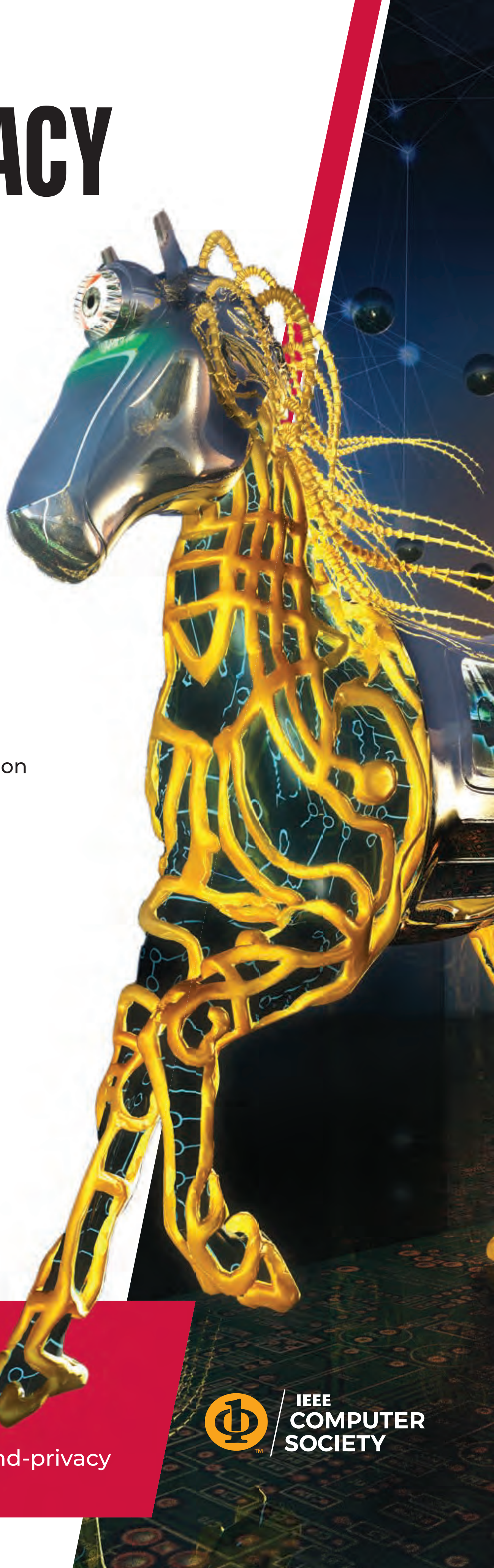
Senior Software Engineer (Sacramento, CA)

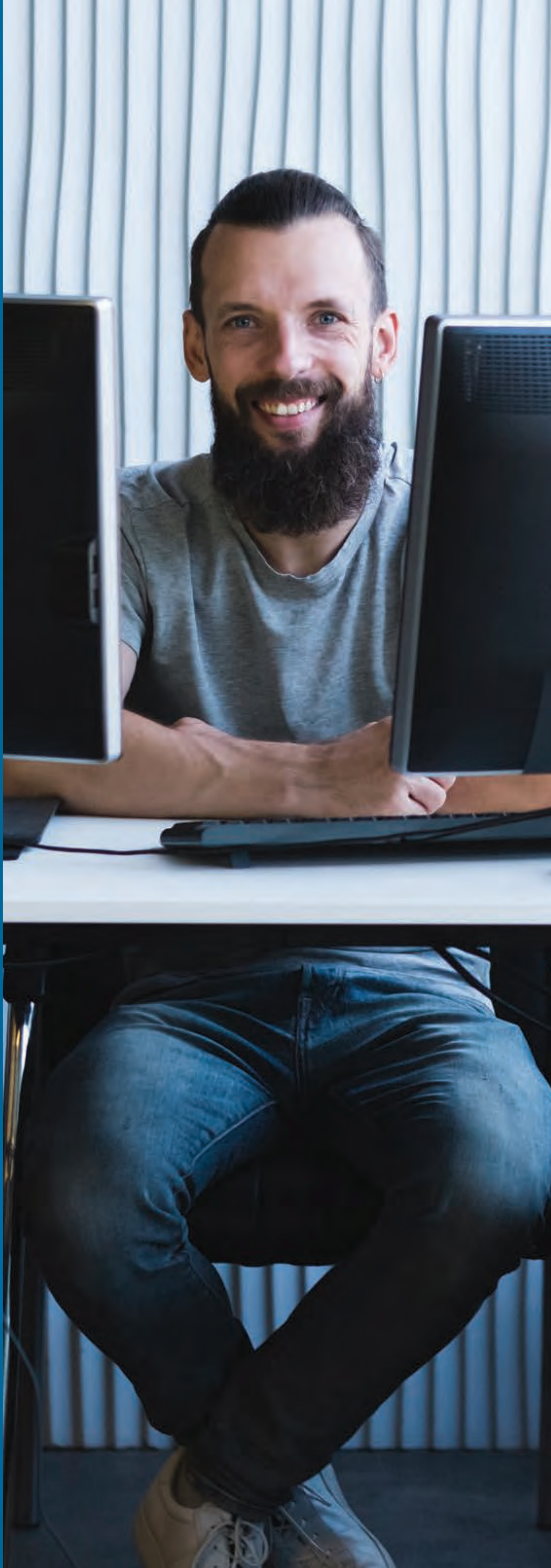
IT company.

