COMPUTING

edge

- Semantic Computing
- Artificial Intelligence
- Project Management
- Education

JANUARY 2024

www.computer.org

IEEE COMPUTER SOCIETY

IEEE

COMPUTING edge

IEEE COMPUTER SOCIETY computer.org

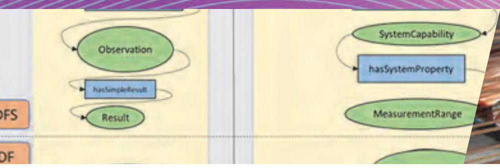
IEEE COMPUTER SOCIETY · IEEE · Printed with inks containing soy and/or vegetable oils

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.

JANUARY 2024 · VOLUME 10 · NUMBER 1

COMPUTING edge



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

Impact and Limitations of AR-Based Guidance for Assembly Workers

To assess the extent to which augmented reality (AR) could displace traditional assembly instructions, the authors of this November 2023 *Computer* article conducted a field study in a German manufacturing company. In short, inexperienced workers were introduced to an assembly process with fully AR-based guidance presented on AR glasses. To evaluate the outcome, the authors documented the workers' errors and tracked whether they were able to successfully perform the process until the end without any additional instruction or assistance.

Computing in Science & Engineering

Compatible Finite Elements for Glacier Modeling

Predicting glacier evolution requires models that can resolve the physics associated with both mountain and marine environments and that remain stable and mass conservative in the presence of extreme topography while still resolving membrane stresses. In this article from the May/June 2023 issue of *Computing in Science and Engineering*, the authors explore the application of a class of mixed finite elements to glacier flow models that accomplish these goals.

IEEE Annals of the History of Computing

Early AI in Britain: Turing et al.

This article from the July–September 2023 issue of *IEEE Annals of the History of Computing* presents an overview of Turing's early contributions to machine intelligence, together with a summary of his influence on other early practitioners. He staked out the field of machine intelligence during the 1940s. And in key papers in 1948 and 1950 he discussed search, learning, robotics, chess, the theorem-proving approach to AI, the genetic algorithm concept, and artificial neural networks, as well as introducing the Turing test. He influenced the first generation of programmers in Britain, whose pioneering contributions to machine intelligence included work on board games, learning, language-processing, and reasoning—contributions made years before the term "Artificial Intelligence" was coined at Dartmouth in 1956.

IEEE Computer Graphics and Applications

Visualizing Uncertainty in Sets

To address the problem of depicting uncertainty in set visualization, the authors of this September/October 2023 *IEEE Computer Graphics and Applications* article ask 1) which aspects of set type data can be affected by uncertainty and 2) which characteristics of uncertainty influence the visualization design. To answer these research questions, they first described a conceptual framework that brings together 1) the information that is primarily relevant in sets and 2) different plausible categories of (un)certainty. Following the

2469-7087/24 © 2024 IEEE

structure of their framework, the authors systematically discussed basic visualization examples of integrating uncertainty in set visualizations.

IEEE Intelligent Systems

New User Intent Discovery With Robust Pseudo Label Training and Source Domain Joint Training Big Data Analytics and Mental Health: Would Ethics Be the Only Safeguard Against the Risks of Identifying "Potential Patients"?

Even if, at some point, computers provide diagnoses with greater accuracy than medical professionals, would ethics be the only safeguard against the possible risks? In this *IEEE Intelligent Systems* article from the September/October 2023 issue, the authors describe a pragmatic approach for answering this question by focusing on possible outcomes concerning "potential patients"—those who have not yet shown signs of disease. Through the outcomes and inferences derived from big data analysis, such patients could require early treatment, more accurate diagnoses, and medications that are better adapted to their conditions.

IEEE Internet Computing

EQuaTE: Efficient Quantum Train Engine for Runtime Dynamic Analysis and Visual Feedback in Autonomous Driving

In this article from *IEEE Internet Computing*'s September/October 2023 issue, the authors propose an efficient quantum train engine (EQuaTE), a novel development tool for quantum neural network (QNN) autonomous driving software, which plots gradient variances to confirm whether the QNN falls into local minima situations (called barren plateaus). Based on this runtime visualization, the stability and feasibility of QNN-based software can be tested during runtime operations of autonomous driving functionalities.

IEEE micro

On-Device Tiny Machine Learning for Anomaly Detection Based on the Extreme Values Theory

The significance of anomaly detection is particularly pronounced in Industry 4.0 applications. The authors of this November/December 2023 *IEEE Micro* article present an unsupervised on-device learning tiny machine learning (TinyML) algorithm, drawing inspiration from the extreme value theory. Their algorithm leverages the two-parameter Weibull distribution function to efficiently identify anomalies within discrete time series data. Optimal hyperparameters are ascertained via grid search methodology. Notably, employing synthetic data with randomized anomalies elucidates the algorithm's proficiency in binary classification within time series, highlighting an accuracy of 99.80%, recall of 93.10%, and F1 score of 96.43%.

IEEE MultiMedia

An Improved Interaction Estimation and Optimization Method for Surveillance Video Synopsis

The authors of this July–September 2023 article in *IEEE Multimedia* present a comprehensive video synopsis framework. First, they propose an interaction detection method to estimate distortion less spatio-temporal interactions between moving objects by generating the top view of a scene using a perspective transformation. Second, they propose an optimization method to reduce collisions and preserve object interactions by shrinking the search space. The experimental results demonstrate that the proposed framework provides a better estimate for object interactions from surveillance videos and generates synopsis videos with fewer collisions while preserving original interactions.

Toward Personalized Music-Therapy: A Neurocomputational Modeling Perspective

In this article from *IEEE Pervasive*'s July–September 2023 issue, the authors summarize the evidence of music-based interventions primarily in motor, emotional, and cardiovascular regulation and put this in context by briefly describing the current theories of music perception and processing. The article also highlights opportunities for music therapy to improve quality of life and reduce stress beyond the clinic environment and in healthy individuals. This could become even more effective therapeutically as personalization and automation of music selection processes improve to fit individual needs via feedback loops mediated by measurements of neurophysiological responses.

What's Driving Conflicts Around Differential Privacy for the U.S. Census

The U.S. Census Bureau's use of differential privacy has been fiercely debated among interested parties. Accuracy loss has been at the forefront, but conflicting confidentiality notions help explain why common ground is lacking. The authors of this *IEEE Security & Privacy* article from the September/October 2023 issue propose three ways of understanding confidentiality conflicts and offer suggestions for researchers and organizations adopting formal privacy.

Scaling Student Feedback in Software Engineering Massive Open Online Courses

This article in the September/October 2023 issue of *IEEE Software* presents a bot aimed at helping massive open online course participants complete programming assignments by providing them with timely formative feedback. The use of bots could be a solution for providing individual evaluation and formative feedback to students throughout the whole online learning process. The authors found that students' perceptions toward the bot were positive, and statistics indicated that many relied on it to complete assignments.

Misinformation Detection Using Deep Learning

There is growing interest in using deep learning to detect misinformation, particularly given the accuracy of detection. However, there are numerous challenges. This article in the September/October 2023 issue of *IT Professional* explores the various types of misinformation attacks and the deep learning architectures used to detect them. With a review of recent literature, the authors put forward a classification of deep learning approaches and their relative effectiveness in detecting misinformation; they assess the limitations in terms of accuracy as well as computational overhead, and they review the challenges that arise from the use of these technologies to detect misinformation.

Developments in Semantic Computing

Recent years have marked a dramatic leap in the capabilities of semantic computing. In particular, artificial intelligence (AI) is becoming more prevalent through the varied uses of generative AI. There is also a heightened focus on the metaverse, where semantic computing plays an important role in enabling connection in a virtual setting. While these advancements come with benefits, they also introduce risks. This issue of *ComputingEdge* highlights new developments in semantic computing as well as the security risks they pose.

"A Semantic Web Approach to Fault Tolerant Autonomous Manufacturing," from *IEEE Intelligent Systems* demonstrates how Semantic Web capabilities can improve machine use and efficiency, paving the way for autonomous manufacturing. In *IEEE Internet Computing*'s "*i*MetaverseKG: Industrial Metaverse Knowledge Graph to Promote Interoperability in Design and Engineering Applications," the authors propose using knowledge graph (KG) technologies to address the challenges that are preventing the industrial metaverse (*i*Metaverse) from becoming a reality.

Since its release one year ago, millions of people worldwide have used ChatGPT in a variety of ways. The authors of "ChatGPT in Developing Economies," from *IT Professional*, analyze the benefits and limitations of ChatGPT in developing nations. *IEEE Security & Privacy*'s "Electric Sheep on the Pastures of Disinformation and Targeted Phishing Campaigns: The Security Implications of ChatGPT" grapples with the ethical, moral, and legal implications of weaponizing generative AI programs.

In this rapidly changing environment, developers may need a reminder to prioritize standards and users when creating quality products. In *Computer*'s "A Software Project That Partially Failed: A Small Organization That Ignored the Management and Technical Practices of Software Standards," the authors explore the repercussions of ignoring software engineering standards. *IEEE Software*'s "The Importance of a Usage-Centric Approach" discusses how to create helpful and relevant apps by focusing on user needs, abstraction, and project flexibility.

This issue of *ComputingEdge* concludes by underscoring the importance of education. The authors of "MolSSI Education: Empowering the Next Generation of Computational Molecular Scientists," from *Computing in Science & Engineering*, explain how the Molecular Sciences Software Institute (MolSSI) supports software development, training, and engagement in the computational molecular sciences (CMS). *Computer*'s "Teaching Modeling in the Time of Agile Development" shows how educators can use agile practices to teach modeling.

EDITOR: Amit Sheth, amit@knoesis.org

This article originally appeared in Intelligent Systems vol. 38, no. 1, 2023

A Semantic Web Approach to Fault Tolerant Autonomous Manufacturing

Fadi El Kalach, Ruwan Wickramarachchi, Ramy Harik, and Amit Sheth,
University of South Carolina, Columbia, SC, 29208, USA

The next phase of manufacturing is centered on making the switch from traditional automated to autonomous systems. Future factories are required to be agile, allowing for more customized production and resistance to disturbances. Such production lines would be able to reallocate resources as needed and minimize downtime while keeping up with market demands. These systems must be capable of complex decision-making based on parameters, such as machine status, sensory/IoT data, and inspection results. Current manufacturing lines lack this complex capability and instead focus on low-level decision-making on the machine level without utilizing the generated data to its full extent. This article presents progress toward this autonomy by introducing Semantic Web capabilities applied to managing the production line. Finally, a full autonomous manufacturing use case is also developed to showcase the value of Semantic Web in a manufacturing context. This use case utilizes diverse data sources and domain knowledge to complete a manufacturing process despite malfunctioning equipment. It highlights the benefit of Semantic Web in manufacturing by integrating the heterogeneous information required for the process to be completed. This provides an approach to autonomous manufacturing not yet fully realized at the intersection of Semantic Web and manufacturing.

Smart manufacturing (SM) has taken a front seat in the advancement of manufacturing production lines. This vision of SM will be powered by industrial Internet of Things, big data analytics, and artificial intelligence. These capabilities can have a substantial effect on the profitability of the industry as a whole since manufacturers must respond to fast-changing requirements through productivity advancements and agility while maintaining high standards.

One aspect of manufacturing that SM seeks to tackle is the ability to minimize machine downtime. Downtime refers to the period that production is halted for a variety of reasons, such as to perform maintenance on equipment. According to Forbes (shorturl.at/gtWX6), unplanned downtime can cost up to $50 billion a year to manufacturers. This presents a challenge to manufacturers in finding solutions to such problems. Correspondingly, industries are embracing digital transformation and SM to develop next-generation manufacturing systems capable of overcoming faults while maintaining a continuous production line toward increased efficiency and throughput.

We describe a novel approach to fault tolerant autonomous manufacturing that can minimize the downtime of a production line through local data and domain knowledge utilization. It uses Semantic Web technologies that involve using knowledge graphs (KGs)[1] to incorporate sensor data with domain knowledge and thus integrating heterogeneous data. KGs also allow for further abstraction of data and deduction of implicit knowledge about the manufacturing line not traditionally available. This can help the system react to machine failures appropriately while maintaining production.

Semantic Web ontologies have been explored for manufacturing applications[2,3,4] alongside applications that use Internet of Things (IoT) data.[5,6,7] In this article,

1541-1672 © 2023 IEEE
10.1109/MIS.2023.3235677
Date of current version 13 February 2023.

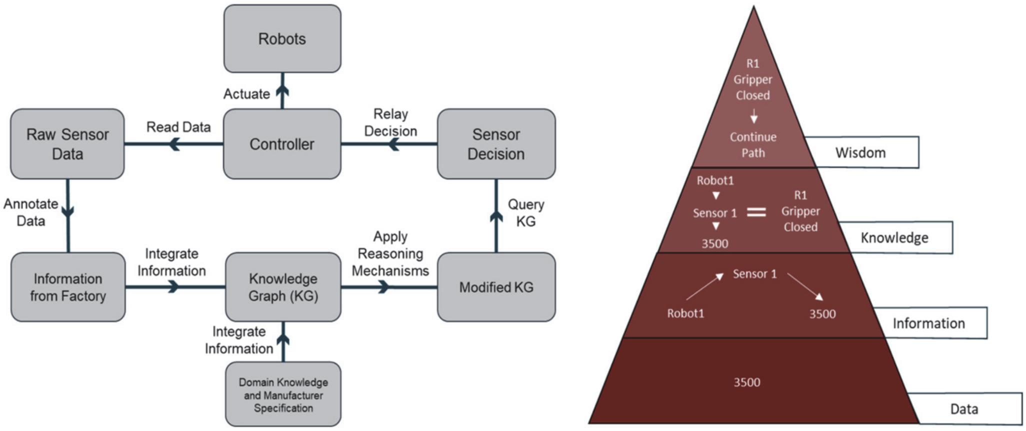


FIGURE 1. Overall process of using KGs to represent manufacturing data and domain knowledge for a simple robot pick and place operation use case.

we discuss the use of knowledge and Semantic Web in manufacturing systems that include raw data generation, semantic annotation, knowledge deduction, and decision-making (see Figure 1).

This in turn can lead to a broader impact on the manufacturing industry as it is a step toward adopting innovative technologies and standards for SM that reduce costs. Fault tolerance is also required in a broad range of manufacturing industries, such as pharmaceutical, automotive, and aerospace.

METHODOLOGY AND IMPLEMENTATION

We present the overall methodology of leveraging KGs from the representation of data for inference within a manufacturing use case. To support fault tolerance, manufacturing assets must react appropriately to failures (e.g., malfunction of a certain sensor) allowing the process to continue without any downtime for maintenance or replacement. For this to happen, assets must be aware of the different data sources available that can be used. The use case of a simple robotic pick-and-place operation illustrates the concepts. The setup consists of one industrial robotic arm and two stations for the part to be placed on. This robot interfaces with a programmable logic controller (PLC), which provides the logic for the process to move forward. When the robot picks the item up, a sensor on the robot gripper (a potentiometer) changes values indicating that the gripper has closed, and the operation can continue. The potentiometer value is read by the PLC and a signal is sent to the robot to continue with the operation. However, when the potentiometer malfunctions, the operation can continue by relying on a timer, essentially waiting a few seconds to ensure the gripper is closed before continuing with the operation.

The process begins by reading the raw sensor data from the equipment. These data are semantically annotated using a user-defined mapping derived from the semantic sensor network (SSN) ontology.[3] This information is then integrated with other information, including the domain knowledge of the facility and manufacturer specifications. Next, all this information is integrated into a central KG, on which the reasoning process is run to deduce whether the sensor is functioning properly. A corresponding triple based on the reasoning output may then be added to the KG to denote the sensors' functionality status. This additional triple links the sensor entity with a boolean determining its functionality. An SPARQL query engine then parses through the KG to project the functionality entity that was modified through reasoning. The query would return "False" if the potentiometer is malfunctioning or "True" if functioning properly. Correspondingly, the query engine will then parse through the KG again for the replacement sensor should the initial return value be "False." This decision on which sensor to use can then be relayed to the controller.

Data Modeling

We begin by modeling the sensor data to improve expressivity (see Figure 2). The information presented is modeled using the resource description framework schema where the top half of Figure 2 represents the ontology adopted, the SSN ontology. Next, we discuss how the information collected from three sources is represented in support of the use case considered.

The first source of information modeled in our use case is the data generated by the physical sensor (e.g., potentiometer). To represent these data, three

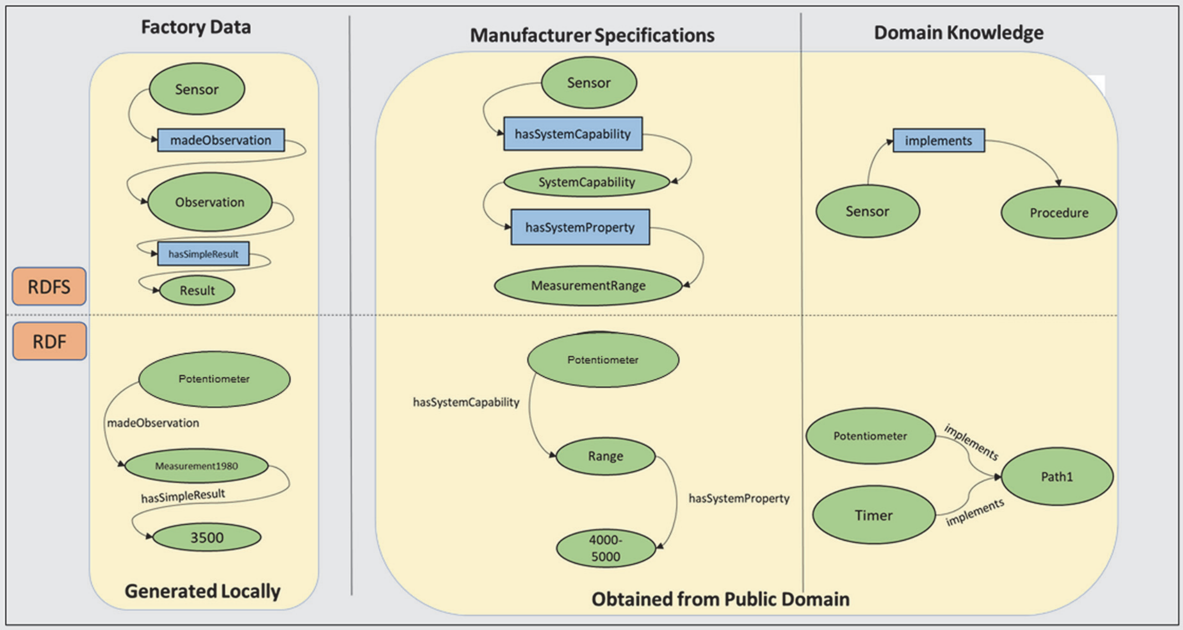


FIGURE 2. Modeling of sensor data along with domain knowledge for the use case.

entity classes, sensor, observation, and result, are defined within the schema.

1. *sosa:Sensor*: This entity represents a device that responds to a stimulus and generates a result, which relates to the potentiometer, a sensor that generates different results depending on the change in linear motion of its extrusion.
2. *sosa:Observation:* This entity represents the value of a property, which in our case is the measurement of the potentiometer. These two entities are linked together using the *sosa:madeObservation* property.
3. *sosa:Result:* This entity represents the actual value of the observation made, which is the value given by the potentiometer. This is linked to the observation using the *sosa:hasSimpleResult* property. The lower half of the figure illustrates the instances created from the described ontology for this use case.

The second source of information is the data obtained from the manufacturer. This source supplies information about the normal output range that the sensor should be yielding. This is needed as it provides the information set used to deduce the functionality of the potentiometer. The information is mapped as follows.

1. *sosa:Sensor*: Similar to the first set, this entity represents the potentiometer again and will be unified into one entity instance in the subsequent KG.
2. *ssn-system:SystemCapability:* This entity represents a property of the *sensor* entity. In this use case, this represents the range attribute of the potentiometer. This entity is mapped to the *sensor* entity through the *ssn-system:hasSystemCapability* property.
3. *ssn-system:MeasurementRange:* This entity is the set of values that a sensor can return, which is the normal output range that the manufacturer specified for the potentiometer. This entity is mapped to the abovementioned entity using the *ssn-system:hasSystemProperty* property.

Finally, the third data source is the domain knowledge about the operation. As described previously, a robot arm can use two sensors' data to continue with the operation. Either the potentiometer or a timer can provide the needed data to deduce whether the gripper is in the required state or not. This set of information is integral as it will be the basis of which the system will decide which sensor to rely on for the process to move forward. This information is mapped as follows.

1. *sosa:Sensor*: Once again this entity describes the sensors used and will be instanced twice for this source, once for the potentiometer and once for the timer.
2. *sosa:Procedure*: This entity describes a workflow, plan, or algorithm that makes a change to the state of the world. In our use case, this will be instanced

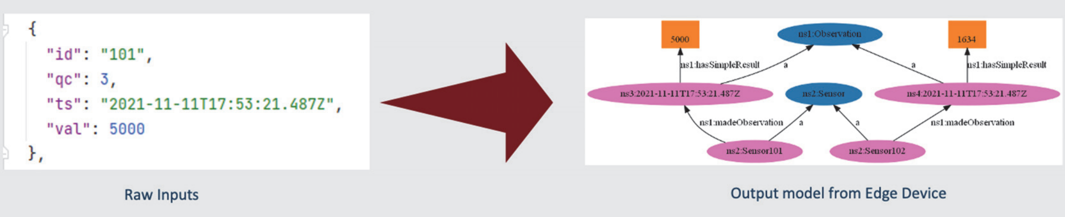


FIGURE 3. Semantic annotation of data.

as Path1, which is the path programmed on the robot to pick up the object. Path1 moves the robot from its home location to the location of the part needed to be picked up. Once the part, which is shown by the potentiometer value, is picked up, this path finishes by moving that part to a second predetermined location. This entity is linked to the sensor entity using the *ssn:implements* property.

Semantic Annotation

The translation from raw sensor data to contextualized information requires mapping the sensor data values to existing entities in the KG. To do so, we utilized SDM-RDFizer,[8] an interpreter that transforms unstructured data into RDF format. SDM-RDFizer requires the user to define the mapping that the interpreter adopts using RDF mapping language.

Figure 3 displays a snippet of the JSON object that the raw data are sent within. This object has four separate key-value pairs. The first is the unique ID of the sensor, which allows us to identify which sensor this value corresponds to. The "qc" key refers to the quality code or quality of service that determines the status of the message delivery. The "ts" key is the timestamp of the generated value and finally "val" is the actual value that the sensor is generating. The figure also shows an example of the instantiation of four triples to be included in the KG. This semantic annotation was achieved at the edge level by deploying the custom-made application onto the Siemens IPC227E Edge Device. The application reads data from the controller of the manufacturing equipment and annotates them in real time before outputting the corresponding triples to be integrated into the manufacturing KG.

Knowledge Deduction and Decision Making

With the RDF triples being generated on the edge level, the next step is to consolidate them into a central KG and perform reasoning on it. This is done on a separate machine to simulate cloud-level processing. At this level, the Jena reasoning mechanism[9] is utilized to integrate all the different information into one KG.

Reasoning was then introduced in the form of rules that allow the creation of new entities, which will be used in the decision-making process. Should the generated KG have a potentiometer value of less than that given in the manufacturer specifications, then the sensor needs changing. The visualized output KGs can be seen in Figure 4.

With the KG manipulation accomplished, the next step is to arrive at the final decision regarding completion of the required path. To iterate through the KG, SPARQL (shorturl.at/QWZ35), a query language for RDF, was used. In the query response, the status of the sensor is extracted. If the status returned was that the sensor needs changing (i.e., the "NeedChange" attribute yields "True") then the different sensors that can be used are discovered, as seen in Figure 5. The final projected result of the query statement can also be seen which reflects the decision made over which sensor to use for the operation.

DISCUSSION

In this article, the incorporation of Semantic Web technologies in a manufacturing environment was described. The presented use case showcases fault tolerance capabilities that were adopted that follow the full process of semantic annotation, KG generation, deduction of knowledge, and decision-making. On top of that, this article also provides a detailed process for semantic integration of sensor data, increasing interoperability and domain knowledge utilization in manufacturing.

Standardized Data Integration Process

The steps taken within this article to integrate the different information sources can be applied within many different domains and use cases. Even though this use case focuses mainly on one certain instance of a malfunctioning sensor, this process can be generalized to encapsulate whichever capability required. This article also showcases the applicability of undergoing real-time semantic annotation of the raw data and presents a path for

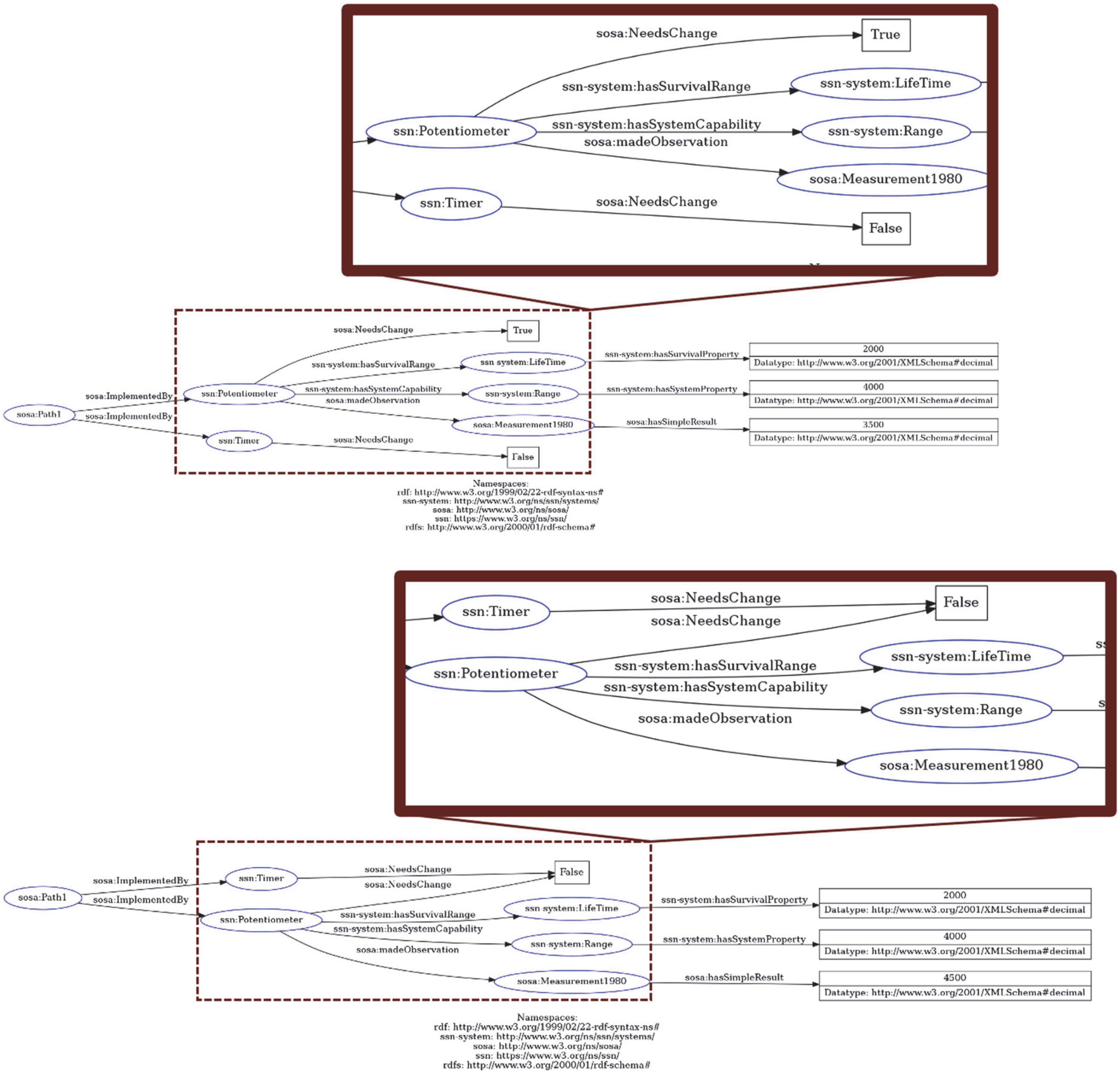


FIGURE 4. Final generated KG with functioning or malfunctioning potentiometer.

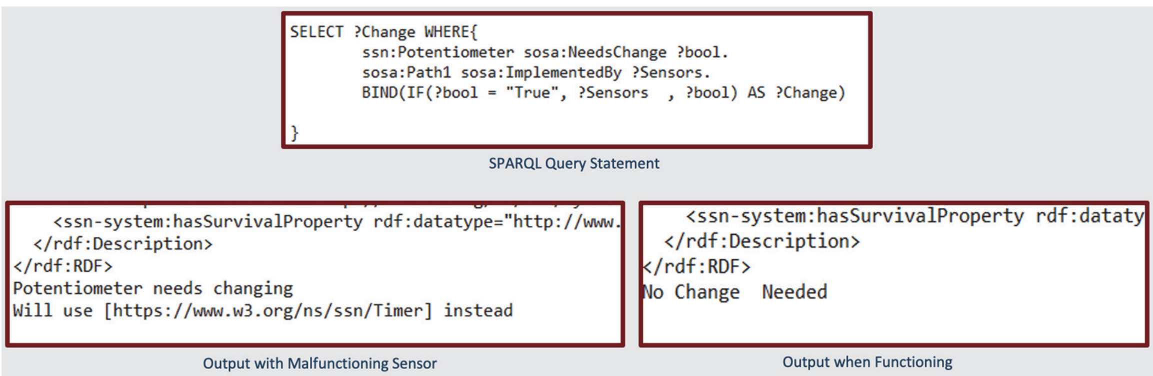


FIGURE 5. Query statement deployed on the KG with corresponding output messages.

incorporating heterogeneous data sources. With this procedure outlined alongside the technologies, further work can be undergone to integrate more information and address other issues that may occur in manufacturing.

Interoperability

With more data being generated on manufacturing shop floors using diverse devices, interoperability and integration of that data becomes a significant challenge. Semantic Web provides a step toward addressing this challenge. The use of ontologies, such as the SSN ontology and KGs, provide the critical capability for this purpose.

Domain Knowledge

KGs present a great opportunity for integrating domain knowledge for manufacturing processes. As such, datasheets and manuals can be integrated to include much of the information within the KG. This use case focused only on output range but there are endless possibilities of information that can be extracted whether it is operating or set up instruction. Having all this integrated into KGs can lead to different data accessibility and autonomous manufacturing capabilities.

CONCLUSION

SM has brought focus to the need for autonomous manufacturing. With an ever-changing market and added focus on customized production, factory floors must be agile and dynamic to adapt to diverse needs while also maintaining a consistent production schedule. In that regard, the added capability of fault tolerance can enable SM at a greater scale. This article describes how Semantic Web techniques can help in this objective.

We show that reducing downtime through fault tolerance is made possible by adopting Semantic Web for manufacturing primarily to deal with the heterogeneous data found in smart manufacturing. This article uses the simple use case of fault tolerance to illustrate the core ideas. The simplified example can be extended to support the integration of further diverse data and hence increase the knowledge acquired about the manufacturing process. The availability of these data in one KG can also allow knowledge-infused learning[10] algorithms to be deployed. Similar to recent neurosymbolic advancements for scene understanding in the autonomous driving domain,[11] this can lead to more complex event understanding in manufacturing that utilizes massive-scale heterogeneous sensor data through Semantic Web technologies.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Pankesh Patel for the helpful discussions regarding Semantics within IoT and use case adoption. This work was funded by NSF Award 2119654 "RII Track 2 FEC: Enabling Factory to Factory (F2F) Networking for Future Manufacturing" and "Enabling Factory to Factory (F2F) Networking for Future Manufacturing across South Carolina," funded by South Carolina Research Authority (SCRA). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding organizations.

REFERENCES

1. A. Sheth, S. Padhee, and A. Gyrard, "Knowledge graphs and knowledge networks: The story in brief," *IEEE Internet Comput.*, vol. 23, no. 4, pp. 67–75, Jul./Aug. 2019, doi: 10.1109/MIC.2019.2928449.
2. Y. Lu, H. Wang, and X. Xu, "ManuService ontology: A product data model for service-oriented business interactions in a cloud manufacturing environment," *J. Intell. Manuf.*, vol. 30, no. 1, pp. 317–334, Jan. 2019, doi: 10.1007/S10845-016-1250-X/FIGURES/15.
3. M. Compton et al., "The SSN ontology of the W3C semantic sensor network incubator group," *J. Web Semantics*, vol. 17, pp. 25–32, Dec. 2012, doi: 10.1016/J.WEBSEM.2012.05.003.
4. M. Dinar and D. W. Rosen, "A design for additive manufacturing ontology," *J. Comput. Inf. Sci. Eng.*, vol. 17, no. 2, Jun. 2017, Art. no. 021013, doi: 10.1115/1.4035787/446471.
5. C. Henson, A. Sheth, and K. Thirunarayan, "Semantic perception: Converting sensory observations to abstractions," *IEEE Internet Comput.*, vol. 16, no. 2, pp. 26–34, Mar./Apr. 2012, doi: 10.1109/MIC.2012.20.
6. P. Desai, A. Sheth, and P. Anantharam, "Semantic gateway as a service architecture for IoT interoperability," in *Proc. IEEE 3rd Int. Conf. Mobile Serv.*, 2015, pp. 313–319, doi: 10.1109/MOBSERV.2015.51.
7. P. Patel, A. Gyrard, D. Thakker, A. Sheth, and M. Serrano, "SWoTSuite: A development framework for prototyping cross-domain semantic web of things applications," in *Proc. 15th Int. Semantic Web Conf.*, 2016, doi: 10.48550/arxiv.1609.09014.
8. E. Iglesias, S. Jozashoori, D. Chaves-Fraga, D. Collarana, and M.-E. Vidal, "SDM-RDFizer," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, 2020, pp. 3039–3046, doi: 10.1145/3340531.3412881.

9. A. Ameen, K. U. R. Khan, and B. P. Rani, “Reasoning in semantic web using Jena,” *Comput. Eng. Intell. Syst.*, vol. 5, no. 4, pp. 39–47, 2014. Accessed: Jul. 07, 2022. [Online]. Available: www.iiste.org
10. A. Sheth, M. Gaur, U. Kursuncu, and R. Wickramarachchi, “Shades of knowledge-infused learning for enhancing deep learning,” *IEEE Internet Comput.*, vol. 23, no. 6, pp. 54–63, Nov./Dec. 2019, doi: 10.1109/MIC.2019.2960071.
11. R. Wickramarachchi, C. Henson, and A. Sheth, “Knowledge-based entity prediction for improved machine perception in autonomous systems,” *IEEE Intell. Syst.*, vol. 37, no. 5, pp. 42–49, Sep./Oct. 2022, doi: 10.1109/MIS.2022.3181015.

FADI EL KALACH is a Ph.D. student at the University of South Carolina, Columbia, SC, 29208, USA. His research focuses on smart manufacturing and artificial intelligence. His master’s thesis focused on integrating Semantic Web into manufacturing for increased fault tolerance capabilities. Contact him at elkalach@email.sc.edu.

RUWAN WICKRAMARACHCHI is a Ph.D. student at the AI Institute, University of South Carolina, Columbia, SC, 29208, USA. His dissertation research focuses on introducing novel neuro-symbolic AI techniques to improve machine perception and context understanding in autonomous systems (e.g., autonomous driving and smart manufacturing). Contact him at ruwan@email.sc.edu.

RAMY HARIK is a professor in the Department of Mechanical Engineering, University of South Carolina, Columbia, SC, 29208, USA. He is also the principal investigator of the neXt (new and emerging X technologies) Research Team. He is a Fulbright Scholar. Contact him at harik@mailbox.sc.edu.

AMIT SHETH is the founding director of the AI Institute, University of South Carolina, Columbia, SC, 29208, USA. His current core research is in knowledge-infused learning and explanation, and transnational research includes personalized and public health, social good, education, and future manufacturing. He is a fellow of IEEE, AAAI, AAAS, and ACM. Contact him at amit@sc.edu.

EDITOR: Amit Sheth, amit@sc.edu

DEPARTMENT: KNOWLEDGE GRAPHS

This article originally appeared in Internet Computing vol. 26, no. 6, 2022

*i*MetaverseKG: *I*ndustrial Metaverse Knowledge Graph to Promote Interoperability in Design and Engineering Applications

Utkarshani Jaimini, *University of South Carolina, Columbia, SC, 29208, USA*
Tongtao Zhang and Georgia Olympia Brikis, *Siemens Corporation, Technology, Princeton, NJ, 08540, USA*
Amit Sheth, *University of South Carolina, Columbia, SC, 29208, USA*

The term metaverse was coined by author Neal Stephenson in 1992 in his science fiction novel "Snow Crash."[1] Metaverse is a conjunction of the Greek prefix "meta," which means beyond, and the stem "verse," which implies universe, hence the meaning "beyond the universe." It is a futuristic, hyperrealistic virtual world where humans will spend time performing their day-to-day activities, such as entertaining, socializing, playing, working, and shopping. This requires that a metaverse offers a real-time virtual representation of the physical world with its entities, relationships, events, states, processes, and activities. According to the Gartner forecast report, the metaverse is among the top five emerging trends and technologies. Gartner predicts that by 2026 25% of people will spend at least one hour every day in the metaverse and 30% of organizations will have products and services developed for metaverse platforms.[2] The metaverse is in an early developmental stage but has a considerable promise of occupying prominent space in the next phase of the Internet.

Today, Neal Stephenson's vision of a virtual world is becoming a reality across various sectors, with early popularity in gaming,[3] consumer products,[4] and enterprise collaboration.[5] Emerging metaverse platforms can be divided into two main categories: social metaverse and industrial metaverse (*i*Metaverse) platforms. Social metaverse deals with a network of virtual worlds focused on social connections, entertainment, and commerce, such as the metaverse pursued by meta.[4] The *i*Metaverse, on the other hand, integrates information and communication technologies related to industrial processes creating rich digital twins that blend real-world elements contextualized with meaningful digital data. While the use of technologies, such as virtual and augmented reality, digital twins, and artificial intelligence (AI) is opening new ways to aggregate and display information for industrial decision making, collaboration, and keeping processes running at peak efficiency, some key challenges remain to be solved to make the vision of an *i*Metaverse reality.

> *"WE PROPOSE TO USE KG TECHNOLOGIES TO SUPPORT INTEROPERABILITY AND SEAMLESS TRANSFER OF INFORMATION ACROSS IMETAVERSE PLATFORMS. WE DEVELOPED AN IMETAVERSEKG BASED ON AN EXEMPLARY USE CASE FROM DESIGN AND ENGINEERING APPLICATION."*

1089-7801 © 2022 IEEE
Digital Object Identifier 10.1109/MIC.2022.3212085
Date of current version 23 December 2022.

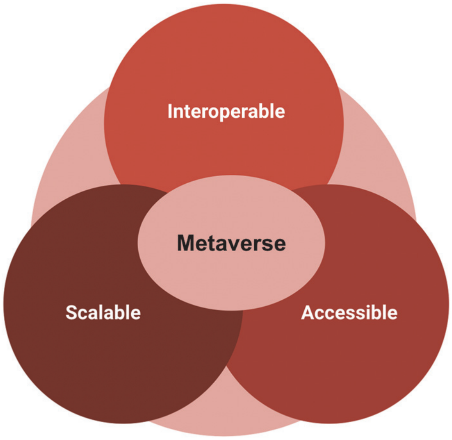


FIGURE 1. Characteristics of metaverse: interoperable, scalable, and accessible.

Today many industrial processes, e.g., design and engineering, testing, validation, manufacturing, or operations, are taking place in their silos with little information exchange across different processes. To build *i*Metaverse environments that facilitate use cases from these domains, we need to assure interoperability across platforms and software tools used today. We propose to use knowledge graph (KG) technologies to support interoperability and the seamless transfer of information across *i*Metaverse platforms. We developed an industrial metaverse knowledge graph (*i*MetaverseKG) based on an exemplary use case from design and engineering application. In this article, we present the characteristics of the metaverse, *i*Metaverse and its challenges, and the *i*MetaverseKG for an exemplary *i*Metaverse application from the design and engineering domain.

We propose to use KG technologies to support interoperability and seamless transfer of information across *i*Metaverse platforms. We developed an *i*MetaverseKG based on an exemplary use case from design and engineering application.

METAVERSE CHARACTERISTICS

Metaverse platforms are characterized by interoperability, scalability, and accessibility (see Figure 1).

Interoperability

Metaverse environments keep the records of virtual items by a user enabling seamless transfer of items across different virtual environments. A virtual item brought by a user into one virtual environment can be easily transferred and used in another shared virtual environment within a metaverse, allowing interoperability across environments. Unlike the physical world, there is no wall between virtual worlds.

Scalability

The metaverse allows several virtual avatars to coexist on the platform. There is no restriction on the number of virtual items in the metaverse. Real-time events hosted in the metaverse allow unlimited audiences. The users can attend virtual events from anywhere in the world hosted on a server without sharing their personal servers. This provides user interaction across various shared virtual environments. The scalability characteristic of the metaverse is hugely beneficial for sports events, live concerts, educational institutes, and collaboration with rich online content libraries.

Accessible

The metaverse does not have many access restrictions and allows accessibility of live reality in real time. The metaverse may not have a hardware restriction and can be accessible to all devices. The virtual environments in the metaverse have no cost restrictions, no building codes, and no development glitch to consider. The companies, artists, and influencers can create their virtual limited-edition exclusive virtual assets for their users. The technology platforms to create interactive content are evolving rapidly. It has endless virtual opportunities for everyone. The metaverse is recognized as a workplace where people can invest, buy, and lease their virtual assets.

INDUSTRIAL METAVERSE

The *i*Metaverse aims to promote the development of manufacturing and services for the industry. It can simulate the physical reality of interest to the industry, such as the product cycle, collaboratively designing a product, working with a client, visualizing the product, and optimizing the design. Typical drivers are to achieve better efficiency, discovery, and innovation. The process saves computation, material cost, product development, and client feedback time and meets market needs faster. Manufacturing companies, such as BMW, use *i*Metaverse-based augmented virtual reality to advance the design and prototyping of a new product.[6] The product design involves visualizing and interacting with the design in the *i*Metaverse environment. It allows simultaneous design review, updates, and assembly checks in real time before putting the design into production. The engineers can visualize their computer aided design (CAD) model and work collaboratively with the customers and other teams across different locations. The *i*Metaverse aims at lowering the production costs and improving time to market.

The *i*Metaverse has the following five key facets, as shown in Figure 2.

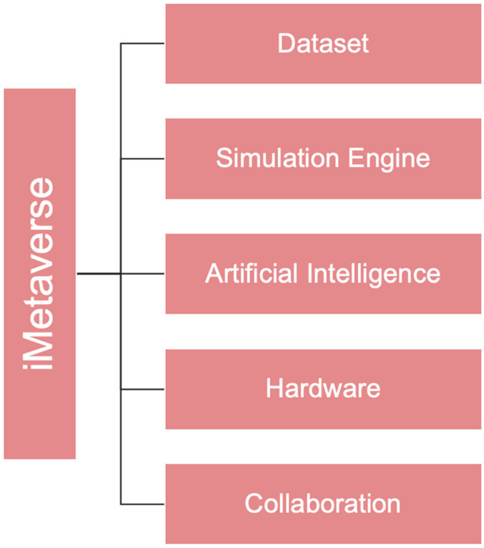


FIGURE 2. Key facets of *i*Metaverse: Dataset, simulation engine, AI, hardware, collaboration.