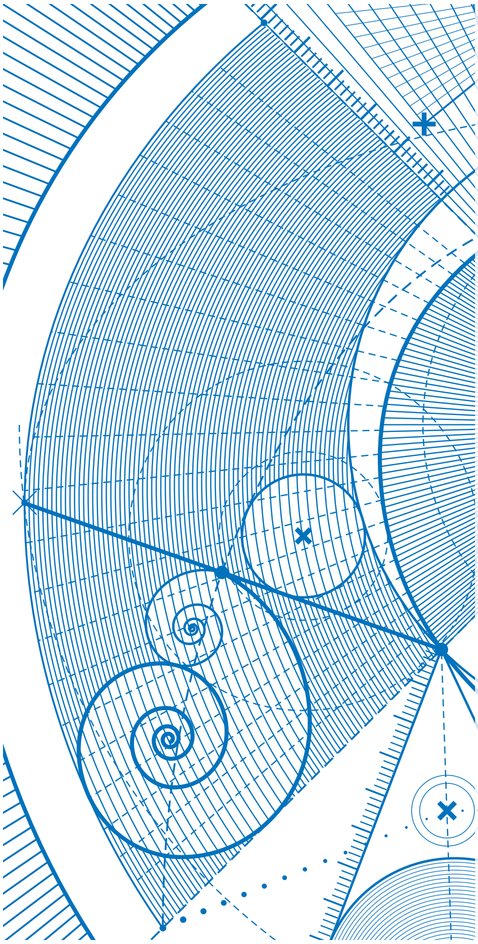
Masters+2 yrs (Comp Science, Engineering or related field) Develop, create and modify general computer applications software or specialized utility programs. Analyze user needs and develop software solutions. Design software or customize software for client use with the aim of optimizing operational efficiency. May analyze and design databases within an application area, working individually or coordinating database development as part of a team using waterfall and Agile, Lean Six Sigma, Main frame Cobol, RQM, Clear Quest, JIRA, IBM web sphere, IBM clear case. Travel and/or Relocation to various unanticipated sites within the U.S. may be required.

**Apply with 2 copies of resume to HR, HTH Enterprise, LLC,
5155 Madison Avenue, Suite 21,
Sacramento, California 95841**

E-Currency and Fairness ▪ Ransomware Defense ▪ A National Cybersecurity Policy
IEEE SECURITY & PRIVACY
SECURITY & PRIVACY
SECURITY & PRIVACY
SECURITY & PRIVACY
Privacy and Automated Airport Screening
BERSECURITY AND
CY ISSUES IN BRAZIL







Conference Calendar

Questions? Contact conferences@computer.org

IEEE Computer Society conferences are valuable forums for learning on broad and dynamically shifting topics from within the computing profession. With over 200 conferences featuring leading experts and thought leaders, we have an event that is right for you.