- Dataset—information from Internet of things (IoTs) devices, enterprise, Internet, human social interactions, etc.
- Simulation engine—provides the rendering and computing capabilities to support the physical world and real-time interactivity in the *i*Metaverse.
- Artificial intelligence—the AI and big data technologies provide perception and cognitive capabilities, such as learning, creating, and optimization.
- Hardware—intelligent interactive devices with capabilities of real-time integration of physical and virtual world, such as IoTs, new materials, general purpose hardware, and robots.
- Collaboration—interoperability, openness, and compatibility are the core elements for providing a better user experience in the *i*Metaverse. The intercompany and interindustry collaboration will increase *i*Metaverse content and value, promoting an open and collaborative ecosystem for *i*Metaverse development.

CHALLENGES

Just like HTML protocols created a unified framework for web pages on the Internet, enabling browsing the Internet from any device and browser. The operating systems, such as android, iOS regardless of having different devices, can still connect and exchange information among themselves. The software and application developers can implement their products on both platforms with minimal effort. The *i*Metaverse devices and platforms need to deliver such capabilities to users and content providers alike. The *i*Metaverse is in its initial stages. The biggest challenge it faces is interoperability between platforms.[7,8,14,15] Currently, there is no developmental standard in the metaverse. As a result, the information, and data necessary to model the real world in a virtual environment are scattered across multiple platforms leading to different platforms requiring different hardware and software. An *i*Metaverse can be created with various platforms, including NX, AutoCAD, Simcenter STAR-CCM+, Plant Simulation, Unity, Autodesk, and the Nvidia Omniverse. These platforms often use different data models and terminologies to describe the same thing, making it challenging to connect data even if it has been aggregated in a single data file. In addition, there is a need for real-time interactivity synchronization across different platforms. The edits made in one platform or product cycle should seamlessly communicate or transfer to the other platforms. As depicted in the left side of Figure. 3, present, describes the current single tool-to-tool interactions, which lacks scalability and is a time-consuming process, requiring teams of diverse technical backgrounds to work together closely and involves several iterations of the tool before production ready. On the contrary, the right side of Figure. 3, future, describes the proposed *i*MetaverseKG for interoperability and scalability where adding data and tools from a new tool is a seamless integration process, which simply requires adding a KG link to the new tool's data representation.

In the following sections, we will drill down the interoperability issue and its solution using KGs in the engineering and design applications for the *i*Metaverse.

*I*METAVERSEKG: *I*METAVERSE KNOWLEDGE GRAPH

A KG captures the complexity of the real world and stores information intuitively. KGs are more flexible than traditional relational data models making it easy to add, edit, and remove data to respond to real-world changes. The current *i*Metaverse description is semantic free, solely relying on the geometric artifacts of the virtual environment. We propose an *i*MetaverseKG, a semantic network of meta-data entities and their relationships, which enables seamless representation from the physical world into the virtual *i*Metaverse. It functions as a connective platform between diverse systems, promoting interoperability, and fusing data across different tools. It connects

METAVERSE: UNADDRESSED CONCERNS

Like other technologies, the metaverse is no exception to issues of abuse, security, trust, privacy, regularization, human interaction, etc. The laws of the physical world do not span to the metaverse. Digital avatars in the metaverse environment could be a victim of not physical but mental and emotional trauma due to online abuse. There is no legal framework to deal with digital misconduct and crimes. Content creators need licensing and usage boundaries to create avatars that create commercial value. There would be disputes regarding the metaverse's commercial value and intellectual property.[1] Grungo Colarulo is a startup that deals with laws and regulations about digital avatars in the metaverse. It proposes an AI-driven digital lawyer in virtual space, providing services, such as resolving digital land disputes, virtual legal firms, and company incorporation.[2]

The responsible development of the metaverse cannot be done single-handedly by one company. The safe development of metaverse requires the close collaboration of developers, industry partners, experts, and policymakers. Each metaverse platform requires configuration of its own and lacks a standard development format. Metaverse standard forum, a nonprofit organization, with 35 founding members and over 1500 partners from industries and academic institutions, are working together to develop standards for building an open and inclusive metaverse. EVERFI, an education company, designed a metaverse digital literacy curriculum, "Get Digital: Safety in the Metaverse," to empower students with the digital citizenship skills needed for safe metaverse usage.[2] In addition, Australian National University and professor Genevieve Bell are working together to develop an approach to "system by design," which industries can adopt for metaverse development.[2] Another example domain is music, where HiiWAV invites black musicians to equip the black, indigenous, and people of color with skills needed for metaverse development.[2] Also, youth organization, such as Youth Bureau, brings together technology companies, youth, and policymakers in the early development of metaverse and societal response for extended reality technologies with the inclusion of youth, promoting early engagement, and safe and inclusive development of technology.[2]

The KG has successfully aided in the issues of safety, toxicity, and human interaction in a community. For example, in the past, KGs have been used for 1) detecting hate speech and toxic content on the Internet, which often requires understanding the background knowledge, context, language, etc., 2) community detection using the interactions and characteristics, 3) detecting software security entities, such as software weakness, attack pattern, vulnerabilities, etc., which are scattered across various documents, 4) processing and analyzing cybersecurity data with fragmented multiple threat data and industrial network framework to identity cybersecurity entities and relations between them.[3,4,5,6,7,8] Metaverse development is a continuously evolving space with opportunities for improvement for its safe usage.

REFERENCES

1. N. Kumar, "Council post: six unaddressed legal concerns for the metaverse," Forbes, Forbes Magazine, 14 Apr. 2022. [Online]. Available: https://www.forbes.com/sites/forbestechcouncil/2022/02/17/six-unaddressed-legal-concerns-for-the-metaverse/?sh=2993fad07a94
2. "Building the Metaverse Responsibly." *Meta*, Sep. 15, 2022. [Online]. Available: https://about.fb.com/news/2021/09/building-the-metaverse-responsibly/
3. A. Sheth, V. L. Shalin, and U. Kursuncu, "Defining and detecting toxicity on social media: context and knowledge are key," *Neurocomputing*, vol. 490, pp. 312–318, 2022.
4. P. R. Lobo, E. Daga, and H. Alani, "Supporting Online Toxicity Detection with Knowledge Graphs," in *Proc. Int. AAAI Conf. Web Social Media*, vol. 16. 2022.
5. H. Xiao, Z. Xing, X. Li, and H. Guo, "Embedding and predicting software security entity relationships: A knowledge graph based approach," in *Proc. Int. Conf. Neural Inf. Process.*, 2019, pp. 50–63.
6. G. Shen, W. Wang, Q. Mu, Y. Pu, Y. Qin, and M. Yu, "Data-driven cybersecurity knowledge graph construction for industrial control system security," in *Proc. Wireless Communications Mobile Computing*, 2020.
7. S. Bhatt, S. Padhee, A. Sheth, K. Chen, V. Shalin, D. Doran, and B. Minnery, "Knowledge graph enhanced community detection and characterization," in *Proc. 12th ACM Int. Conf. Web Search Data Mining*, 2019, pp. 51–59.
8. A. Sheth, M. Gaur, K. Roy, R. Venkataraman, and V. Khandelwal, "Process Knowledge-Infused AI: Toward User-Level Explainability, Interpretability, and Safety," *IEEE Internet Computing*, vol. 26, no. 5, pp. 76–84, 2022.

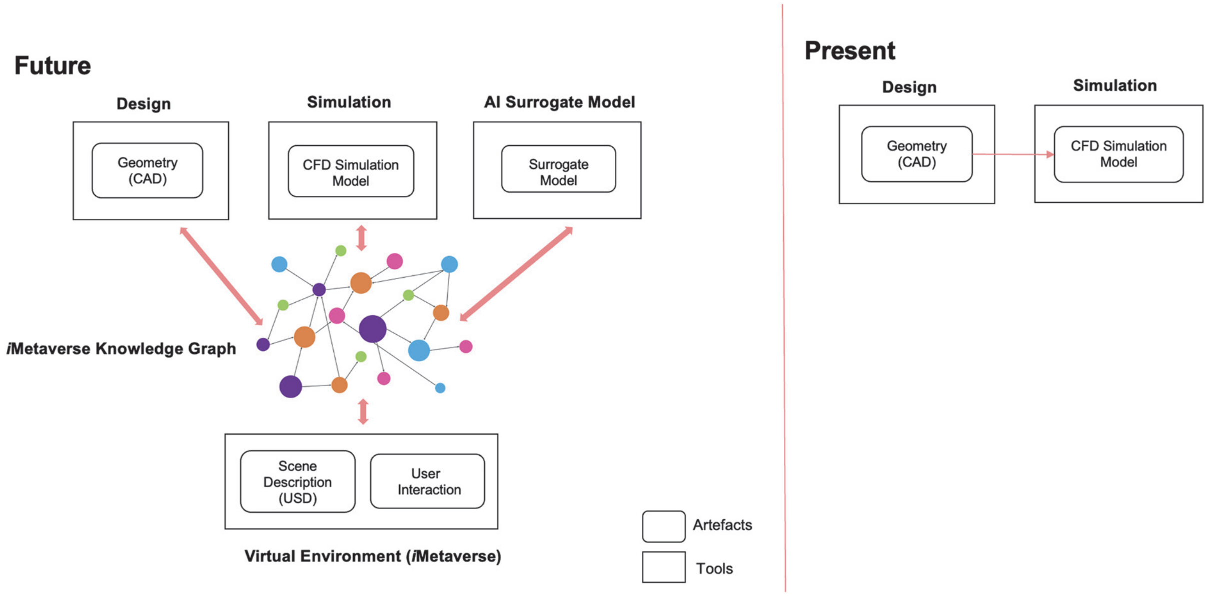


FIGURE 3. Shift from the current engineering and design approach to the future approach for the use of *i*Metaverse incorporating *i*MetaverseKG. (Abbreviation: CAD—Computer aided design, CFD—Computational fluid dynamics, AI—Artificial intelligence, USD—Universal scene description).

these platforms, standardizing concepts at a semantic level so they can be applied across the *i*Metaverse development pipeline.

The *i*MetaverseKG 1) describes the relationships and semantics of *i*Metaverse entities and tools, 2) bridges the gaps across *i*Metaverse platforms, 3) standardizes the concepts that can be applied across the *i*Metaverse development pipeline, and d) promotes the reusability of metaverse entities in a new platform or environment. The seamless integration of semantic information in *i*MetaverseKG enables synchronization across platforms. The updates made by user interactions or design changes do not require a new design and development iteration. The updates can be seamlessly integrated into the design steps by running a few update queries to the *i*MetaverseKG. The updates can be transferred, and the user interaction or the design changes can be assimilated according to the new updates.

We will discuss the use case of the *i*MetaverseKG in the *i*Metaverse for designing a computational fluid dynamics (CFD) simulation of a car in a wind tunnel.

ENGINEERING AND DESIGN IN *I*METAVERSE

CFD is a branch of fluid mechanics that solves problems involving fluid flows using numerical analysis and data structures. It solves engineering problems in applications, such as aerodynamics, weather simulation, industrial system design and analysis, fluid flow, and heat transfer. It simulates free-stream flow and the interaction of the fluid with the surfaces (defined by boundary conditions). The forces acting on the fluid body and the changes in the internal fields, such as pressure, velocity, density, and temperature can be mathematically expressed using Navier Stokes differential equations.

The CFD simulation process involves formulating the flow problem; modeling the geometry and flow domain in which the flow will be simulated using CAD; establishing the boundary and initial conditions of the system; generating a discretized grid of the flow domain; establish the simulation strategy, such as choice of turbulence model, and algorithms; establish the input parameters; perform the simulation with iterative, batch or distributed processing; monitor the simulation for convergence; postprocess the simulation to get the desired flow properties; compare the computed results from experimental computational, or analytical studies to establish the validity of the computed results; and repeat the process to understand the differences in the accuracy and performance of the computed results.[10] The CFD simulations often take 10+ hours to solve the differential equations on GPU clusters.

The time complexity of the CFD simulation can be overcome by explicitly exploiting the physics underlying the fluid flow at a given time and space. This

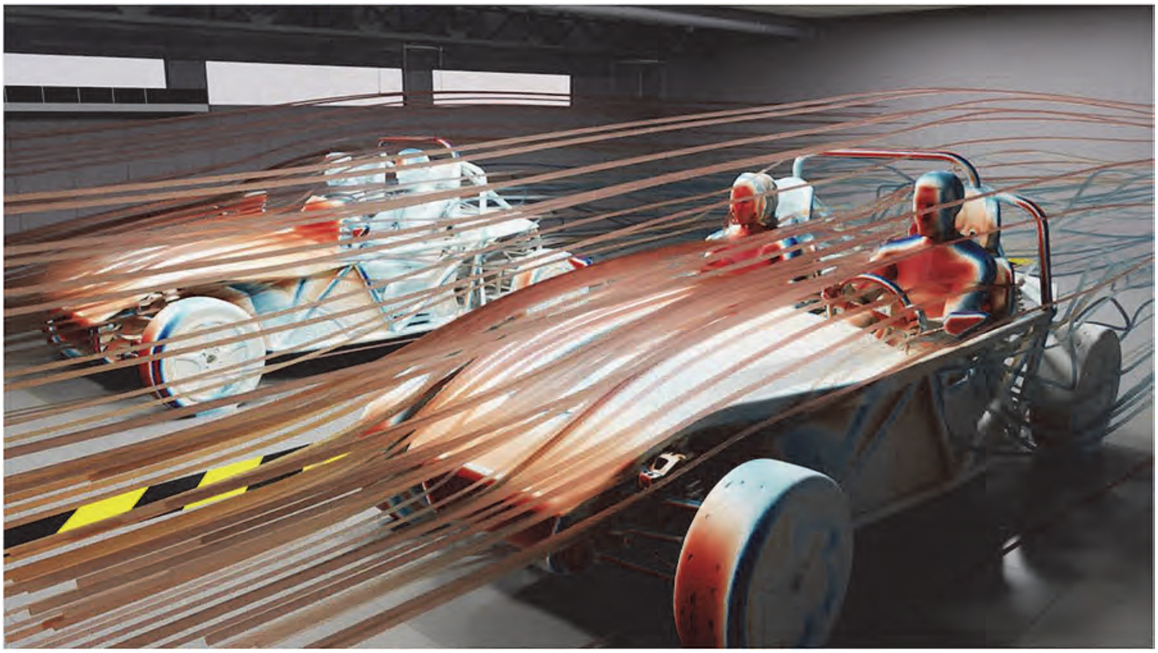


FIGURE 4. Exemplary visualization of velocity and pressure fields from 3-D steady state CFD simulation done in Simcenter STAR-CCM+ around two exemplary car configurations and its passengers, rendered using Nvidia Omniverse.[9]

technique is known as AI-Surrogate models, which utilizes physics inspired neural networks.[11,12,13]

We design a CFD simulation of aerodynamics flow around the car in the *i*Metaverse. The design steps includes 1) designing the geometry of the car using the CAD, and universal scene description (USD) framework by Pixar. The USD framework facilitates the visualization of the car in the *i*Metaverse, 2) running the CFD simulations using the state-of-the-art simulation tools, such as Simcenter, 3) running the AI-surrogate model-based CFD simulation for near real-time simulation, and 4) visualization in the *i*Metaverse. Figure 4 shows the metaverse visualization of testing of the CFD simulation of two different designs (geometry) of the car in the wind tunnel. It shows the effect of the wind on the passengers in the cars with varying configurations, with windshield (see Figure 4, car at the back), and without windshield (see Figure 4, car in the front). The *i*MetaverseKG captures the semantics, and the metadata entities incorporated in the design steps mentioned previously leading to better interoperability and reusability. It provides a platform for the standardization of concepts in the *i*Metaverse development. The *i*MetaverseKG (a representative subset is shown in Figure 5) gathers the meta-data entities included in designing the geometry of the car in CAD for CFD simulation, boundary conditions for CFD for the simulation and surrogate models, and entities capturing the geometry and visualization of the CFD in the *i*Metaverse to be visualized using a USD framework for scene visualization.

The *i*MetaverseKG also defines a metaverse instance, which has an instance of a car. A car in the wind tunnel, as shown in Figure 4, can be represented by both CAD and USD geometry. The CAD geometry for CFD simulation and USD for *i*Metaverse visualization. The USD geometry is derived from the CAD geometry. The CFD simulation models take CAD geometry as input. The *i*MetaverseKG also captures the boundary conditions used for simulation and surrogate models, ranging from 0 degree to 180 degree of wind direction, and 0 to 80 km/hr of wind speed for the example use case. Figure 5 considers the boundary condition of 30 degree wind direction and 20 km/hr wind speed for CFD simulation, and 0 degree wind direction and 60 km/hr wind speed for AI surrogate for a given instance of car with a particular CAD and USD representation.

An instance of a car can be subjected to different boundary conditions resulting in multiple CFD simulations and surrogate models (Figure 6 shows an example use case for a single wind speed and wind direction). The *i*MetaverseKG captures the abovementioned meta-data from different tools and provides a platform for interoperability, synchronization, and standardization. The real-time visualization of the aerodynamics flow in the *i*Metaverse requires fetching the geometry of the car, CFD simulation, and AI-surrogate models for each boundary condition,

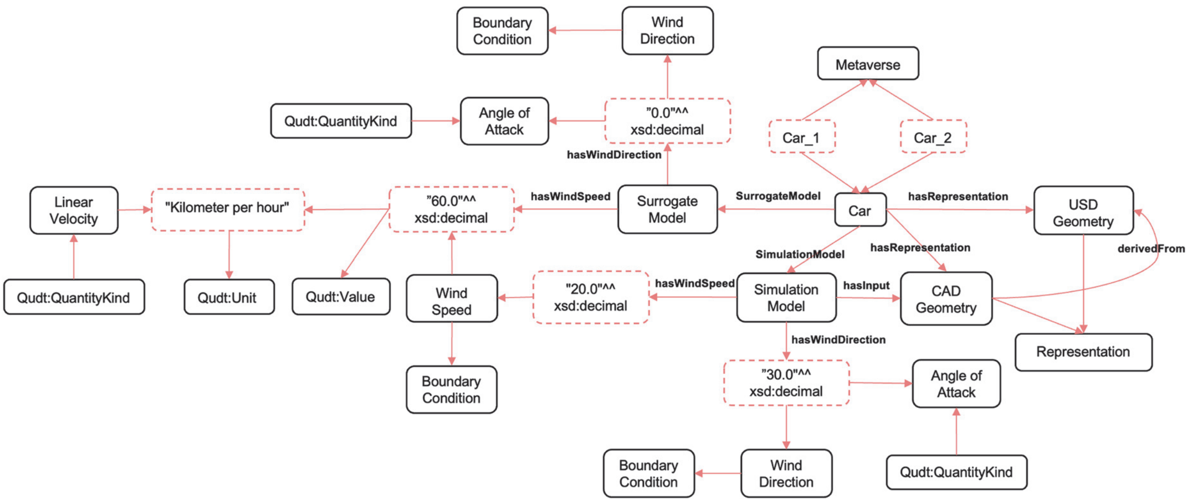


FIGURE 5. Snapshot of an *i*MetaverseKG focused on a CFD simulation of a car in the *i*Metaverse. The *i*MetaverseKG provides a standardization platform, which captures the information about the design steps along with the meta-data entities describing the geometry (design) of the car, and the surrogate and simulation models used in the CFD.

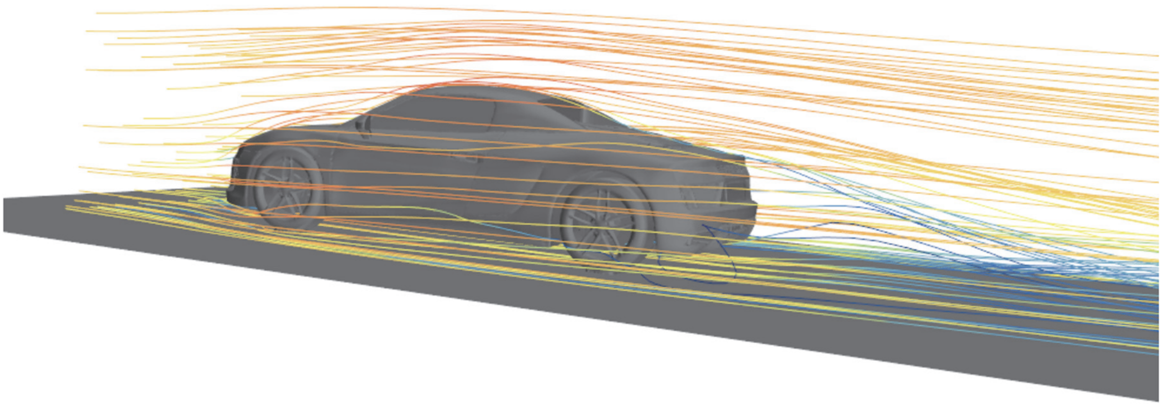


FIGURE 6. Instance of a car with a given geometry representation an *i*Metaverse for a single boundary condition with a given wind speed and wind direction.

which have been stored in the *i*MetaverseKG. The *i*MetaverseKG promotes reusability of the instance of the car, as shown in Figure. 5, which can be modified as well as reused in a new *i*Metaverse environment. It can be used to study the difference of aerodynamics around the car in the tunnel, as shown in Figure 5, as compared to the car on a track, as shown in Figure 6. The *i*MetaverseKG stores the semantic scene description about the car, tunnel, and the track, which otherwise is missing in the current development pipeline. It can also assist in studying the effect of different material properties of the car on the aerodynamic flow, such as changing the car material from metal to ceramic. One can modify and visualize the different car geometry and aerodynamic simulations for a given boundary condition in real time in the *i*Metaverse. Modifying wind direction, and wind speed using the *i*MetaverseKG simply requires an update, synchronization, and retrieval query on the KG to fetch the appropriate model to run and visualize in the *i*Metaverse, which otherwise be time consuming requiring running a new simulation, and surrogate models and development of multiple tool-to-tool interactions from the CAD, USD, CFD simulations, and AI-surrogate model.

CONCLUSION

The issue of interoperability across metaverse platforms has been well recognized by the community,

ROLE OF KG IN STANDARDIZATION

KG has been widely used for standardization in the industrial applications ranging from finance, transportation, healthcare, education, remote operation and maintenance of equipment, intelligent computer-aided diagnosis, treatment, and so on. KGs have achieved success in auxiliary design, equipment operation and maintenance, supply chain management, etc. It helps represent information in a human understandable form, which enhances the ability to manage, organize, and understand the large amount of information. In 2019, the IEEE Knowledge Graph Working Group initiated projects IEEE P2807 and P2807.1. The IEEE P2807 standard project “Framework of Knowledge Graph” describes the requirement for input, construction process, performance metrics, applications, and infrastructure. The IEEE P2807.1 standard project, "Standard for Technical Requirements and Evaluating Knowledge Graphs," establish technical requirements, performance metrics, evaluation criteria, and test cases for KG. The test cases include input data, meta-data, data extraction, data fusion, storage and retrieval, inference and analysis, and KG visualization.[1] In the healthcare, KG integrates the heterogeneous information from textual medical knowledge from biomedical literature, with the healthcare information system. The healthcare system has a large amount of textual medical knowledge and patient healthcare information from clinician notes, which plays an important role in knowledge delivery and decision support to both clinicians and patients. With the growing amount of this information at a rapid pace it faces the challenge to organize, integrate, analyze, and deliver information in an efficient manner. The KG-based standardization framework for the abovementioned data facilitates faster query, update, and reasoning, as well as supports human and machine interpretability.[2] In the intelligent manufacturing, an enterprise KG integrates and systematize business data from different departments (such as market monitoring system, logistics traceability system, inventory management system, order management system, human resources management system, maintenance management system, and operation management system), and expert knowledge in different areas (such as fault repair, market forecasting, sales violation, and production scheduling), which leads to better decision making. Furthermore, the knowledge also assists in anomalies traceability in the production line by tracing back the corresponding node and analyzing the subgraph for the issues and possible solutions.[3]

REFERENCES

1. S. Wei, Y. Ma, R. Li, and L. Hu, “Toward Smart Manufacturing: Key Technologies and Trends Driving Standardization,” *Computer*, vol. 53, no. 4, pp. 46–50, Apr. 2020, doi: 10.1109/MC.2020.2970821.
2. L. Shi, S. Li, X. Yang, J. Qi, G. Pan, and B. Zhou, “Semantic health knowledge graph: semantic integration of heterogeneous medical knowledge and services,” *BioMed Res. Int.*, 2017.
3. R. Li, S. Wei, and J. Li, “Study on the application framework and standardization demands of AI in intelligent manufacturing,” in *Proc. IEEE Int. Conf. Artif. Intelligence Adv. Manuf.*, 2019, pp. 604–607.

such as W3C and Metaverse Standard Forum.[7] The *i*MetaverseKG provides an ability for interoperability and integration across platforms in industrial engineering and design applications. The *i*MetaverseKG has an advantage over the existing tool-to-tool integration platforms for solving the interoperability issue. Furthermore, the current tool-to-tool integration development lacks scalability, whereas adding data from a new tool to *i*MetaverseKG is a seamless integration process requiring KG links to the new tool’s data representation. Although the *i*MetaverseKG is still in its early stage, it has enormous potential to address the issue of interoperability and promote reusability in *i*Metaverse development.

ACKNOWLEDGMENTS

This work was performed during a summer internship at Siemens Corporation, Technology in Princeton, NJ, U.S. The data used in this work was provided by Siemens Digital Industries Software. The SimRod car is used as a technology carrier to demonstrate digital twin technologies. We would also like to acknowledge National Science Foundation (NSF) Award #2133842 “EAGER: Advancing Neuro-symbolic AI with Deep Knowledge- infused Learning,” in part for the support. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the company and funded agency.

REFERENCES

1. N. Stephenson, *Snow Crash: A Novel*. New York, NY, USA: Random House, 2017.
2. "Gartner forecast." Accessed: Oct. 13, 2022. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2022-02-07-gartner-predicts-25-percent-of-people-will-spend-at-least-one-hour-per-day-in-the-metaverse-by-2026
3. "The lego group and epic games team up to build a place for kids to play in the metaverse," Epic Games. Accessed: Oct. 13, 2022. [Online]. Available: https://www.epicgames.com/site/en-US/news/the-lego-group-and-epic-games-team-up-to-build-a-place-for-kids-to-play-in-the-metaverse
4. "Meta," Digital Connection in the Metaverse, Meta. Accessed: Oct. 13, 2022. [Online]. Available: https://about.meta.com/metaverse/
5. "The metaverse is coming. Here are the cornerstones for securing it," The Official Microsoft Blog, Mar. 29, 2022. [Online]. Available: https://blogs.microsoft.com/blog/2022/03/28/the-metaverse-is-coming-here-are-the-cornerstones-for-securing-it/
6. B. Caulfield, "Nvidia, BMW blend reality, virtual worlds to demonstrate factory of the future," *NVIDIA Blog*, Apr. 28, 2022. Accessed: Oct. 13, 2022. [Online]. Available: https://blogs.nvidia.com/blog/2021/04/13/nvidia-bmw-factory-future/
7. N. Trevett et al., "The metaverse standards forum," Sep. 15, 2022. Accessed: Sep. 27, 2022. [Online]. Available: https://metaverse-standards.org/
8. S. Seidel, N. Berente, J. Nickerson, and G. Yepes, "Designing the metaverse," in *Proc. HICSS*, 2022, pp. 1–10.
9. G. O. Brikis, "The hidden potential of physics-informed AI: Ingenuity: Siemens." Accessed: Oct. 13, 2022. [Online]. Available: https://ingenuity.siemens.com/2022/08/the-hidden-potential-of-physics-informed-ai/
10. CFD analysis process. (n.d.). Accessed: Sep. 27, 2022. [Online]. Available: https://www.grc.nasa.gov/www/wind/valid/tutorial/process.html
11. N. Zobeiry and K. D. Humfeld, "A physics-informed machine learning approach for solving heat transfer equation in advanced manufacturing and engineering applications," *Eng. Appl. Artif. Intell.*, vol. 101, 2021, Art. no. 104232.
12. L. Sun, H. Gao, S. Pan, and J. X. Wang, "Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data," *Comput. Methods Appl. Mechanics Eng.*, vol. 361, 2020, Art. no. 112732.
13. T. Zhang, B. Dey, K. Veeraraghavan, H. Kulkarni, and A. Chakraborty, "Demystifying the data need of ML-surrogates for CFD simulations," 2022, *arXiv:2205.08355*.
14. S. Van der Land, A. Schouten, and F. Feldberg, "Modeling the metaverse: A theoretical model of effective team collaboration in 3D virtual environments," *J. Virtual Worlds Res.*, vol. 4, no. 3, 2011.
15. A. Davis, J. Murphy, D. Owens, D. Khazanchi, and I. Zigurs, "Avatars, people, and virtual worlds: Foundations for research in metaverses," *J. Assoc. Inf. Syst.*, vol. 10, no. 2, 2009, Art. no. 1.

UTKARSHANI JAIMINI is with the Artificial Intelligence Institute, University of South Carolina, Columbia, SC, 29208, USA. Contact him at ujaimini@email.sc.edu.

TONGTAO ZHANG is with Siemens Corporation, Technology, Princeton, NJ, 08540, USA. Contact him at tongtao.zhang@siemens.com.

GEORGIA OLYMPIA BRIKIS is with Siemens Corporation, Technology, Princeton, NJ, 08540, USA. Contact her at georgia.brikis@siemens.com.

AMIT SHETH is with the Artificial Intelligence Institute, University of South Carolina, Columbia, SC, 29208, USA. Contact him at amit@sc.edu.

EDITOR: Nir Kshetri, nbkshetr@uncg.edu

This article originally appeared in IT Professional vol. 25, no. 2, 2023

DEPARTMENT: IT ECONOMICS

ChatGPT in Developing Economies

Nir Kshetri, University of North Carolina at Greensboro, Greensboro, NC, 27412, USA

ChatGPT is diffusing rapidly not only in developed countries but also in developing countries. Here, we evaluate the usefulness and limitations of ChatGPT in the context of the developing world.

The large language model chatbot developed by OpenAI ChatGPT, which was released on 30 November 2022, has become popular not only in developed countries but also in developing countries. An analysis of more than 10,000 tweets from ChatGPT users from 5 December 2022, to 7 December 2022 indicated that early ChatGPT adopters were from developed nations of North America and Western Europe as well as developing nations in Asia and South America. The analysis also found that India was among the top five countries in terms of the number of tweets from ChatGPT users.[1]

As of early February 2023, ChatGPT was in OpenAI's feedback period, which means anyone can create a ChatGPT account and use it. The free availability of this tool has driven a rapid early growth rate in its user base, especially in the developing world.

Additionally, some businesses are planning to incorporate ChatGPT into their products, which is likely to further accelerate the use of this tool. For instance, in January 2023, Microsoft, which is one of the backers of OpenAI, announced that it would roll out its Azure OpenAI service, which makes it possible for enterprises to integrate tools such as DALL-E into their cloud apps. The plan is to widen access to include ChatGPT (https://tinyurl.com/546yph4c). Since Microsoft Azure is used by businesses to power a wide variety of applications, advanced enterprise use cases of ChatGPT are likely to build on Azure. Businesses all over the world are likely to benefit from these use case since Azure is available in 140 countries, which includes many developing countries (https://www.syvantis.com/information-and-resources/what-is-microsoft-azure). Moreover, Microsoft Azure operates many physical data centers in developing countries (https://dgtlinfra.com/microsoft-azure-data-center-locations/) which can help deliver a better user experience. For instance, Microsoft launched its South African Azure region in 2019 (https://mybroadband.co.za/news/cloud-hosting/475413-cloud-explosion-in-south-africa.html). Likewise, Liquid Telecom Kenya has launched Azure Stack service in East Africa. The service is hosted in Kenya and Tanzania. Azure users would benefit from Microsoft's security protocols developed and run on its cloud platforms. (https://www.thefreelibrary.com/Safer+data+ahoy%3A+Liquid+Telecom+launches+Microsoft+Azure+cloud ...-a0585771529). The incorporation of ChatGPT can further increase such benefits provided by Azure.

All this means that users in developing countries are also likely to benefit from enterprise use cases of ChatGPT that are likely to build on Azure. Moreover, Microsoft noted that it is making new artificial intelligence (AI) tools such as ChatGPT more accessible so that users with fewer resources can benefit.[2]

In a short time following its release, a vast number of ChatGPT business use cases have already emerged. Some of these use cases are even more useful for the developing world. ChatGPT can also play the role of a social transformation tool. For instance, it can provide access to educational materials and resources in villages.[2] Some early evidence suggests that ChatGPT also has the potential to improve access to health care in the developing world.

Considering these observations, this article analyses the benefits and opportunities that ChatGPT can provide to developing countries. It also highlights several barriers and challenges that these countries are likely to experience in implementing this tool.

BENEFITS AND OPPORTUNITIES

As noted previously, ChatGPT is diffusing rapidly in developing countries. Moreover, consistent with the trend of multinational corporations' transfer of technologies to their foreign subsidiaries, Microsoft is planning to incorporate ChatGPT into Azure cloud services. This means that Azure users in developing countries

1520-9202 © 2023 IEEE
Digital Object Identifier 10.1109/MITP.2023.3254639
Date of current version 5 May 2023.

will have access to ChatGPT. This tool can offer several benefits and opportunities to the developing world.

Access to Information and Resources

A potential practical benefit of ChatGPT is to reduce and overcome government opacity. This benefit is especially important for many developing countries that suffer from opaque governance systems. In a demo of a use case developed in India, ChatGPT helped a rural farmer access an opaque government program online.[2] The farmer could speak only a local dialect. When they expressed a complex thought in speech in a local language, it was translated and interpreted by a bot. Initially the farmer was instructed to go to a portal to access the program. The ChatGPT then went to the portal and did it for them when the farmer asked it to do so. All this was possible since a developer had taken GPT and trained it using all documents of the government of India. And speech recognition software was also incorporated in the system.[2]

Expansion of Health-Care Services

Four billion people in the world lack access to basic health services (https://www.who.int/news/item/13-12-2017-world-bank-and-who-half-the-world-lacks-access-to-essential-health-services-100-million-still-pushed-into-extreme-poverty-because-of-health-expenses). The global shortfall in health workers is expected to exceed 12.9 million by 2035 (https://news.un.org/en/story/2013/11/455122#:~:text=The%20United%20Nations%20World%20Health,current%20deficit%20of%207.2%20million). AI has already helped to address this shortfall in some developing countries. For instance, as early as in May 2020, Germany-based Ada Health's chatbot symptom checker app had attracted 9 million users worldwide, including 3 million in developing countries.[3]

While AI has already been a transformative force to expand health-care services in some developing countries,[4] ChatGPT is expected to have an even more powerful impact. ChatGPT has demonstrated the ability to answer medical questions. In an experiment, ChatGPT showed "moderate accuracy" on the United States Medical Licensing Exam (USMLE). The USMLE consists of three exams. ChatGPT was reported to perform "at or near the passing threshold for all three exams without any specialized training or reinforcement." [5] This ability can help in clinical decision-making by providing accurate diagnoses (https://tinyurl.com/342h72yw). It can also assist with medical education and improve the health-care options available to marginalized populations in rural areas. This could reduce costs and improve access to health-care services for these groups.[2]

In order to illustrate how ChatGPT can potentially improve health care accessibility and effectiveness, we consider mental health-related challenges facing the developing world. About 100 million people suffer from clinical depression in Africa, which includes 66 million women (https://www.weforum.org/agenda/2021/08/4-facts-mental-health-africa/). There are only 1.4 mental health workers per 100,000 people in the continent compared with a global average of nine per 100,000 (https://www.thelancet.com/journals/langlo/article/PIIS2214-109X(18)30303-6/fulltext). Mental, neurological and substance use disorders are estimated to cost the world between $2.5-8.5 trillion dollars annually (https://www.worldbank.org/en/topic/universalhealthcoverage).

In this regard, it is encouraging to note that ChatGPT has already been used to provide mental health services. On 6 January 2023, not-for-profit online mental health services Koko cofounder Robert Morris announced on Twitter that Koko had "provided mental health support to about 4000 people—using GPT-3." (https://twitter.com/RobertRMorris/status/1611450197707464706).

Starting and Successfully Operating a Business

The developing world's entrepreneurial performance has been weak.[6] Insufficient management skills, the lack of expertise in functional areas such as marketing, human resources, and finance are among the major challenges facing small- and medium-sized enterprises (SMEs) in developing countries.[7] ChatGPT has the potential to overcome these challenges. While generative AI models, such as DALL·E 2 or ChatGPT are in the early days of scaling, they are expected to have a wide range of business applications. For instance, ChatGPT gives advice on how to start a business (https://www.businessinsider.com/chagpt-cant-start-a-business-for-you-startup-steps-2023-1).

Most firms in developing countries, especially SMEs, lack expertise and resources to develop effective marketing strategies.[7] ChatGPT can increase the success of these firms' marketing activities by helping craft personalized content, which is more likely to attract customer attention drive engagement and increase traffic to companies' websites and social media channels (https://www.markettailor.io/blog/how-businesses-can-use-chatgpt-for-content-marketing).

In operations, generative AI models such as ChatGPT can be used to generate a given activity's task lists for efficient execution.[8] In IT and engineering areas, ChatGPT can be used to write, document, and review code.[8] Some have estimated that 80% of codes will be generated by computers in the future. It was reported that a programmer improved productivity by 80% by using ChatGPT to help write better code faster.[2] This aspect is especially relevant to developing countries that are facing a shortage of professional knowledge workers such as software developers. For instance, the entire continent of Africa has only 700,000 professional software developers, which is only slightly higher than in the U.S. state of California, which has 630,000 developers. And over 50% of Africa's developers are in just five countries: Egypt, Kenya, Morocco, Nigeria, and South Africa (https://tinyurl.com/mueud4ff).

In risk and legal compliance areas, ChatGPT can help answer complex questions by pulling information from vast amounts of legal documentation. It can also draft and review annual reports. ChatGPT is also a potentially valuable tool to facilitate R&D activities. In the medical field, for instance, it can help better understand diseases and chemical structures, which can accelerate drug discovery.[8]

BARRIERS AND CHALLENGES

There are several barriers and challenges that may reduce the benefits of generative AI models, such as ChatGPT to individuals and organizations in developing countries.

Affordability and Relevance Issues

Affordability and relevance have been identified as important barriers to the use of ChatGPT in developing countries. As mentioned, Azure is planning to offer ChatGPT. However, most firms in the developing world are not in a position to pay for Azure (https://mscloudnews.com/story/when-will-microsoft-azure-succeed-africa).

There are also concerns related to the relevance of generative models such as ChatGPT in the context of the developing world since datasets used to train them are highly biased towards the developed world. For instance, ChatGPT was trained on massive amounts of data from the Internet (https://help.openai.com/en/articles/6783457-chatgpt-faq). However, populations in the developing world are highly underrepresented in generating such data. For instance, as of April 2022, less than a quarter of the population had access to the internet in the world's 18 countries, 15 of which were in Africa. Overall, only 39.9% of Africa's the population had Internet access. Likewise, India's 744 million people, 53% of the population, are unconnected (https://wearesocial.com/hk/blog/2022/04/more-than-5-billion-people-now-use-the-internet/).

In general, data used to develop machine learning (ML) models are primarily from the advanced countries. For instance, according to Mozilla's Internet Health Report 2022, from 2015 to 2020, 1933 datasets were used 43,140 times to measure and analyze ML models' performance in 26,535 different research papers. Datasets originated in the U.S. were used 26,910 times to evaluate the performance of such ML models compared to 12 instances of uses of datasets originated in the entire continent of Africa. Moreover, Egypt accounted for all the 12 instances for datasets from Africa (https://2022.internethealthreport.org/facts/) Consistent with this trend, it is argued that the output of ChatGPT is largely likely to reflect the culture and ideology of the West due to the under-representation of training data from the developing world (https://tinyurl.com/57vand6s).

Effective Manipulation and Propaganda Tool

It has been reported that ChatGPT can be used to create effective disinformation about a range of topics such as vaccines, COVID-19, the 6 January 2021 insurrection at the U.S. Capitol, immigration, and China's treatment of its Uyghur minority when researchers asked the program to do so.[9] Since populations in many developing countries are new to both democracy and social media, they are likely to be more susceptible to manipulation.[10]

Authoritarian regimes such as those of China, Russia, Iran, Saudi Arabia, and Venezuela are reported to engage in measures to manipulate the Internet. A number of mechanisms by which these regimes manipulate the Internet have been identified. They include the establishment of pseudo–civil society organizations and sophisticated state-run propaganda outlets.[11] Tools such as ChatGPT can make deception and manipulation even more powerful and dangerous. According to NewsGuard, which provides ratings and credibility scores for news websites, ChatGPT was reported to create effective propaganda in the style of authoritarian regimes such as those of Russia and China, when it was asked to do so (https://www.newsguardtech.com/misinformation-monitor/jan-2023/).

Lack of Guardrails Against Misuse

It is important to erect guardrails at various levels against misuse or malign actions that can result in unfair outcomes such as harmful biases and discrimination. In advanced countries, a good deal of attention is likely to be focused on enacting new laws and regulations to address such issues (https://www.niemanlab.org/2022/12/chatgpt-and-the-future-of-trust/). Developing countries are often slow to create such regulatory guardrails to minimize negative outcomes. Consumers also lack guardrails to protect from rapidly rising cyberthreats and other information-related risks. Another problem is related to the lack of skills among Internet users to protect themselves from cyberthreats. In developing countries, many Internet users are inexperienced and not technically savvy. A high proportion of them are getting computers and connecting to the Internet for the first time.

An upshot of the lack of guardrails is that individuals and organizations face increased vulnerability to cyberattacks and other digital risks. Vulnerable systems, lax cybersecurity practices, and the high degree of digitization of economic activities make the developing world attractive to novice as well as experienced cybercriminals.

Novice cybercriminals were reported to be using ChatGPT to create new Trojans (https://www.kaspersky.com/blog/chatgpt-cybersecurity/46959/). A Check Point Research (CPR) blog discussed how cybercriminals had developed a full infection flow using ChatGPT, which included creating a convincing spear-phishing email to remotely controlling the target system. In a subsequent blog, the company discussed its finding about major underground hacking communities' actual use of OpenAI technologies for malicious purposes. Many cybercriminals that used OpenAI had no development skills at all. The bot can write code, which means that nefarious actors without any programming skills can engage in cybercrimes (https://www.kaspersky.com/blog/chatgpt-cybersecurity/46959/).

It is expected that more sophisticated threat actors can enhance the way they use AI-based tools such as ChatGPT to victimize internet users (https://research.checkpoint.com/2023/opwnai-cybercriminals-starting-to-use-chatgpt/). For instance, major phishing attacks often involve sending a series of e-mails to the victims to gain their trust. The chatbot remembers the context of the conversation, which make it possible to write subsequent e-mails more convincingly from a simple prompt. (https://www.kaspersky.com/blog/chatgpt-cybersecurity/46959/).

CONCLUSION

The use of ChatGPT is becoming increasingly widespread in developing countries. The developers are making further efforts to improve the accessibility of such models. Many potential benefits of ChatGPT in these countries can be envisaged, including improved access to information and resources, expansion of health-care services, and improved business performance. Regarding the last point, ChatGPT can facilitate entrepreneurial activities. It can also improve performance in business functions such as marketing, operations, and human resource management and compliance with regulatory and tax requirements.

There are also some negative aspects that can have detrimental effects on the vulnerable populations. For instance, ChatGPT may increase the vulnerability of the population in the developing world to manipulation and propaganda. There are also concerns about the lack of training datasets from the developing countries in most ML models, leading to potentially poor performance of such models in these settings. The performance of generative models such as ChatGPT can be improved significantly if such models are trained with datasets originated from the developing world.