Find a region:	Africa	■	Australia	◆	North America	◗
	Asia	▲	Europe	●	South America	★

February

3 February

- ICSC (IEEE 14th Int'l Conf. on Semantic Computing) ◗

18 February

- SANER (IEEE 27th Int'l Conf. on Software Analysis, Evolution and Reengineering) ◗

19 February

- BigComp (IEEE Int'l Conf. on Big Data and Smart Computing) ▲

22 February

- CGO (IEEE/ACM Int'l Symposium on Code Generation and Optimization) ◗

March

2 March

- WACV (IEEE Winter Conf. on Applications of Computer Vision) ◗

9 March

- DATE (Design, Automation & Test in Europe Conf. & Exhibition) ●
- IRC (4th IEEE Int'l Conf. on Robotic Computing) ▲

16 March

- ICSA (IEEE Int'l Conf. on Software Architecture) ★

22 March

- VR (IEEE Conf. on Virtual Reality and 3D User Interfaces) ◗

23 March

- ICST (13th IEEE Conf. on Software Testing, Validation and Verification) ●
- PerCom (IEEE Int'l Conf. on Pervasive Computing and Communications) ◗

April

5 April

- ISPASS (Int'l Symposium on Performance Analysis of Systems and Software) ◗

14 April

- PacificVis (IEEE Pacific Visualization Symposium) ▲

20 April

- ICDE (IEEE 36th Int'l Conf. on Data Eng.) ◗

May

3 May

- FCCM (IEEE 28th Annual Int'l Symposium on Field-Programmable Custom Computing Machines) ◗

4 May

- HOST (IEEE Int'l Symposium on Hardware Oriented Security and Trust) ◗

18 May

- SP (IEEE Symposium on Security and Privacy) ◗

- FG (IEEE Int'l Conf. on Automatic Face and Gesture Recognition) ★
- IPDPS (IEEE Int'l Parallel and Distributed Processing Symposium) ◗

23 May

- ICSE (IEEE/ACM 42nd Int'l Conf. on Software Eng.) ▲

30 May

- ISCA (ACM/IEEE 47th Annual Int'l Symposium on Computer Architecture) ●

June

14 June

- CVPR (IEEE Conf. on Computer Vision and Pattern Analysis) ◗

16 June

- EuroS&P (IEEE European Symposium on Security & Privacy) ●

19 June

- JCDL (ACM/IEEE Joint Conf. on Digital Libraries) ▲

29 June

- DSN (50th Annual IEEE/IFIP Int'l Conf. on Dependable Systems and Networks) ●

30 June

- MDM (21st IEEE Int'l Conf. on Mobile Data Management) ●

July

6 July

- ICME (IEEE Int'l Conf. on Multimedia and Expo) ●

8 July

- ICDCS (IEEE 40th Int'l Conf. on Distributed Computing Systems) ▲

13 July

- COMPSAC (IEEE Annual Computer Software and Applications Conference) ●

August

31 August

- RE (IEEE 28th Int'l Requirements Eng. Conf.) ●

September

21 September

- ASE (35th IEEE/ACM Int'l Conf. on Automated Software Eng.) ◆

28 September

- ICSME (IEEE Int'l Conf. on Software Maintenance and Evolution) ◆
- SecDev (IEEE Secure Development) ◗

October

18 October

- MODELS (ACM/IEEE 23rd Int'l Conf. on Model Driven Eng. Languages and Systems) ◗

21 October

- FIE (IEEE Frontiers in Education Conf.) ●

25 October

- VIS (IEEE Visualization Conf.) ◗

November

9 November

- FOCS (IEEE 61st Annual Symposium on Foundations of Computer Science) ◗

15 November

- SC ◗

16 November

- LCN (2020 IEEE 45th Conf. on Local Computer Networks) ◆

December

10 December

- AIKE (IEEE Third Int'l Conf. on Artificial Intelligence and Knowledge Eng.) ◗