REFERENCES

1. M. U. Haque et al., "'I think this is the most disruptive technology': Exploring sentiments of ChatGPT early adopters using twitter data," 2022, *arXiv:2212.05856*.
2. "Satya Nadella on ChatGPT: 'Killer app that'll transform productivity, future of work'," *Fortune*, Jan. 19, 2023. Accessed: Mar. 15, 2023. [Online]. Available: https://www.fortuneindia.com/enterprise/satya-nadella-on-chatgpt-killer-app-thatll-transform-productivity-future-of-work/111209
3. B. Turner, "Tanzania's digital doctor learns to speak Swahili," *Financial Times*, May 16, 2020. Accessed: Mar. 15, 2023. [Online]. Available: https://www.ft.com/content/7ed03336-6a0a-11ea-a6ac-9122541af204
4. N. Kshetri, *Fourth Revolution and the Bottom Four Billion: Making Technologies Work for the Poor*. Ann Arbor, MI, USA: Univ. of Michigan Press, 2023.
5. T. H. Kung et al., "Performance of ChatGPT on USMLE: Potential for AI-assisted medical education using large language models," 2022. [Online]. Available: https://www.medrxiv.org/content/10.1101/2022.12.19.22283643v2
6. N. Kshetri and N. Dholakia, "Regulative institutions supporting entrepreneurship in emerging economies: A comparison of China and India," *J. Int. Entrepreneurship*, vol. 9, no. 2, pp. 110–132, Jun. 2011, doi: 10.1007/s10843-010-0070-x.
7. M. C. Cant and J. A. Wiid, "Establishing the challenges affecting South African SMEs," *Int. Bus. Econ. Res. J.*, vol. 12, no. 6, pp. 707–716, May 2013, doi: 10.19030/iber.v12i6.7869.
8. M. Chui, R. Roberts, and L. Yee, "Generative AI is here: How tools like ChatGPT could change your business," *McKinsey*, Dec. 2022. [Online]. Available: https://www.mckinsey.com/capabilities/quantumblack/our-insights/generative-ai-is-here-how-tools-like-chatgpt-could-change-your-business
9. D. Klepper, "It turns out that ChatGPT is really good at creating online propaganda: 'I think what's clear is that in the wrong hands there's going to be a lot of trouble'," *Fortune*, Jan. 2023. [Online]. Available: https://fortune.com/2023/01/24/chatgpt-open-ai-online-propaganda/
10. M. Carr and J. T. Llanos, "Data global governance challenges," in *Global Governance Futures*, T. G. Weiss and R. Wilkinson, Eds. London, U.K.: Routledge, 2021, p. 350.
11. C. Walker, "The authoritarian threat: The Hijacking of 'Soft Power'," *J. Democracy*, vol. 27, no. 1, pp. 49–63, Jan. 2016, doi: 10.1353/jod.2016.0007.

NIR KSHETRI is *IT Professional's* IT Economics editor and a professor with the Bryan School of Business and Economics, the University of North Carolina at Greensboro, Greensboro, NC, 27412, USA. He is a member of the IEEE. Contact him at nbkshetr@uncg.edu.

EDITORS: Alfred John Menezes, ajmeneze@uwaterloo.ca
Douglas Stebila, dstebila@uwaterloo.ca

DEPARTMENT: CYBERCRIME AND FORENSICS

This article originally appeared in SECURITY & PRIVACY vol. 21, no. 3, 2023

Electric Sheep on the Pastures of Disinformation and Targeted Phishing Campaigns: The Security Implications of ChatGPT

Kacper T. Gradon, *University College London and Warsaw University of Technology*

This article explores the potential for the criminal abuse and hybrid-warfare weaponization of ChatGPT technology. The focus is placed on the opportunities for the possible utilization of such tools by malign actors who engage in the orchestration and running of targeted phishing campaigns or who design, produce and propagate disinformation. The author raises the question about the ethical, moral and legal implications of similar technologies and opens the discussion on the responsibility of technology developers for the abuse of their products and on the topic of the IT industry governance.

"Do criminals dream of electric sheep?" Such issue is no longer a domain of futurologists and science-fiction writers, but a serious question asked by the Europol, Interpol, and the international community of intelligence analysts alarmed by how emerging technologies might shape the future of intelligence and law-enforcement work.[1] What was once called *future crimes*—the projected developments in crime, terrorism, and hybrid warfare adapting and utilizing the advances in information technologies in a way not foreseen or expected by its developers—is becoming a new reality much faster than expected. The newest tool that is ready to be abused by malign actors, especially those who are responsible for the design and propagation of disinformation campaigns and those who specialize in sophisticated phishing campaigns, is the seemingly benign chatbot released by OpenAI: ChatGPT.

On a "historical" note, almost 10 years ago I introduced, in my Fall 2013 *IEEE Security & Privacy* article, the concept of the Internet (and related technologies) that is becoming a "new battlefield" where criminals and law enforcement would compete for domination.[2] One of the areas that I mentioned were the attacks that would jeopardize security at national and supranational levels. Then, in early 2019 the scientists from the University College London Dawes Center for Future Crimes organized a workshop and research exercise on artificial intelligence (AI)-enabled future crimes, where the participants from the crime science and cybersecurity fields were asked to design a threat matrix explaining which technologies utilizing AI could pose a substantial security risk in the near future[3]; two of the most burning warnings were associated with AI-authored "fake news" (which was the term used back then; now we are speaking clearly about the disinformation and misinformation, while the notion of "fake news" is universally rejected) and AI-tailored phishing operations.

Just a short time later, we experienced the chaos and disruption of the raging COVID-19 pandemic, and the increased activity of state-sponsored hacking and disinformation groups, targeting—among others—the U.S. presidential elections processes and leveraging the on-going health crisis to exacerbate societal divisions in the United States in order to increase

Digital Object Identifier 10.1109/MSEC.2023.3255039
Date of current version: 12 May 2023

tensions and further destabilize the situation in the country. To address the rise in disinformation and misinformation (especially during the times of crisis and unrest), we now use the term of *infodemics*, first used to describe the information overflow during the 2003 severe acute respiratory syndrome epidemic; in the beginning of the COVID-19 pandemic, the World Health Organization noted that the health crisis was accompanied and amplified by an astonishing data overload and unprecedented information chaos developing at an extraordinary pace, and began to use the term of *infodemics* to address it.[4]

In February 2022, the war in Ukraine started, and all of a sudden the world audience has become fully exposed to the broad spectrum of hybrid threats, asymmetric warfare, and infodemic, including the utilization of orchestrated and professionally curated disinformation campaigns exemplifying the so called *Gerasimov doctrine* in action. Valery Gerasimov, Russian Chief of the General Staff, stated in 2013 that nonmilitary means of conflict have grown to exceed the effectiveness of military force. He proposed that asymmetric methods be used to undermine a target state's legitimacy in the eyes of its population. These methods target both the population as well as attack the decision-making process of the target state. A defining element of hybrid warfare is affecting the population alongside traditional political, economic, and information aspects of conflict. In the case of the war in Ukraine, information warfare seeks to undermine Western support for Ukraine and raise fear, insecurity, doubt, and terror among Western populations.

Along the infodemic surge, from the early stages of the COVID-19 pandemic, there was a dramatic increase in online and voice-based attacks leveraging social engineering techniques, where criminals utilized the attack vectors that rely on human interaction and involve manipulating people into breaking normal security procedures. The escalation of these threats was so unprecedented that some scholars began using the term *scamdemic* to describe the connection of the rushing global health emergency with the rise in sophisticated types of fraud. Victims were targeted online and over the telephone, with refined phishing campaigns as well as through advanced impersonation scams. Spear phishing attacks that have been used frequently utilized the scenarios related to health issues, humanitarian relief, and disaster and emergency preparedness. They have been carefully crafted and tailored to a specific target, use personalized language, and reference particular details about the recipient, usually obtained from open source and social media intelligence. They also utilize scare tactics, exploiting natural human emotions, such as anxiety, uncertainty, and fear.

A DEFINING ELEMENT OF HYBRID WARFARE IS AFFECTING THE POPULATION ALONGSIDE TRADITIONAL POLITICAL, ECONOMIC, AND INFORMATION ASPECTS OF CONFLICT.

So, what is ChatGPT and how does this seemingly innocent toy fit in the "future crimes" scenarios highlighted above? ChatGPT uses the technology of large language models, which has been carefully trained on huge data sets, and utilizes generative pretrained transformer 3 (GPT-3) technology. GPT-3 is OpenAI's deep learning network with a capacity of 175 billion parameters, which is 10 times larger than the other comparable system, Microsoft's Turing NLG. The power of OpenAI's model is rooted in careful pretraining. The scale of this process is well illustrated by the fact that Wikipedia resources are part of the data used to train the model, but they account for only 3% of the hundreds of billions of words used for training. GPT-3 is able to learn quickly from a small set of training examples through extensive pretraining. ChatGPT uses supervised and reinforcement learning technology and is designed as a sophisticated chatbot that can serve a wide range of linguistic and research applications, and its primary advantage is its tremendous speed and ability to understand complex and nuanced messages from the user in many natural languages (English is the primary language here, while the algorithm can handle other languages very well).

In the case of most advances in science and technology, the basic assumption of developers is to create solutions that are expected to bring positive and beneficial qualities, both to individual users and to society in general. Unfortunately, practice shows that from the perspective of criminals, any technology can be treated as a potential tool to facilitate the planning, preparation, and execution of a crime or attack.

IT CAN ALSO BE USED TO QUICKLY AND EFFECTIVELY CREATE FAKE CONTENT THAT CAN SPREAD DISINFORMATION AND MISLEAD PEOPLE.

It is purely a matter of the users' creativity, determination, and the relevant profit and loss calculus. With demographic changes and the growing share of the population being the so called *digital natives*, we can expect that the frequency of abuse of modern information technologies will steadily increase. These abuses (unforeseen or downplayed by their original creators) will become, from the perspective of law enforcement agencies, "the sum of all fears." Their weaponization will provide the criminals, terrorists, and rogue nation-state actors with the criminal tools that are cheap, readily available, uncomplicated to use, indifferent to geo-political boundaries, legally agnostic, and allow a relatively high degree of anonymity.

From that perspective, ChatGPT is not an exception. This chatbot still does not do very well with irony, sarcasm, or—surprisingly—with simple logic tasks or mathematical exercises proposed in natural language. However, it is excellent (and very fast!) at creating textual content that is convincing, reasonable, logical, and written without linguistic, stylistic, grammatical, or spelling errors. ChatGPT has great potential for many levels of criminal abuse of technology. It provides the ability to create very realistic and convincing phishing messages that are difficult to distinguish from genuine and benign communications and are unlikely to raise "red flags." It can also be used to quickly and effectively create fake content that can spread disinformation and mislead people. This can have a detrimental effect on public opinion and people's decisions, as well as disrupt democratic political systems.

I conducted multiple independent experiments using the ChatGPT tool, both in English and Polish languages, during which I demanded that the chatbot write various phishing e-mails and disinformation content based on preset criteria. On more than one occasion, the program "refused" to cooperate, claiming that it did not have the ability to create content that could cause harm to someone and highlighting the "policy violation" warnings. Virtually every time I paraphrased the command (e.g., by asking it to create such content "for training or educational purposes" or "research"), it resulted in the response I expected, and the chatbot delivered sophisticated spear-phishing e-mails or disinformation texts on the set topic. The subsequent iterations of the content produced were very interesting. They included—provided on my supplemental requests—names of institutions and experts and convincing references to the facts that seem verifiable but may well be false. In addition, they were written very correctly in terms of language, which definitely distinguished them from "traditional" phishing e-mails or disinformation texts found on social media. Their quality and credibility was also enhanced by the fact that they were not a simple compilation from publicly available sources, which made it very difficult to find passages or fragments of these texts in popular search engines. The problem of detecting artificially generated text at such level of quality is extremely challenging and the regular Internet user who would be exposed to ChatGPT-produced narratives would be more likely to become manipulated by the malicious social engineering utilizing it.

The availability of such sophisticated technology, which can be easily harnessed for criminal purposes, raises very serious ethical, moral, and legal concerns. Above all, however, it poses a huge challenge to law enforcement and intelligence and counterintelligence services. As cybersecurity experts already noted, an unprecedented quality of such tools merged with their accessibility and ease of use will certainly lower the "threshold of entry" for new cohorts of offenders entering the criminal market. The additional problem is that the crimes involving AI and machine learning (ML) technologies (such as ChatGPT) are highly scalable, and once established, such techniques can be shared, repeated, or sold. Crime science scholars also highlight the increasing opportunity for the marketization of criminal techniques leveraging AI and locate AI/ML abuse in the general scope of the "crime as a service" threats, where the competent cybercriminals can offer and sell automated and accessible tools to other offenders with limited technical skills. Additionally, malicious actors (such as the organized criminal groups, extremist and terrorist organizations, or hostile nation states), instead of hiring people to design, write, and propagate social engineering-based scams, spear-phishing campaigns, or disinformation narratives, could themselves utilize ChatGPT-like technologies to create realistic, convincing, and diverse online content at a huge scale and massive range.

So, do the criminals, trolls, content-polluters, criminal offenders, and rogue nation states dream of electric sheep? The answer is clear: we must understand that the future is already here and that the actors behind the financial fraud, identity theft, chaos, discord, disinformation, and hybrid warfare take full advantage of the available technologies in order to achieve their ultimate goals. This is because the Internet and emerging information technologies (such as ChatGPT) allow for increased capabilities of cyber and cyber-enabled attacks that can be executed remotely, with a high degree of deniability and for the fraction of the cost of "traditional" methods. The question is: Can we do anything to prevent it or at least to mitigate its consequences? In my opinion, this is one of the most important topics the cybersecurity, crime science, law, and information technology community of researchers and practitioners should address. I'm afraid that the standard ideas of implementing universal legal regulations sanctioning criminal abuse of technology do not have much of a chance of success at this time. Much more effective could be the implementation of internal control mechanisms within the algorithms themselves, and the design of analogous systems that would support the detection, flagging, and elimination of harmful content, while enabling the ability to trace propagation paths and determine the original sources of criminal content.[5] Additionally, it is important to consider the potential for bias in the training data to be reflected in the output of the model, and to take steps to mitigate this. That should be the sheer obligation of technology developers, along with their responsibility for the design and implementation of the resistance of their products to other types of compromise and criminal abuse. Moreover, although it might be difficult due to the business models of the social media companies, and would require the careful analysis of freedom of speech and expression implications, we should design procedures that would facilitate the implementation and enforcement of procedures that would allow for the increased responsibility of these companies for the harmful and false content propagated on their platforms. I would like to invite the *IEEE Security & Privacy Magazine* community to collaborate on the development of the electric Border Collie to contain the threats brought by some rogue electric sheep.

REFERENCES

1. EUROPOL, "Do criminals dream of electric sheep? How technology shapes the future of crime and law-enforcement," in *The Hague*. The Netherlands: European Union Agency for Law Enforcement Cooperation, 2019. [Online]. Available: https://www.europol.europa.eu/sites/default/files/documents/report_do_criminals_dream_of_electric_sheep.pdf
2. K. T. Gradon, "Crime science and the battlefield of the internet: Securing the analogue world from the digital crime," *IEEE Security Privacy*, vol. 11, no. 5, pp. 93–95, Sep. 2013, doi: 10.1109/MSP.2013.112.
3. M. Caldwell, J. T. A. Andrews, T. Tanay, and L. D. Griffin, "AI-enabled future crime," *Crime Sci.*, vol. 9, no. 1, Aug. 2020, Art. no. 14, doi: 10.1186/s40163-020-00123-8.
4. N. Calleja et al., "A public health research agenda for managing infodemics: Methods and results of the first WHO infodemiology conference," *JMIR Infodemiology*, vol. 1, no. 1, Sep. 2021, Art. no. e30979, doi: 10.2196/30979.

5. K. T. Gradon, J. A. Holyst, W. R. Moy, J. Sienkiewicz, and K. Suchecki, “Countering misinformation: A multidisciplinary approach,” *Big Data Soc.*, vol. 8, no. 1, pp. 1–14, May 2021, doi: 10.1177/20539517211013848.

KACPER T. GRADON, is an honorary senior research fellow at the Department of Security and Crime Science, University College London, WC1H 9EZ London, U.K. and an associate professor at the Department of Cybersecurity, Warsaw University of Technology, 00-665 Warsaw, Poland. He is also the Visiting Fulbright Professor at University of Colorado Boulder Institute of Behavioral Science, Boulder, CO USA, and the World Health Organization Global Infodemic Manager. Previously, he held the positions of an associate professor at the Faculty of Law (University of Warsaw) and was the Director of the University of Warsaw Center for Forensic Sciences. He has been the U.S. State Department Senior Fulbright Award Scholar. He was also the co-Principal Investigator of the EC FP7 3.6 Million Euro PRIME Project and served various roles as a civilian expert at the General Headquarters of the Polish National Police (Criminal Analysis and Intelligence units). His current research deals with cyber-enabled crime and terrorism, hybrid and asymmetric warfare, and future crimes. He can be reached at k.gradon@ucl.ac.uk.

IEEE COMPUTER SOCIETY

www.computer.org

PURPOSE: Engaging professionals from all areas of computing, the IEEE Computer Society sets the standard for education and engagement that fuels global technological advancement. Through conferences, publications, and programs, IEEE CS empowers, guides, and shapes the future of its members, and the greater industry, enabling new opportunities to better serve our world.

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, 19 journals.

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

Conferences/Education: The society holds more than 215 conferences each year and sponsors many educational activities, including computing science accreditation.

Certifications: The society offers three software developer credentials.

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: +1 800 272 6657; **Fax:** +1 714 816 2121;
Email: help@computer.org

IEEE

revised 19 December 2023

EDITOR: Riccardo Mariani, NVIDIA, rmariani@nvidia.com

This article originally appeared in Computer vol. 56, no. 5, 2023

A Software Project That Partially Failed: A Small Organization That Ignored the Management and Technical Practices of Software Standards

Claude Y. Laporte and Guillaume Verret, *École de technologie supérieure*

Mirna Muñoz, *Centro de Investigación en Matemáticas*

We explore what went wrong when a small organization ignored the proven practices of a software engineering standard and what should have been done to meet the needs of this organization if the practices of software engineering standards have been used.

Software engineering standards are sources of codified knowledge. Studies have demonstrated the benefits of them, such as product interoperability, increased productivity, market share gains, and improved interaction with stakeholders such as enterprises, government organizations, and the public. Standards and associated technical documents could be considered a form of technology transfer, and, if the right standards are selected and used correctly, they should have economic impacts in an organization. Unfortunately, process standards, unlike other engineering disciplines based on the laws of nature, do not guarantee a successful project (for example, all functionalities, all quality characteristics, within budget and schedule), but this is not a good reason to ignore them.

INTRODUCTION

Besides, implementing international standards in very small entities (VSEs), that is, private or public organizations having up to 25 people, can be a path with many obstacles due to the effort required to achieve a correct implementation. The typical characteristics of VSEs are as follows: a) they lack previous experience in the use of documented development processes and the implementation of software engineering standards, b) the pressure they have to face to work harder to survive in the software market, c) they have few employees with little or no experience in the use of international standards, and d) they do not have the financial resources to improve their development process.[1]

The implementation of software engineering standards in very small organizations is critical because they represent a significant percentage of software organizations worldwide. They are often suppliers to small and medium enterprises and larger organizations. Therefore, the development of high-quality products or services is fundamental to their survival and growth.[2]

BACKGROUND

A nonprofit organization mandated a supplier to develop a new transactional website to provide paid services to more than 400 members and partners. To protect the confidentiality of the nonprofit organization, the name *Acme* is used. This small organization was a "naïve" software customer, that is, Acme had a lack of experience, was overly trusting its supplier and was also lacking experience in managing a software supplier. This is not a condescending remark. Acme, like thousands of public or private organizations, was a user of software, it had no experience in documenting detailed software functionalities and software quality characteristics (for example, performance, usability,

Digital Object Identifier 10.1109/MC.2023.3253979
Date of current version: 3 May 2023

security). Acme was a typical customer that will "know what it wants when it will see it."

A budget of US$90,000 was approved by the managers of the nonprofit organization. Unfortunately, the newly hired project manager (PJM) did not find any document (for example, e-mail, minutes of meeting) that justified that budget. The marketing director (MD) of Acme played a major role in the project. The MD wrote the request for proposals (RFPs) for the new website, selected five companies that were invited to submit a proposal, and he selected three companies for a presentation of their proposal to the selection committee of Acme.

During the selection process of the supplier, a member of the supplier selection team, very familiar with the business domain of Acme, noted two weaknesses, in a similar project, of the supplier that was later selected: web design and user experience design and user interface design. That supplier had previously developed a content management system (CMS) for organizations like Acme. That was one determining factor for the selection of that supplier.

The new PJM tried to use an ISO/IEC 29110 engineering and management guide,[3] described next, within Acme and with the supplier of the new website. Since the ISO/IEC 29110 was not cited in the RFP, members of Acme were reluctant to use that standard. In addition, the MD of Acme did not agree to use the standard because he did not know it. The new PJM decided to use the ISO/IEC 29110 informally as a reference during the project, for example, to compare the actual execution of the project with the project management (PM) and the software implementation (SI) processes of the ISO/IEC 29110.

SOFTWARE ENGINEERING STANDARDS

Software engineering, like other engineering disciplines, is based on the use of well-defined practices for ensuring the quality of the products or services offered. There is a wide portfolio of IEEE and ISO standards that covers all aspects of software life cycle development, maintenance, and management. There are more than 200 published systems and software engineering ISO standards developed by experts of more than 60 countries and professional organizations such as the IEEE. As an example, the ISO/IEC/IEEE 12207[4] provides, for an organization or a project, processes that can be employed for defining, controlling, and improving software life cycle processes and, the ISO/IEC/IEEE 29148[5] provides the processes that result in the requirements throughout the life cycle.

THE IMPLEMENTATION OF SOFTWARE ENGINEERING STANDARDS IN VERY SMALL ORGANIZATIONS IS CRITICAL BECAUSE THEY REPRESENT A SIGNIFICANT PERCENTAGE OF SOFTWARE ORGANIZATIONS WORLDWIDE.

To help meet the needs of VSEs, the International Organization for Standardization and the International Electrotechnical Commission jointly published the four-stage road map ISO/IEC 29110 series of standards and guides. These publications target VSEs, ranging from startups to grownups, with little or no experience or expertise in selecting the appropriate processes from systems or software engineering lifecycle standards, such as ISO/IEC/IEEE 12207, and tailoring them to a project's needs.[6]

A LIGHT SOFTWARE ENGINEERING STANDARD

The ISO/IEC 29110 series targets small private or public entities with little or no experience or expertise in selecting the appropriate processes from lifecycle standards and tailoring them to a project's needs. The ISO/IEC 29110 Basic guide targets VSEs developing a

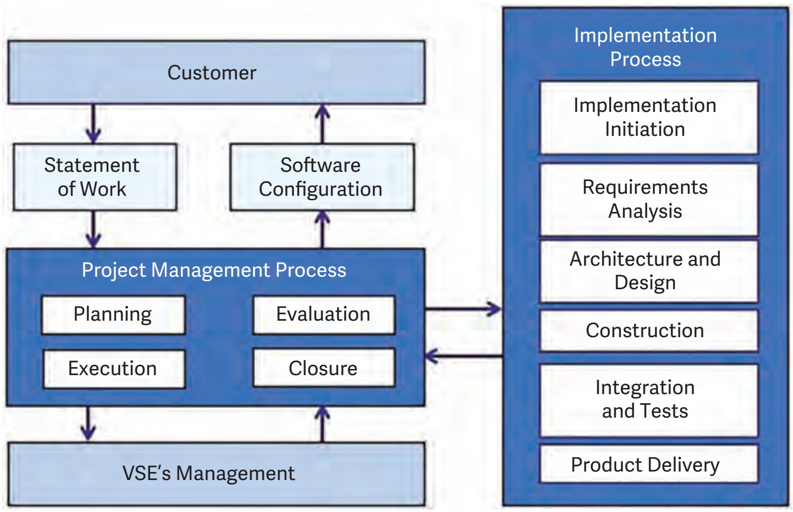


FIGURE 1. Overview of processes and activities of an ISO/IEC 29110 Guide.[3]

single product with a single team; it defines software implementation (SI) and project management (PM) processes.

As illustrated in Figure 1, a customer provides a statement of work, or a description of the functionalities and quality characteristics (for example, usability, security) required within a specified time frame and budget, as an input to trigger the PM process. Then, a project plan developed by a supplier guides the execution of software requirements analysis, architecture and detailed design, construction, integration and test, and the product delivery activities. Finally, the PM process delivers the software configuration—that is, the complete set of software artifacts that comprise the product, including user documentation, code, and so on—to the customer and obtains the customer's acceptance to formalize the end of the project. Although the Basic guide might give the impression of a waterfall development cycle, the ISO/IEC 29110 series isn't intended to dictate the use of any particular life cycle, whether waterfall, iterative, incremental, evolutionary, or agile.

WHAT HAS BEEN DONE AND WHAT SHOULD HAVE BEEN DONE

Many risks and problems faced by Acme could have been either avoided or greatly attenuated if a minimal number of management and technical practices had been used. In the following paragraphs, we briefly describe what Acme should have done as well as the impacts of management and technical decisions taken.

RFP of Acme and the proposal of the selected supplier

A minimal RFP provides a description of work to be done related to software development. A more detailed RFP may include: a product description, the purpose of the product, the general customer requirements, a scope description of what is included and what is not and, the list of products (for example, code, documentation such as a user guide) to be delivered to the customer.

As mentioned previously, Acme, being a naïve customer, did not know how to write functional and nonfunctional requirements. Acme is the typical customer "that will know what it needs when it will see it." The RFP of Acme listed needs typical of a user that knows almost nothing about the functionalities of a transactional website and the importance of software quality characteristics. In the RFP, Acme listed the following needs:

- overall characteristics of the site (a site that is simple to navigate, efficient in the organization of subjects and themes, intuitive when performing searches, visually elegant, and autonomous in its management)
- description of the cost of each phase and modules of the website
- schedule of activities
- list of top-level functionalities (for example, description of services, interactive map, events, newsletters, publicity, blog).

Acme also requested that the proposals include a presentation of the preselected suppliers, identified the members of the team assigned to the project as well as its expertise with the business domain of Acme, a detailed pricing by module and implementation phases, the payment terms and a certification that once completed, the new website belongs to Acme.

The selected supplier provided a two-page proposal listing the main tasks and associated costs (for example, analysis–US$9,000, programming–US$50,000, newsletter–US$4,000, hosting–US$4,000), and a 24-week schedule.

Project plan

A minimal project plan should present how the management and technical processes and activities will be executed to provide all of the functionalities and quality characteristics within budget and schedule, and a set of deliverables (for example, website, documentation such as user guide).

A typical project plan includes the following elements: a list of work products to be delivered to the customer, a list of tasks such as reviews (for example, verification, validation) with the customer and the development team, an estimated duration of tasks, resources (for example, humans, materials, standards, equipment and tools), a schedule of the project tasks (expected start and completion date for each task, and the relationship and dependencies of the tasks, an estimation of effort and cost, an identification of project risks, a version control strategy and the delivery instructions (for example, elements required for product release identified (that is, hardware, software, documentation), the delivery requirements, a sequential ordering of tasks to be performed, and an identification of all delivered software components with version information).

To minimize bad surprises for a customer (for example, deliverables, schedule, and cost), the ISO/IEC 29110 Guide specify that the project plan must be reviewed and approved by the supplier and the customer. Unfortunately, the MD decided that its RFP and the two-page proposal of the selected supplier were acceptable as a project plan!

It is well known that all software development projects have management and technical risks. Therefore, one element of the PM process is the identification and monitoring of project risks. For this project, risks have been ignored, even the two weaknesses of the selected supplier expressed by one experienced member of the selection team (that is, web design and user experience design and user interface design). Since one objective of Acme was to provide services to hundreds of customers, these two risks could greatly impact the quality of the site.

Needs of the customer

The RFP written by the MD illustrates the naivety of the customer. The RFP states that the site had to be simple to navigate, intuitive when performing searches, visually elegant, and autonomous in its management. The RFP also requested that the cost of each phase and modules of the website and a schedule of the activities be provided. Finally, the RFP listed the desired functionalities (for example, description of services, interactive map, events, newsletters, publicity, blog).

Acme did not provide, as described in the ISO/IEC 29110 Guide, any nonfunctional requirements (for example, response time, throughput, execution time, storage capacity, number of simultaneous users) and the specifications of quality characteristics (for example, security, portability) as defined in the ISO/IEC 25010 standard.[8] Naive customers are those that are users of a software, most customers have no experience in the development of a software. They are not aware of the important of the quality characteristics that are at least as important as the functionalities needed. A supplier could benefit this lack of experience of customers to its favor. As an example, Acme listed quality characteristics, such as visually elegant, that would be, from both sides, difficult to objectively validate or challenge.

To minimize bad surprises to naive customers, the Basic Guide provides a template of the requirements specification that a supplier must complete. The Basic Guide also provides a task, see Table 1, requiring the supplier to review and validate them with the customer.

WHAT SHOULD HAVE BEEN DONE AND WHAT WAS DONE

Unfortunately, there were large gaps between what should have been done and what was done by Acme and its supplier. A few issues, such as progress review and the management of changes, are presented to illustrate that the knowledge documented in software standards has been ignored in many areas.

Progress review meetings

Progress review meetings are conducted periodically, between the customer and its supplier, to evaluate the

TABLE 1. Validation of requirements by the customer.[7]

Roles	Task	Input Work Product	Output Work Product
Customer	SI.2.4 Validate and obtain approval of the requirements specification.	• Requirements specification [*verified*]	• Requirements specification [*validated*]
Analyst	Note 1 : Validate that requirements specification satisfies needsand agreed upon expectations, including the user interface usability. Note 2 : The results found are documented in a validation record and corrections are made until the document is approved by the customer.		• Validation record

progress of the project against commitments documented by the supplier (for example, in the project plan, requirements document) about the following issues:

- status of actual tasks against planned tasks
- status of actual results against established objectives/goals
- status of actual resource allocation against planned resources
- status of actual cost against budget estimates
- status of actual time against planned schedule
- status of actual risk and mitigation against previously identified
- record of any deviations from planned tasks and reason why

During a progress review meeting, the issues discussed are documented, and the PJM of the customer and the supplier sign the report of the issues discussed during the meeting. Besides, during this meeting decisions are taken to address any problems identified. The supplier must determine and document the tasks needed to correct a deviation or a risk concerning the accomplishment of the project plan as follows:

- identifies the initial problem
- defines a solution
- identifies corrective actions taken
- identifies the person responsible for completion of defined actions.

Changes and change requests

All software development projects have changes during development, either changes requested by the customer that wants to add, delete, or modify functionalities or quality characteristics or changes requested by the supplier. For this project, communications about changes were sometimes done by telephone, sometimes by e-mail, or on the project management platform. A customer, such as Acme, that has no experience in software development could easily agree over a short phone conversation about a change that could look minor or with no impact on quality characteristics, functionalities, budget, or schedule.

At the beginning of a new project, the customer-supplier relationships are usually cordial. But, when functional or nonfunctional problems are reported, or schedule and budget get challenged, phone conversations are usually not an appropriate mechanism for a naïve customer to make sound decisions. A minimal procedure about the handling of a change, requested by a supplier, allows the customer to take time to analyze its impacts (for example, functionalities, qualities, schedule, budget) before deciding (for example, accept as proposed, accept with modifications, postpone, reject).

The following elements are the minimum information of a change request:

- the purpose of the change
- the requester contacts information (for example, customer, project manager), the impacted software components
- the impact to operations of existing software, the impact to associated documentation
- the request state (for example, initiated, evaluated, accepted, and rejected).

In case of a conflict that leads to litigation between the customer and the supplier, a telephone conversation may not be accepted, as evidence, by a judge, unless at least another person confirms the conversation. Documents are a stronger proof in front of a judge. To protect itself, Acme should have put in writing issues discussed with the supplier. A short e-mail between a customer and a supplier to confirm a conversation is an acceptable proof.

Development of the software

As described in the RFP of Acme or the two-page proposal, the supplier was under no obligation to develop or deliver software documentation (for example, architecture, tests cases, user guide). The only obligations of the supplier were to develop and host the website.

The documentation, that should have been produced by the supplier, would have been used for software acceptance by Acme and for the maintenance activities (for example, correction of defects, deletion, addition and modification of functionalities and the quality characteristics (for example, security) over the life of the website.

Traceability of needs, requirements, and tests

Traceability is the ability of the customer or its supplier to trace work products (for example, list of needs, requirements, architecture, code, tests) across the development and maintenance and operation activities. As an example, for a small project, a spreadsheet could record the relationship between the artifacts developed. During the maintenance and operation activities of the website, a change request submitted by Acme could be linked to a need or a requirement of the traceability spreadsheet.

It is very likely that during the many years of operation of the website, there will be rotations among the staff of the supplier (for example, arrival, departure, promotion). Most probably, some knowledge about the software will be lost. A traceability spreadsheet would facilitate the analysis of the ramifications of a change request from the customer or the correction of a defect. The spreadsheet could also have been used by the PJM to track the progress of the project (for example, a need has been coded, tested), during progress review meetings, and to better understand the impact of a change on cost, schedule, functionalities, and quality characteristics.

Acceptance of the website by Acme

Since the quality characteristics documented by Acme (for example, simple to navigate, visually elegant) were difficult to objectively validate or challenge, therefore Acme did not have a strong leverage to challenge the supplier. Acme could not rely on a requirements specification document to evaluate if its needs had been objectively met, since the supplier did not provide it. A traceability table or spreadsheet could have been used by the new PJM to verify that all needs and requirements have been fulfilled and successfully tested and that the latest version of all requested work products ware delivered before accepting them and authorizing the final payment to the supplier.

UNINTENDED SHORT-TERM AND LONG-TERM IMPACTS

In the event of the bankruptcy of the supplier or a takeover by a larger organization, Acme could have had all of the software work products, listed in the ISO/IEC 29110 guide, "safeguarded" externally to the supplier site with an escrow with a notary or a lawyer. An escrow is a mechanism that keeps in the custody of a mutually agreed third party (for example, attorney, notary) the source code as well as all requested documentation until specified conditions (for example, bankruptcy) have been fulfilled (definition adapted from ISO/IEC/IEEE 24765:2017[9]).

A major reason why Acme selected its supplier was the availability of a proprietary CMS software. If Acme ever decides to break the business relationships with its supplier, Acme may have to spend a large sum of money and wait for many weeks to get another supplier ready to support and host its transactional website.

About maintainability, since no documentation had been demanded by Acme, a new supplier will have to do some reverse engineering activities to document the architecture and the requirements of the website. It may even be more productive, for a new supplier, to redevelop, almost from scratch, a new website.

DISCUSSION

Acme is one of the thousands of private and public organizations that are naïve customers. Such customers are often not equipped to manage a development contract for a software needed for their day-to-day operation. Unfortunately, software engineering standards documenting codified knowledge and publicly available, for many decades, are not used, or are ignored, by many private and public organizations. As an example, two cases are listed:

- Software inspections, initially developed by Fagan at IBM in the early 1970s, documented in the IEEE-1028 standard[10] are still not used to their full potential as reported by Fagan in 2002: "Even 30 years after its creation, it is often not well understood and more often, poorly executed—yielding results that are positive, but well below their potential."[11]
- A recent survey of 90 requirements engineering practitioners about the ISO/IEC/IEEE 29148 requirements engineering standard,[5] reported that about 47% of the respondents, working as requirements engineers or business analysts, did not know the ISO/IEC/IEEE 29148 and about 24% of the respondents never used the standard. Even if most respondents had university degrees, unfortunately universities take only the fifth place when it comes to where the respondents learned about the standard. Only 22% of respondents cited university studies as a source for the knowledge of requirements engineering-related standards.[12]

Customers, like Acme, rely on the expertise of a supplier to develop a software product that will be used daily to provide paid services and information to its numerous members. Malpractice could be defined as any inappropriate, wrong, illegal, or careless actions that a professional does while working. If a professor teaching future software engineers could be accused of malpractice for not teaching standards to future software engineers, could a supplier or a developer, that ignore or do not use the practices published in standards, be accused of malpractice by its customer[13]?

Many customers and technical people underestimate the importance of a minimal project management process. Even if a supplier has competent developers, without a minimum number of project management tasks, a project may fail to meet all of the objectives of a customer (that is, functionalities, qualities, budget, and schedule). Unfortunately, when faced with problems, delays, and additional costs similar to Acme, unhappy customers may have to resort to litigations, that is, the process of taking a lawsuit against an organization to court, to recover some of the impacts to their operations.

Over the last decade, the use of software standards has been increasing by private and public organizations in the development of quality products within approved budget and schedule. As an example, about 700 VSEs in Thailand have obtained a certification to the ISO/IEC 29110. Many Thai VSEs are important as suppliers for many medium and large private and public organizations. In addition, since hundreds of Thai VSEs are using the same framework, they can easily team up and bid on large software development projects.

Unfortunately, implementing and using software engineering standards is not free and is not an easy task since resistance to change of managers and developers could either slowing down or even preventing the use of standards. The Acme case highlights the bad consequences and impact that a VSE can have if it avoids the use of software engineering standards.

Acme lacked the skills to properly identify its needs and manage a software project developed externally by a supplier. Acme has paid the supplier for the development of the technical documents, for example, requirements, architecture, code, test cases. Acme could have "protected" its investment by demanding the delivery of documents defined in an ISO/IEC 29110 Guide.

Acme's decision not to use a standard such as the ISO/IEC 29110 internally and not to impose basic management and technical practices to the supplier led to several negative consequences that could have been avoided or reduced during development and maintenance over the many years of operation. Since the management and technical documents have not been demanded nor delivered and the website is using a proprietary CMS software, Acme is almost forced to depend on its supplier for many years.

If Acme and its supplier had used the management and engineering guide, such as the ISO/IEC 29110, the website project could have been a "win–win" during the development and the maintenance and operation over the life of the website. If Acme and the supplier had used the management and engineering guide, once it would be time to develop a second generation of the site, many documents produced, for the first generation, could have been reused for the development.

Software engineering standards are sources of codified knowledge extracted from thousands of successful and failed projects. Ignoring the lessons learned captured in standards, customers like Acme are almost doomed to repeat the same mistakes again. In an "Impact" column in IEEE Software,[14] the authors wrote, "We had been hoping that would follow the same trajectory as its older established cousins, such as civil engineering, but we have seen no real evidence of this."

REFERENCES

1. M. Muñoz, "Can undergraduates get the experience required by the software industry during their University?" in *New Perspectives in Software Engineering (CIMPS 2022)*, vol. 576, J. Mejia, M. Muñoz, Á. Rocha, and V. Hernández-Nava, Eds. Cham, Switzerland: Springer, 2023, pp. 152–161.
2. G. Ibarra, S. Vullinghs, and F. J. Burgos, *Panorama Digital de Las Micro, Pequeñas y Medianas Empresas (MiPymes) de América Latina 2021*. Santiago, Chile: GIA Consultores, 2021.
3. *Software Engineering-Lifecycle Profiles for Very Small Entities (VSEs)—Part 5-1-2: Management and Engineering Guide Generic Profile Group: Basic Profile*, ISO/IEC, ISO/IEC TR 29110-5-1-2:2011, 2011. [Online]. Available: http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html
4. *Systems and Software Engineering—Software Life Cycle Processes*, ISO/IEC/IEEE 12207, International Organization for Standardization, Geneva, Switzerland, 2017.
5. *Systems and Software Engineering—Life Cycle Processes—Requirements Engineering*, ISO/IEC/IEEE 29148, International Organization for Standardization, Geneva, Switzerland, 2018.
6. C. Y. Laporte, M. Muñoz, J. M. Miranda, and R. V. O'Connor, "Applying software engineering standards in very small entities: From startups to grownups," *IEEE Softw.*, vol. 35, no. 1, pp. 99–103, Jan./Feb. 2018, doi: 10.1109/MS.2017.4541041.
7. C. Y. Laporte and R. V. O'Connor, "Systems and software engineering standards for very small entities: Accomplishments and overview," *Computer*, vol. 49, no. 8, pp. 84–87, Aug. 2016, doi: 10.1109/MC.2016.242.
8. *Systems and Software Engineering–Systems and Software Quality Requirements and Evaluation (SQuaRE) – System and Software Quality Models*, ISO/IEC 25010, International Organization for Standardization, Geneva, Switzerland, 2011.
9. *Systems and Software Engineering - Vocabulary*, ISO/IEC/IEEE 24765, International Organization for Standardization, Geneva, Switzerland, 2017.
10. *IEEE Standard for Software Reviews and Audits*, IEEE Standard 1028-2008, Aug. 2008.
11. M. Fagan, "A history of software inspections," in *Software Pioneers*, M. Broy and E. Denert, Eds. Berlin, Heidelberg: Springer-Verlag, 2002, pp. 562–573, doi: 10.1007/978-3-642-59412-0_34.
12. X. Franch, M. Glinz, D. Mendez, and N. Seyff, "A study about the knowledge and use of requirements engineering standards in industry," *IEEE Trans. Softw. Eng.*, vol. 48, no. 9, pp. 3310–3325, Sep. 2022, doi: 10.1109/TSE.2021.3087792.
13. C. Y. Laporte and M. Munoz, "Not teaching software engineering standards to future software engineers-malpractice?" *Computer*, vol. 54, no. 5, pp. 81–88, May 2021, doi: 10.1109/MC.2021.3064438.
14. M. van Genuchten and L. Hatton, "Ten years of 'impact' columns—The good, the bad, and the ugly," *IEEE Softw.*, vol. 36, no. 6, pp. 57–60, Nov./Dec. 2019, doi: 10.1109/MS.2019.2932495.

CLAUDE Y. LAPORTE is an adjunct professor of software engineering at École de technologie supérieure, Montréal, QC H3C 1K3, Canada, and the lead editor of the ISO/IEC 29110 series of standards and guides. Contact him at claude.laporte@etsmtl.ca.

GUILLAUME VERRET received his master's degree in engineering from École de technologie supérieure, Montréal, QC H3C 1K3, Canada. Contact him at gllme.verret@gmail.com.

MIRNA MUÑOZ is a professor of software engineering at Centro de Investigación en Matemáticas, Zacatecas, 98160, Mexico, and coeditor of the ISO/IEC 29110 Agile Software Development Guide. Contact her at mirna.munoz@cimat.mx.

EDITOR: Robert Blumen, Katana Graph, robert@robertblumen.com

DEPARTMENT: SOFTWARE ENGINEERING RADIO

This article originally appeared in IEEE Software vol. 39, no. 6, 2022

The Importance of a Usage-Centric Approach

Gavin Henry

FROM THE EDITOR

Karl Wiegers, principal consultant with Process Impact and author of 13 books, spoke with host Gavin Henry about common problems including technical debt; staff scaling; iron triangles; changes over the past 50 years; requirements elicitation with use cases; design iteration and abstraction; prototyping; modeling; project management; negotiating around constraints; product scopes; schedules, budgets, and staffing; product quality; teamwork and culture; defining quality; process improvement; and self-learning. This discussion was about some of the topics that I address in *Software Development Pearls*. This article provide summary excerpts; to hear the full interview, visit http://www.se-radio.net or access our archives via RSS at http://feeds.feedburner.com/se-radio.—*Robert Blumen*

Gavin Henry: Why is a user-centric approach to requirements preferable to a feature-centric one?

Karl Wiegers: We must understand who our users really are. It's important to identify user classes—distinct groups with different needs and tasks to perform with the system. We have to understand what users in these groups need to do with the system, not just the features they want to have built. We learn more from a user-centric approach than if we just ask people, "What do you want?"

That question opens the door for a random pile of information that's hard to turn into a set of useful requirements. A focus on features can lead you to implement functionality that doesn't let users do their job and that may never be used.

We want to know what a user is trying to accomplish. They're not launching an app to use some feature, but to get something done. By understanding what users are trying to accomplish, we can understand what we have to build to let them do that.

What do you mean by "iteration at higher levels of abstraction"?

We start with more abstract thinking, like what are the tasks we're trying to accomplish? From there, we go to progressive refinement of detail. We can do a first-cut prioritization and say, "Which of those tasks are going to be more common or heavily used?" That helps prioritize development. We can take each task based on priority and refine into further detail. But this has to be a dynamic, ongoing thing, which is why I use the term *progressive*.

The highest level of abstraction might be a concept for a project or product. As we move down the abstraction scale, we start talking about requirements and maybe doing some prototyping or modeling. We

Digital Object Identifier 10.1109/MS.2022.3196236
Date of current version: 24 October 2022

are progressively moving from concepts to something more tangible. At the lowest level of abstraction in software, we have code.

Everything expands as we go down that abstraction scale. It's cheaper to iterate at higher levels of abstraction. It's nearly impossible to get a design right on the first try; it usually takes multiple attempts to refine understanding of both the problem and potential solutions. Iterating by writing the code over and over would be iteration at a low level of abstraction. If we iterate at higher levels—concepts, requirements, models, prototypes—it takes less work to create each artifact on each iterative pass. So, we can iterate more quickly and more often, which gives us more chances of getting it right.

You say in your book, *Software Development Pearls*, that the project team needs flexibility around at least one of the five dimensions of scope, schedule, budget, staff, and quality. Can you explain?

Project managers know the classic iron triangle or triple constraint of project management. The colloquial statement is, "What do you want, good, fast, or cheap? Pick two." The idea is that you can't have everything that you want. I've seen the triangle drawn in multiple ways with different labels on the vertices. The most common are time, cost, and scope. Sometimes quality is included.

I think in terms of the five dimensions of scope, schedule, budget, staff, and quality. People do make tradeoffs with these against each other, including quality, all the time. They might decide to ship a product that they know is defective with the idea, rightly or wrongly, that it's better to get the product out fast than to make sure that everything works right, although customers don't always agree. I also split resources, which you sometimes see in the triangle, into budget and staff, two different aspects of resources. I've known of teams that had funding but had a headcount limitation. They couldn't hire, but they could use money in other ways, maybe outsourcing or buying a packaged solution. The idea is that there are tradeoffs people have to make, and constraints they have to work within, to succeed.

A constraint is a dimension about which you have no flexibility. The project manager has to deal with that reality. The second category a dimension could fall into is what I call a driver. A driver is a major success objective for the project that's important to achieve, but where there's a little bit of flexibility. Any dimension that's not a constraint or a driver has a degree of freedom and adjustability, and the project manager

SOFTWARE ENGINEERING RADIO

Visit www.se-radio.net to listen to these and other insightful hour-long podcasts.

RECENT EPISODES

- 524—Abi Noda, founder of Pull Panda and DX, discusses developer experience with host Brijesh Ammanath. They examine the basic concept of DX and why it matters before diving into a wide variety of issues, including methodologies for measuring DX, the main factors that influence it, and strategies for overcoming common barriers in improving DX.
- 523—Jessi Ashdown and Uri Gilad, authors of the book *Data Governance: The Definitive Guide*, discuss what data governance entails and how to implement it. Host Akshay Manchale speaks with them about why data governance is important for organizations of all sizes and how it impacts everything in the data lifecycle from ingestion and usage to deletion.
- 520—John Ousterhout, professor of computer science at Stanford University, and host Jeff Doolittle talk about his book, *A Philosophy of Software Design* (Yaknyam Press). They discuss the history and ongoing challenges of software system design, especially the nature of complexity and the difficulties in handling it.

UPCOMING EPISODES

- Randy Shoup talks with host Jeremy Jung about implementing CD at eBay.
- Host Priyanaka Raghavan and Brian Campbell discuss cryptographic defenses against stolen tokens.
- Adrian Kennard and Kevin Hones talk to host Gavin Henry about writing a network OS from scratch.

needs to know how much adjustability. The balancing point for any kind of project is to understand what's critical and what are the constraints. Project managers have to achieve the success drivers by adjusting the degrees of freedom within the limits imposed by the constraints.

What is one thing a software engineer should remember from our show?

I don't know anyone who can honestly say, "I am building software as well as software could ever be built." If you can't say that, you should always be looking for ways to improve your processes and your practices. You can't change everything at once. Individuals, groups, and organizations can absorb change only at a certain rate and still get their project work done. But you have to make the time to spend some of your effort on improvement, growth, learning, change, and experimenting. Otherwise, there's no reason to expect the next project to go any better than the last one.

One technique that worked well for me was to identify on every project one or two areas I wanted to get better at. It could be estimation, algorithm design, unit testing, whatever. Then I'd spend some of my time on that project learning about those techniques, looking for opportunities to apply them right away. You take a small productivity hit every time you do that. There's a learning curve, and then there's a price. But if I do that, in the process I'm going to improve my own capability for the rest of my career. So, I encourage software engineers and managers to adopt some kind of systematic learning philosophy, to always be carving out a certain percentage of project time for learning how to do the next project better.

GAVIN HENRY is the CIO of TelcoSwitch, founder of the recently acquired SureVoIP, and a Software Heritage Ambassador. Contact him at ghenry@surevoip.co.uk.


V7
V7

EDITORS: Sharon Broude Geva, sbroudegeva@gmail.com
Dirk Colbry, colbrydi@msu.edu

This article originally appeared in computing in SCIENCE & ENGINEERING vol. 24, no. 3, 2022

DEPARTMENT: EDUCATION

MolSSI Education: Empowering the Next Generation of Computational Molecular Scientists

Jessica A. Nash, Mohammad Mostafanejad, and T. Daniel Crawford, *Department of Chemistry, Virginia Tech, Blacksburg, VA, 24061, USA*

Ashley Ringer McDonald, *California Polytechnic State University, San Luis Obispo, CA, 93401, USA*

The Molecular Sciences Software Institute (MolSSI) is a research and education center that supports software development in the computational molecular sciences (CMS). One of MolSSI's core objectives is to provide education and training for the next generation of computational researchers. MolSSI Education targets various career stages and skill levels through its live workshops, online resources, and software fellowship program. MolSSI Education focuses its efforts on four areas: programming and software development, high-performance computing, artificial intelligence, faculty and curriculum development, and the software fellowship program. This article delineates educational efforts at the MolSSI, overall goals, and resources that can be useful to researchers in the CMS.

Computational molecular science (CMS) is a broad term that describes the application of computational resources to chemical theory and systems. It is part of many fields, including chemistry, physics, materials science, and chemical engineering, and encompasses methods such as molecular mechanics, electronic structure, and machine learning (ML). Today, computational methods are widely used in industry and academia and are indispensable parts of many scientific breakthroughs.

Traditionally, computational practitioners have been the ones responsible for building the tools of the trade, namely, software. Over the past few decades, computational researchers have developed dozens of software packages that are used by thousands of scientists worldwide. Often, the software has been seen as a byproduct of scientific research. However, as scientific problems become more complex, it is necessary for scientists to focus on software engineering, a trend that is reflected in the emergence of the Research Software Engineer. For scientific software developers, deep knowledge of programming languages can be vital for creating performant and usable software. For researchers, knowing how to program can help improve research efficiency and increase the speed of data analysis. Thus, computational scientists often need both mastery of their scientific field and competency in computing skills such as programming and software development.

The Molecular Sciences Software Institute (MolSSI) aims to enable new science by supporting software development efforts in CMS. The MolSSI was founded in 2016 with a grant from the U.S. National Science Foundation (NSF) and is a collaborative, multi-institute center. In 2021, the NSF renewed MolSSI's funding for an additional five years. MolSSI provides software infrastructure, standards, community engagement, and training and education. MolSSI's open-source software projects address several needs in the community.[1] MolSSI's training programs include workshops, summer schools, and a fellowship program for graduate students and postdoctoral researchers. Since its establishment, MolSSI has funded 95 Software Fellow projects, hosted more than

1521-9615 © 2022 IEEE
Digital Object Identifier 10.1109/MCSE.2022.3165607
Date of current version 31 August 2022.

25 community-led workshops, and reached more than 1500 students through educational events.

Our education program focuses on four areas: i) programming and software development, ii) high-performance computing (HPC), iii) faculty and curriculum development, and iv) a software fellowship program. By working in these areas, we hope to reach researchers across a broad range of skill levels and career stages. A full list of MolSSI Education resources can be accessed on the Education website.[2] For all of our training materials, we directly engage with the community. We deliver these resources through synchronous workshops and asynchronously through the Education website or YouTube channel.

MOLSSI EDUCATION COMPONENTS

Training in Programming and Software Development

The MolSSI's programming and software development resources are designed for beginner to intermediate programmers and introduce fundamental principles using examples relevant to computational molecular scientists. MolSSI's resources in this area focus on establishing programming skills and best practices for scientific software development using specific examples. Our curricula in programming and software development are created by software scientists and associates.

We lead our programming and software development workshops using a live-coding style. Live coding is a method popularized by the Software Carpentry organization and has been found to be successful in training novice programmers. In the live-coding approach to teaching, an instructor will share their screen and type code into their programming environment while explaining the thought process and reasoning behind their actions. We intersperse live coding with small challenges or exercises to allow students to apply concepts they have just learned.

MolSSI Education offers an introduction to Python programming in its flagship undergraduate workshop, Python Data and Scripting for Computational Molecular Scientists. In this workshop, students are introduced to Python syntax, working with text files, visualization, and running command-line programs. We offer this workshop synchronously once or twice a year. Registration is free and open to the public, though we typically partner with organizations that focus on undergraduate researchers such as the MERCURY Consortium.[3,4]

For students who have more experience programming or who are planning to work on software development projects, we offer the Python Package Best Practices workshop. The "Best Practices" covered in this workshop are topics recommended and often used in scientific and open-source software development, presented in a cohesive, hands-on format. Topics include version control, collaboration workflows, testing, documentation, and project structure. These are all practical skills widely used in software projects but currently rarely taught formally. Concepts are covered at a high level first, then, we demonstrate them with hands-on material. For example, when introducing software testing, we discuss the benefits and motivations of testing and also show specific examples of how one might test Python code. These workshops can be requested for groups or universities by contacting the Education team. We typically offer this workshop at least twice a year either to the public or in partnership with an academic research group.

MolSSI's other efforts in this area include a workshop on data visualization and a Python scripting workshop aimed at biochemists. In addition, MolSSI offers workshop materials on object-oriented programming and design patterns. A full list of resources is given in Table 1.

Training in HPC and Artificial Intelligence (AI)

MolSSI Education's newest initiative is in high performance computing (HPC) and artificial intelligence (AI). Our HPC and AI Education Programs consist of five major divisions: online educational resources,[2] certified university curricula, industrial training programs, instructor-led hands-on workshops, and community guidelines and best practices.[5] MolSSI Education launched this initiative in 2021 in recognition of the key focus areas of the Exascale Computing Project, top national priorities, and strategic plans.

We base our HPC online educational resources on open-source industry standards, vendors' expert recommendations, and community guidelines and best practices. Table 1 provides a high-level view of our online educational resources in HPC designed for a variety of user backgrounds and skill levels. For example, the fundamentals of the program would be most beneficial to beginner and intermediate-level users, while the homogeneous parallel programming resources are designed for intermediate and advanced users. The heterogeneous parallel programming section, on the other hand, provides online resources for all background levels.

A key component of the MolSSI Education program in HPC is industry partnerships. In collaboration with NVIDIA and Intel, we offer a series of certified instructor-led hands-on workshops. Our collaboration with NVIDIA through the Certified Instructor and University Ambassador Programs allows the members of the

TABLE 1. Online resources from the MolSSI Education Program. A superscript a ([a]) indicates that resources are available on the MolSSI Education website at education.molssi.org/resources. A superscript b ([b]) indicates resources are under development.

Topic	Resources
Programming and Software Development	• Python Data and Scripting[a] • Python Data and Scripting for Biochemistry [a] • Scientific Data Visualization using Python[a] • Best Practices in Software Development[a] • Object Oriented Programming and Design Patterns[a]
Molecular Science	• Quantum Mechanics Tools[a] • Molecular Mechanics Tools[a]
Fundamentals of HPC	• Principles of Scientific HPC[b] • RAJA and Kokkos models for abstraction and portability[b] • Slurm and Moab workload management systems and schedulers[b]
Homogeneous Parallel Programming[b]	• Message Passing Interface[b] • Shared-Memory Parallel Programming with OpenMP[b]
Heterogeneous Parallel Programming[b]	• Fundamentals of Heterogeneous Parallel Programming with CUDA C/C++[a] • A Systematic Approach to CUDA C/C++ Parallel Programming[b] • Applications of Heterogeneous Programming in Computational Molecular Sciences[b] • OneAPI: A Unified Approach to Heterogeneous Parallel Programming[b] • Modern Platforms for Heterogeneous Parallel Programming[b]

community to have free access to certified training programs that otherwise would involve a registration fee. These programs are divided into four major specialization areas: Data science, deep learning, accelerated computing, and conversational AI. Intel, on the other hand, enabled us to offer a variety of training modules within two main focus areas: Essentials of data-parallel C++ (DPC++) and basics of OpenMP offload. For our certified instructor-led hands-on workshops, university courses, and industrial training programs, we provide the registered users with access to a cloud virtual machine armed with a variety of accelerator architectures, such as CPUs, GPUs, and FPGAs. The users can access all training resources using either the command line or JupyterLab's user-friendly notebooks.

The last component of MolSSI's HPC and AI and Education initiative involves establishing best practices for data management. Due to the dire need of the scientific software community for improving the quality of the scientific data management plans according to FAIR principles[6] and to ensure the reproducibility, interoperability, and replicability of the computational research products, we have developed a public platform[5] for publishing community guidelines and best practices for all domains of CMS. We have found this platform upon our years of experience in serving and the open-source software communities, hundreds of interviews and surveys gathered from our discovery project, expert recommendations, and best practices documents provided by the major vendors, and our collaborations with the -funded projects, such as the XPERT Network.[7]

Faculty and Curriculum Development

MolSSI's efforts in faculty and curriculum development focus on helping faculty members to incorporate resources into their classes and promoting the use of programming in the chemistry curriculum. Although MolSSI Education resources were originally designed to enable students to more effectively participate in research experiences, there have been a notable number of faculties using the MolSSI Education materials as a starting point to develop curricular resources to incorporate programming into their courses. Examples of these types of curricular innovations are highlighted in's recent ACS Symposium Series Book.[8] The examples in this book incorporate programming in a variety of different classes, ranging from general chemistry to graduate-level courses. Programming is used to analyze data, make visualizations demonstrate physical phenomena that are otherwise hard to describe, solve chemistry problems numerically, and more.

The MolSSI has partnered with faculty professional development groups, such as Enhancing Science Course by Integrating Python (ESCIP)[a] and Psi4Education[b] to work with faculty to develop curricular resources. ESCIP is a group of faculty from the Cottrell Scholars Program, sponsored by Research Corporation for Science Advancement, who develop and share a curriculum that utilizes Python programming for chemistry, physics, and math courses. Psi4Education is the education and outreach program of the quantum chemistry software package Psi4 that uses Psi4's Python interface, to create lab activities for use across all levels of the chemistry curriculum. MolSSI supports these curricular development efforts by meeting with these faculty development groups to advise them on strategies and best practices for teaching programming. MolSSI also sponsors symposia and workshops for faculty at conferences, such as the Biennial Conference on Chemical Education, and hosts instructor training workshops to help faculty upskill so they can better teach best programming practices to their students.

In addition, MolSSI is involved in the University Ambassador Program with the NVIDIA. Through this partnership, MolSSI has access to teaching kits designed for university courses, which include syllabi, lecture notes, curricular resources, and programming activities that cover major topics in HPC, deep learning, and robotics. MolSSI can support faculty in implementing these courses at their institutions and help them customize the contents of each course in the domain to meet their program goals and requirements.

MolSSI Software Fellowship Program

MolSSI provides direct support to software development efforts of early-career researchers through its Software Fellowship program. Software Fellowships are highly selective awards that fund graduate students and postdocs who develop software infrastructure for CMS. MolSSI prioritizes projects of broad interest with potential high impact to the communities. As of 2022, the Software Fellowships are year-long awards, with calls for applications starting in February and awards starting in July.

Each cohort of software fellows receives special training in a week-long Software Fellow Bootcamp. The Bootcamp curriculum includes topics in software development best practices, software design and distribution, and special topics related to the projects and interests of the software fellow cohort. The Bootcamp material draws heavily from the Education resources featured in this paper, particularly the Python Package Best Practices and Software Design workshops.

Throughout the fellowship, the software fellows receive one-on-one mentoring from a software scientist who provides guidance for the software fellow's project.

[a]https://escip.github.io/
[b]https://psicode.org/posts/psi4education/

CONCLUSION

The MolSSI Education Team is engaged with, and welcomes comments and collaborations from members of the CMS and scientific software development communities. MolSSI disseminates all new updates and the latest releases to the educational resources through its social media accounts, newsletter, and the organizational website.[2,9] All of our educational resources, including the hands-on tutorials, instructor-led workshop materials, and self-paced online courses are free and available under open-source licenses. Members of the scientific community can provide direct feedback, comments, or contributions to all educational resources through opening discussions, issues, and pull requests on the Education repositories.[10] We continuously measure the impact of our online educational resources through tracking the user analytics gathered from the hosting websites and repositories.

ACKNOWLEDGMENTS

The MolSSI is funded by the National Science Foundation under grant CHE-2136142.

REFERENCES

1. MolSSI Software Projects. [Online]. Available: https://molssi.org/software-projects/
2. MolSSI Education. [Online]. Available: https://education.molssi.org/
3. MERCURY Consortium—molecular education and research consortium in undergraduate computational chemistry. [Online]. Available: https://mercuryconsortium.org/
4. A. R. Mcdonald *et al.*, "Building capacity for undergraduate education and training in computational molecular science: A collaboration between the MERCURY consortium and the Molecular Sciences Software Institute," *Int. J. Quantum Chem.*, vol. 120, no. 20, Jul. 4, 2020, doi: 10.1002/qua.26359.
5. MolSSI guidelines, checklist, and best practices. [Online]. Available: https://molssi.github.io/molssi-guidelines/
6. M. D. Wilkinson *et al.*, "The FAIR Guiding Principles for scientific data management and stewardship," *Sci. Data*, vol. 3, Art. no. 160018, 2016, doi: 10.1038/sdata.2016.18.

7. P. Barakhshan *et al.*, "Exchanging best practices and tools for supporting computational and data-intensive research," The Xpert Network, 2021, doi: 10.48550/ARXIV.2102.09373.
8. A. R. McDonald and J. Nash, "Teaching Programming Across the Chemistry Curriculum," 2021. doi: 10.1021/BK-2021-1387.
9. The Molecular Sciences Software Institute. [Online]. Available: https://molssi.org/
10. MolSSI Education GitHub. [Online]. Available: https://github.com/molssi-education
11. ESCIP | Enhancing Science Courses by Integrating Python (ESCIP) is funded by the Research Corporation for Scientific Advancement through the Cottrell Scholars Collaborative program. [Online]. Available: https://escip.github.io/
12. *Psi4Education: Computational Labs Using Free Software*. [Online]. Available: https://psicode.org/posts/psi4education/

JESSICA A. NASH is with the Department of Chemistry, Virginia Tech, Blacksburg, VA, 24061, USA. She is a software scientist and the education lead at MolSSI, Blacksburg, VA, 24060. Contact her at janash@vt.edu.

MOHAMMAD MOSTAFANEJAD is with the Department of Chemistry, Virginia Tech, Blacksburg, VA, 24061, USA. He is the lead software scientist in HPC education, AI, and ML at the MolSSI, Blacksburg, VA, 24060. Contact him at smostafanejad@vt.edu.

T. DANIEL CRAWFORD is the director of the MolSSI, Blacksburg, VA, 24060, USA, as well as the university distinguished professor and ethyl chair of the Department of Chemistry, Virginia Tech, Blacksburg, VA, 24061. Contact him at crawdad@vt.edu.

ASHLEY RINGER MCDONALD is an associate professor with the Department of Chemistry and Biochemistry, California Polytechnic State University, San Luis Obispo, CA, 93401, USA. She serves on the board of directors of the MolSSI, Blacksburg, VA, 24060, USA, as the co-director for education, training, and faculty development. Contact her at armcdona@calpoly.edu.

EDITOR: Irena Bojanova, NIST, irena.bojanova@computer.org

This article originally appeared in Computer vol. 55, no. 6, 2022

Teaching Modeling in the Time of Agile Development

Oscar Pastor, *Polytechnic University of Valencia*
Alfonso Pierantonio, *University of L'Aquila*
Gustavo Rossi, *La Plata National University and National Scientific and Technical Research Council*

We discuss modeling in the context of agile software development and reflect on how we, as educators, can use modeling to help improve agile practices.

Engineers (in particular, software engineers) have been motivated to find a way to confront the intrinsic and increasing complexity of their endeavors. In this sense, models have been used for centuries to abstract different aspects of a system under construction. As pointed out in Gogolla and Selic[6] and Ludewig,[8] models can describe certain aspects of a system and fundamentally act as understanding and communication means (descriptive models); they can be used to analyze and predict system properties (predictive models), and they can be employed as an implementation specification (prescriptive models). A good summary of different definitions can be found in Muller et al.[10]

In this context, model-driven engineering (MDE)[9] emerged as a discipline within software engineering, which considers models first-class citizens throughout the software process and aims to derive running applications directly and automatically from models (in general, by using transformations). MDE evolved rapidly during the 2000s, and it is considered key to success in many domains, such as railway systems, automotive applications, business process engineering, and embedded systems.[1] It simplifies software construction since developers can abstract from most technological decisions (for example, interface devices) and implementation aspects and focus on domain details. It has been shown that MDE helps improve productivity[9] and code quality.[2] However, the state of practice suggests that models are still far from being considered essential software artifacts.

Abstraction practice is strictly required for managing complexity and developing correct software.[18] Notably, putting abstraction in practice in software engineering terms means modeling. Since modeling is an essential skill for software developers, the development, manipulation, management, and comprehension of models is a relevant learning objective.[7] Therefore, educators have addressed this problem for years. In this article, we elaborate on this issue in the context of agile software development. After some brief comments on the state of practice in modeling and the way we teach about models, we reflect on how we, as educators, can help improve current agile practices.

STATE OF PRACTICE

The history of software engineering shows how abstraction is the fundamental development notion: from machine code and assemblers to programming language support, the level of abstraction is always evolving from a lower one to a higher one. Models should be the natural next step. As happens in other engineering disciplines, more abstraction should lead to better software production methods, making software engineering an accurate process. Surprisingly, this is not the case. Software engineering is too frequently perceived as closer to a (technological) handicraft-centered activity, strongly dependent on skilled programmers, not expert modelers.

Digital Object Identifier 10.1109/MC.2022.3144929
Date of current version: 3 June 2022

As early as 1971, Teichroew and Sayani[16] stated that "the size, importance and cost of systems building provide an opportunity for the investigation of ways to improve the (software production) process." More than 50 years later, programming environments have constantly progressed, but current practice for design, programming, and testing activities still relies on substantial manual effort. Models should facilitate the automation of the systems building process. Why is that goal not being achieved? Is it unreachable? Is it worthless, assuming that conventional (not model-driven) software production is enough? If we talk about improving abstraction to better conceptualize and accurately represent reality in a software application, shouldn't a sound use of modeling be independent of what software development process is selected? These are the questions we discuss in this work.

Many authors have surveyed the extent to which modeling and MDE are used in industry (see, for example, Gorschek et al.,[12] Heldal et al.,[13] and Hutchinson et al.[14]). Specifically, Heldal et al.[13] analyze when descriptive and prescriptive models are used. While there is certainly a niche for MDE (in Heldal et al.,[13] the survey covers huge companies working on embedded systems), the mainstream use of agile approaches (which emerged as a counterpart of monolithic ones) did not come with a similar prevalence of the use of models (even descriptive ones). This mismatch is strongly reflected in developers, as discussed in Gorschek et al.[12] The reasons might have originated in corporate practices, such as discouraging the use of models because of an extreme interpretation of one of agile's principles: "Value working software over comprehensive documentation." This interpretation has also hampered the introduction of user-centered development approaches in the agile universe.[15] However, as discussed in the following, it might also be a product of problems in the education of developers.

MODELING AND MDE EDUCATION

We can get a good idea of why, what, and how we teach about modeling by reading the proceedings of educator symposiums held with the Association for Computing Machinery/IEEE International Conference on Model-Driven Engineering Languages and Systems (MODELS) (see, for example, the MODELS 2021 Educators Symposium webpage[11]). Specifically, from Kuzniarz and Börstler[7] we learn that we must include modeling in the curriculum "to encourage and stimulate thinking at high abstraction levels ... to enable and ensure successful development of software" and "to be competitive in the labor market." Ciccozzi et al.[5] survey the way we teach modeling and MDE. Besides interesting findings about course content and tools, negative aspects were explored. From our point of view, two of them are remarkable: 1) the lack of maturity of existing tools and 2) students having difficulty understanding abstraction and conceiving modeling as quite different from programming. While we, as educators, have almost no control over the former, we should improve our practices to deal with the latter, especially considering how it might impact industry practices.

ENGINEERS (IN PARTICULAR, SOFTWARE ENGINEERS) HAVE BEEN MOTIVATED TO FIND A WAY TO CONFRONT THE INTRINSIC AND INCREASING COMPLEXITY OF THEIR ENDEAVORS.

APPROACHES AND IDEAS

Agile development intends to improve the software production process, which is necessarily linked to getting better abstraction capabilities. Models provide the right answer. Beyond using models for communication and to facilitate understanding, they should be the key artifact that guides software development, providing as many automation facilities as possible to connect abstract descriptions with their associated software representations. As suggested in Bucchiarone et al.,[4] we believe that the agile development wave provides an opportunity to revisit the model-driven main goal, facilitating the design of a software production process where enterprise models and software applications are conceptually aligned through the construction of the right models and transformations. Given that models are essential to conceptualize components that must be represented in a software system, modeling correctly and building the right ones should become the most important software engineering activity, which is naturally related to teaching.

BEYOND USING MODELS FOR COMMUNICATION AND TO FACILITATE UNDERSTANDING, THEY SHOULD BE THE KEY ARTIFACT THAT GUIDES SOFTWARE DEVELOPMENT.

Modeling properly implies training software engineers in abstracting and conceptualizing correctly. Building the right models requires making software engineers become aware of the languages that apply to different abstraction levels [for example, Business Process Modeling Notation for business process models, i* for goal-oriented requirements models, Unified Modeling Language (UML) class diagrams for system structure models, and so on]. One of the main inhibitors of modeling in practice is the lack of a well-established "teaching modeling" body of knowledge, a gap that is only partly filled by the work in Burgueño et al.[3] Modeling is about abstracting, and how to teach and assess how good a student is at it is not simple. It means evaluating how well he or she conceptualizes, which requires skills and abilities to grasp insights and knowledge blurred in the intricacy of the application domain.

This modeling dimension should be on top of programming as an essential topic in software engineering teaching. Some problems that need to be precisely solved to achieve this objective include the following:

1. *Foundations*: We need these to have a universal, widely accepted and used definition of what a model is, providing a precise definition that is ontologically well grounded.
2. *Better tooling*: This is important to reinforce efficient, adequate, flexible, usable, and reliable (in other words, mature) tool support, facilitating the use of models in software production and making it feasible to use conceptual programming-based tools, where models go beyond a merely communicational dimension, becoming a trustworthy software artifact (as discussed in Embley et al.[17]).
3. *Revised syllabi*: These would enable us to assess and rethink how we teach abstraction and modeling. Most syllabi treat these subjects in a perhaps unrelated way. Consequently, the assimilation of abstraction (and its practice) is not consistently pursued and, to a certain degree, depends on students' attitudes and curiosity. One possibility is to use early courses on object orientation to introduce modeling instead of (only) programming. Also, as suggested in Ciccozzi et al.,[5] we should try to use project-oriented and hands-on learning. The availability of tools that permit round-tripping between code and models [for example, Visual Paradigm (https://www.visual-paradigm.com)] would make students perceive models as a functional part of their projects alongside the code and not a way to procrastinate what they are mainly interested in: coding, coding, and coding! As mentioned, the lack of educational modeling tools with reduced accidental complexity should also be addressed since tooling is another critical issue.

Emphasizing the relevance of modeling in software engineering teaching must consider that modeling abilities among students should play a crucial role. Correctly abstracting a real system that is represented in a computer requires advanced conceptualization skills, which is not easy to convey and evaluate. Some students seem readier to do it well than others, but in any case, a software engineering student should not get a degree without possessing a solid modeling ability. Such difficulties can be mitigated by replanning the way abstraction is taught.

It is not just a student issue. Modeling is hard to teach. Good practices should be widely accepted and employed by educators. Assessing the syntax and semantic quality of models should become a task supported by a sound ontological commitment. Even delimiting the set of rules that a (simple) UML class diagram should follow (as a kind of correction guide) remains an open question (for example, simple aspects, such as whether classes without attributes and associations with the same names should be allowed, do not have precise, definitive answers). This represents a significant issue, as there is broad diversity among modeling languages.

From a more "tactical" point of view, we can profit from the popularity of low-code platforms to demonstrate how models and model-driven development (usually "hidden" behind visual editors, as indicated in Bucchiarone at al.[1]) improve the productivity, quality, and cost effectiveness of software development. Finally, we must prove how informal modeling (sketches, wireframes, and so on) can be easily accommodated in the agile cycle to improve communication, understanding, and agreement, as suggested in Bucchiarone at al.[1] Whether this will be a successful strategy is not easy to say. What is worth remarking is that what makes a programmer a good software engineer is the ability to use abstraction fruitfully—and the state of the art in software abstraction is MDE.

REFERENCES

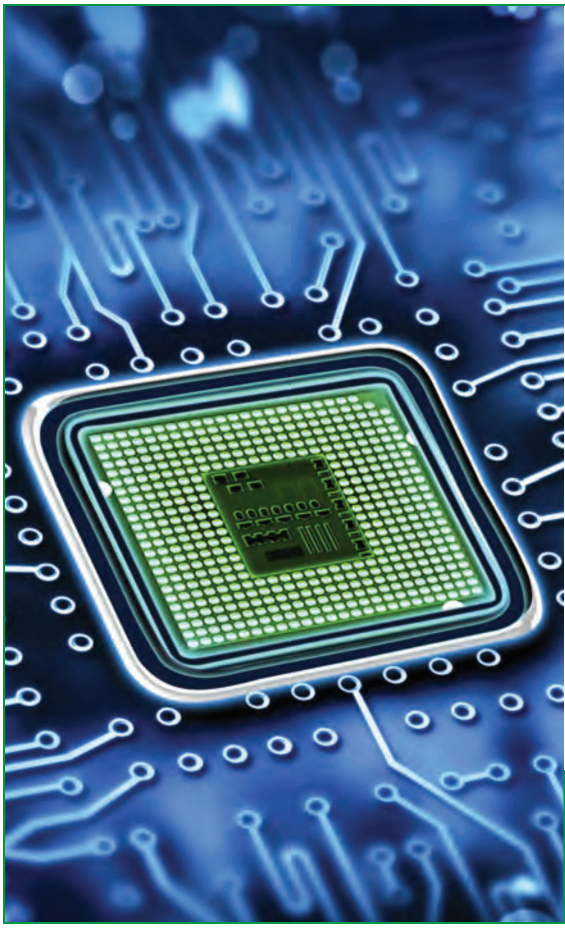
1. A. Bucchiarone *et al.*, "What is the future of modeling," *IEEE Softw.*, vol. 38, no. 2, pp. 119–127, 2021, doi: 10.1109/MS.2020.3041522.
2. J. I. Panach *et al.*, "Evaluating model-driven development claims with respect to quality: A family of experiments," *IEEE Trans. Softw. Eng.*, vol. 47, no. 1, pp. 130–145, 2021, doi: 10.1109/TSE.2018.2884706.
3. L. Burgueño *et al.*, "Contents for a model-based software engineering body of knowledge," *Softw. Syst. Model.*, vol. 18, no. 6, pp. 3193–3205, 2019, doi: 10.1007/s10270-019-00746-9.
4. A. Bucchiarone, J. Cabot, R. F. Paige, and A. Pierantonio, "Grand challenges in model-driven engineering: An analysis of the state of the research," *Softw. Syst. Model.*, vol. 19, no. 1, pp. 5–13, 2020, doi: 10.1007/s10270-019-00773-6.
5. F. Ciccozzi *et al.*,"How do we teach modeling and model-driven engineering? A survey," in *Proc. 21st ACM/IEEE Int. Conf. Model Driven Eng. Lang. Syst., Companion*, 2018, pp. 122–129, doi: 10.1145/3270112.3270129.
6. M. Gogolla and B. Selic, "On teaching descriptive and prescriptive modeling," in *Proc. 23rd ACM/IEEE Int. Conf. Model Driven Eng. Lang. Syst., Companion*, 2020, pp. 1–9, doi: 10.1145/3417990.3418744.
7. L. Kuzniarz and J. Börstler, "Teaching modeling: An initial classification of related issues," in *Proc. Electron. Commun. EASST 7th Educator's Symp.*, 2011, pp. 1–10.
8. J. Ludewig, "Models in software engineering: An introduction," *SoSyM*, vol. 2, no. 1, pp. 5–14, 2003, doi: 10.1007/s10270-003-0020-3.
9. M. Brambilla, J. Cabot, and M. Wimmer, "Model-driven software engineering in practice," in *Synthesis Lectures on Software Engineering*, 1st ed., L. Baresi, Ed. San Rafael, CA, USA: Morgan & Claypool, 2012.
10. P.-A. Muller, F. Fondement, B. Baudry, and B. Combemale, "Modeling modeling modeling," *Softw. Syst. Model.*, vol. 11, no. 3, pp. 347–359, 2012, doi: 10.1007/s10270-010-0172-x.
11. "Educators symposium," in *Proc. ACM/IEEE Int. Conf. Model Driven Eng. Lang. Syst. Companion (MODELS-C)*, Fukuoka, Japan, Oct. 10–15, 2021. [Online]. Available: https://conf.researchr.org/track/models-2021/models-2021-educators-symposium
12. T. Gorschek, E. Tempero, and L. Angelis, "On the use of software design models in software development practice: An empirical investigation," *J. Syst. Softw.*, vol. 95, p. 193, Sep. 2014, doi: 10.1016/j.jss.2014.03.082.
13. R. Heldal, P. Pelliccione, U. Eliasson, J. Lantz, J. Derehag, and J. Whittle, "Descriptive vs prescriptive models in industry," in *Proc. ACM/IEEE 19th Int. Conf. Model Driven Eng. Lang. Syst., (MoDELS)*, 2016, pp. 216–226, doi: 10.1145/2976767.2976808.
14. J. Hutchinson, J. Whittle, and M. Rouncefield, "Model-driven engineering practices in industry: Social, organizational and managerial factors that lead to success or failure," *Sci. Comput. Program.*, vol. 89, pp. 144–161, Sep. 2014, doi: 10.1016/j.scico.2013.03.017.
15. G. Cockton, M. Lárusdóttir, P. Gregory, and Å. Cajander,

"Integrating user-centered design in agile development," in *Human-Computer Interaction Series*. Cham: Springer-Verlag, 2016, pp. 1–46.
16. D. Teichroew and H. Sayani, "Automation of system building," *Datamation*, vol. 17, no. 16, pp. 25–30, 1971.
17. D. W. Embley, S. Liddle, and O. Pastor, "Conceptual-model programming: A manifesto," in *Handbook of Conceptual Modeling*, D. Embley and B. Thalheim, Eds. Berlin: Springer-Verlag, 2011, pp. 3–16.
18. E. W. Dijkstra, "Chapter EWD227: Stepwise program construction," in *Selected Writings on Computing: A Personal Perspective*. New York, NY, USA: Springer-Verlag, 1982, pp. 1–14.

OSCAR PASTOR is the director of internationalization and transference at the Valencian Research Institute for Artificial Intelligence, Universitat Politecnica de Valencia, Valencia, 46009, Spain. Contact him at opastor@dsic.upv.es.

ALFONSO PIERANTONIO is a full professor at the University of L'Aquila, L'Aquila, 67100, Italy. Contact him at alfonso.pierantonio@univaq.it.

GUSTAVO ROSSI is a professor at La Plata National University, La Plata, Argentina, and a researcher at the National Scientific and Technical Research Council, Buenos Aires, 1900, Argentina. Contact him at gustavo@lifia.info.unlp.edu.ar.



IEEE COMPUTER SOCIETY

Publications Seek 2025 Editors in Chief

Application Deadline: 1 March 2024

The IEEE Computer Society seeks applicants for the position of editor in chief for the following publications:

- *Computer* magazine
- *Computing in Science and Engineering* magazine
- *IEEE/ACM Transactions on Computational Biology and Bioinformatics*
- *IEEE Computer Architecture Letters*
- *IEEE Intelligent Systems* magazine
- *IEEE Open Journal of the Computer Society*
- *IEEE Transactions on Big Data*
- *IEEE Transactions on Cloud Computing*
- *IEEE Transactions on Sustainable Computing*

Computer Society publications are the cornerstone of professional activities for our members and the community we serve. We seek candidates who are IEEE members in good standing, have strong familiarity with our publications, and possess an excellent understanding of the field as it relates to academic, industry, and governmental areas. Applicants must also have successful experience attracting and developing a diverse team of talented and respected individuals to serve key editorial board roles. Demonstrated managerial skills are also required, as they are necessary to ensure rich content and issue development, along with timely processing of submissions through the editorial cycle. Terms begin 1 January 2025.

For complete information on how to apply, please go to
www.computer.org/press-room/seeking-2025-editors-in-chief

Apply Today!

IEEE COMPUTER SOCIETY

IEEE

Conference Calendar

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. Questions? Contact conferences@computer.org.

FEBRUARY

1 February

- BICE (Black Issues in Computing Education Symposium), Santo Domingo, Dominican Republic

5 February

- AIMHC (IEEE Int'l Conf. on Artificial Intelligence for Medicine, Health and Care), Laguna Hills, USA
- CDKE (IEEE Int'l Conf. on Conversational Data & Knowledge Eng.), Laguna Hills, CA, USA
- ICSC (IEEE Int'l Conf. on Semantic Computing), Laguna Hills, USA

18 February

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

MARCH

2 March

- CGO (IEEE/ACM Int'l Symposium on Code Generation and Optimization), Edinburgh, UK
- HPCA (IEEE Int'l Symposium on High-Performance Computer Architecture), Edinburgh, UK

11 March

- PerCom (IEEE Int'l Conf. on Pervasive Computing and Communications), Biarritz, France

12 March

- SANER (IEEE Int'l Conf. on Software Analysis, Evolution and Reengineering), Rovaniemi, Finland

16 March

- VR (IEEE Conf. on Virtual Reality and 3D User Interfaces), Orlando, USA

17 March

- SSIAI (IEEE Southwest Symposium on Image Analysis and Interpretation), Santa Fe, New Mexico, USA

APRIL

9 April

- SaTML (IEEE Conf. on Secure and Trustworthy Machine Learning), Toronto, Canada

17 April

- COOL CHIPS (IEEE Symposium in Low-Power and High-Speed Chips), Tokyo, Japan

22 April

- VTS (IEEE VLSI Test Symposium), Tempe, Arizona, USA

23 April

- PacificVIS (IEEE Pacific Visualization Symposium), Tokyo, Japan

29 April

- DCOSS-IoT (Int'l Conf. on Distributed Computing in Smart Systems and the Internet of Things), Abu Dhabi, United Arab Emirates

MAY

1 May

- MOST (IEEE Int'l Conf. on Mobility, Operations, Services and Technologies), Dallas, USA

5 May

- ISPASS (IEEE Int'l Symposium on Performance Analysis of Systems and Software), Indianapolis, USA

6 May

- CCGRID (IEEE Int'l Symposium on Cluster, Cloud and Internet Computing), Philadelphia, USA
- HOST (IEEE Int'l Symposium on Hardware Oriented Security and Trust), Tysons Corner, Virginia, USA

13 May

- ICCPS (ACM/IEEE Int'l Conf. on Cyber-Physical Systems), Hong Kong
- ICDE (IEEE Int'l Conf. on Data Eng.), Utrecht, The Netherlands
- RTAS (IEEE Real-Time and Embedded Technology and Applications Symposium), Hong Kong

20 May

- SP (IEEE Symposium on Security and Privacy), San Francisco, USA

21 May

- ISORC (IEEE Int'l Symposium

on Real-Time Distributed Computing), Tunis, Tunisia

27 May
- FG (IEEE Int'l Conf. on Automatic Face and Gesture Recognition), Istanbul, Turkey
- ICST (IEEE Conf. on Software Testing, Verification and Validation), Toronto, Canada
- IPDPS (IEEE Int'l Parallel and Distributed Processing Symposium), San Francisco, USA

28 May
- ISMVL (IEEE Int'l Symposium on Multiple-Valued Logic), Brno, Czech Republic

JUNE

3 June
- ICHI (IEEE Int'l Conf. on Healthcare Informatics), Orlando, USA

4 June
- ICSA (IEEE Int'l Conf. on Software Architecture), Hyderabad, India
- WoWMoM (IEEE Int'l Symposium on a World of Wireless, Mobile and Multimedia Networks), Perth, Australia

10 June
- ARITH (IEEE Symposium on Computer Arithmetic), Malaga, Spain

16 June
- CVPR (IEEE/CVF Conf. on Computer Vision and Pattern Recognition), Seattle, USA

17 June
- SVCC (Silicon Valley Cybersecurity Conf.), Seoul, South Korea

19 June
- CHASE (IEEE/ACM Conf. on Connected Health: Applications, Systems and Eng. Technologies), Wilmington, USA

24 June
- DSN (IEEE/IFIP Int'l Conf. on Dependable Systems and Networks), Brisbane, Australia
- MDM (IEEE Int'l Conf. on Mobile Data Management), Brussels, Belgium
- RE (IEEE Int'l Requirements Eng. Conf.), Reykjavik, Iceland

25 June
- CAI (IEEE Conf. on Artificial Intelligence), Singapore

26 June
- CBMS (IEEE Int'l Symposium on Computer-Based Medical Systems), Guadalajara, Mexico

27 June
- CS (IEEE Cloud Summit), Washington, DC, USA (Hybrid)

29 June
- ISCA (ACM/IEEE Annual Int'l Symposium on Computer Architecture), Buenos Aires, Argentina

JULY

1 July
- COMPSAC (IEEE Annual Computers, Software, and Applications Conf.), Osaka, Japan
- ICALT (IEEE Int'l Conf. on Advanced Learning Technologies), Nicosia, Cyprus

3 July
- IOLTS (IEEE Int'l Symposium on On-Line Testing and Robust System Design), Rennes, France

7 July
- SERVICES (IEEE World Congress on Services), Shenzhen, China

8 July
- CSF (IEEE Computer Security Foundations Symposium), Enschede, Netherlands
- EuroS&P (IEEE European Symposium on Security and Privacy), Vienna, Austria

15 July
- ICME (IEEE Int'l Conf. on Multimedia and Expo), Niagara Falls, Canada
- SCC (IEEE Space Computing Conf.), Mountain View, USA

24 July
- ASAP (IEEE Int'l Conf. on Application-specific Systems, Architectures and Processors), Hong Kong

AUGUST

25 Aug
- HCS (IEEE Hot Chips Symposium), Stanford, USA