OUTLOOK AND FUTURE DIRECTIONS

19

Maritta Heisel[*], **Rami Bahsoon**[†], **Nour Ali**[‡], **Bruce Maxim**[§], **Ivan Mistrik**[¶]

[*]*Universität Duisburg-Essen, Germany* [†]*University of Birmingham, UK* [‡]*University of Brighton, UK*
[§]*University of Michigan-Dearborn, USA* [¶]*Independent Researcher, Germany*

The last decade has seen a paradigm shift in software engineering due to the need of software to process huge amounts of data. This book has extensively discussed several approaches that make use of cloud systems and the processing of big data. However, the current situation will not be the end of the story. The development towards ever larger quantities of data has just begun. Big data is not only about the volume of data but also about the efficiency, velocity, value, and other aspects. The resources of the cloud promise to achieve big data characteristics. This situation will lead to even more and new challenges for software architecture in the context of big data and cloud computing.

In the following, we discuss new trends, challenges and opportunities concerning the future use of big data and clouds, and their impact on software architecture. Being able to handle huge amounts of data and to share computing resources in a cloud make it possible to offer entirely new applications to society. Existing applications will advance and will have the potential to become much more powerful than they are today. In Section 19.1, we highlight some of these new or advanced applications. In Section 19.2 we discuss advances of technologies that support advances in big data applications. In addition, a number of architecturally significant requirements (see Chapter 1 of this book) will be crucial to handle big data and cloud applications in the future. These requirements and their relevance for software architecture are discussed in more detail in Section 19.3. To achieve such requirements, in turn, the architecting processes that are followed in the development of new and advanced applications have to be adjusted accordingly. Section 19.4 presents some of the challenges that architectural processes will face in the future. Of course, this chapter cannot completely cover all of the trends, challenges and opportunities that will shape the future of big data and clouds. In line with its high importance, this topic continuously gains attention. Section 19.5 points readers to other works for further reading.

19.1 NEW OR ADVANCED APPLICATIONS

In the future, gigabytes will not be a relevant quantity of data anymore. Instead, larger chunks of data will be processed on a regular basis, such as terabytes and petabytes. Billions (not only millions) of devices will be connected to the Internet and constantly emit data that will have to be collected and processed. This makes it possible to offer entirely new applications for individuals as well as enterprises, for example, sophisticated ambient-assisted living and remote healthcare systems. Other applications

Software Architecture for Big Data and the Cloud. DOI: 10.1016/B978-0-12-805467-3.00019-3

such as social networks and recommender systems will become much more widespread or much more powerful, because the amount of data that can be handled will increase by orders of magnitude. Even today, we can see that our daily lives have changed considerably, because portable devices are continuously connected to the Internet. We constantly upload data, we stay in contact with friends and relatives, and we use more and more sophisticated web services. This is only possible because enormous amounts of data are produced and processed. In the future, new and advanced applications will change our lives even more profoundly. In the following, we briefly describe several advanced or new applications. Further applications (as sources of big data) are enumerated by Yang et al. [8].

Lifelogging. Today, only a relatively limited number of people use activity trackers to measure their body functions (called lifelogging) and obtain statistics about their health and fitness. In the future, this number will grow dramatically. Many existing apps provide advice for a healthy style of living based on data they gather. Such data are not only of value to their producers, but also to third parties. For example, health insurance companies could offer special rates to those users who are willing to share their health data on a regular basis. On the one hand, such applications can contribute much to the well-being of individuals and significantly reduce the costs needed for health systems. On the other hand, however, such applications come with severe risks for privacy.

Ambient-assisted living. In societies where the percentage of elderly people is growing, there is a huge demand to enable elderly people to live at home as long as possible. In ambient-assisted living, the homes of elderly people are equipped with sensors that can detect health problems of a person, for example, when a person falls to the ground and cannot get up again. In such a case, a healthcare provider could be notified automatically. Furthermore, appliances could react to verbal commands, supporting elderly people in their daily activities. Such systems will be of great benefit to individuals and the society; however, they have to be trustworthy.

Internet of Things. The Internet of Things (IoT) is characterized by the fact that sensor networks connect incredible numbers of small devices to the Internet. These devices send their data in regular intervals. All of that sensor data have to be transmitted, (partially) stored, filtered, cleaned, and processed, i.e., big data management techniques have to be used. These techniques will have to be further developed. In particular, to make sense of orders of magnitude more data than is available today will be an extraordinary technological challenge. Based on the sensor data, appropriate actions have to be launched. Such actions depend on the context. For example, the owner of a house could be notified that the stove is turned on even if nobody is at home, or the traffic lights in a city can be controlled according to the overall volume of traffic in the city.

Smart Xs. The Internet of Things is a prerequisite for the advent of smart grids, smart homes, smart cities, smart engineering, and other “smart” applications. For example, smart grids allow consumers of electricity to control their electric appliances so that they are turned on when the price for electricity is low. Furthermore, contracts with electricity providers can be changed frequently, optimizing the cost for power consumption. Smart grids are beneficial because they reduce the overall power consumption and cost of power consumers, but also come with a privacy risk. Smart cities and engineering plants optimize traffic and production flows, but can also be subject to attacks. Apart from compromising security, cyber-attacks can also produce safety risks. When critical infrastructures break down or hospitals are made inoperable, human lives are at stake. Therefore, resilience is an important requirement for smart Xs.

Social Networks. Billions of people use social networks today, and their number will rise in the future. Already today, a study indicates that only the spouse knows a person better than Facebook does. With more data being collected about individuals and better possibilities for analyzing that data, severe privacy issues may arise that have to be addressed. Unfortunately, users of social network sites do not have as much influence on what happens to their data as this would be desirable. A challenge for the future is to empower users, providing them with means to better protect their privacy and to exercise more control over their data.

Recommender Systems. Such systems, even though common already today, will grow in importance and also in power in the future. The amount of data on which a recommendation can be based will grow drastically. However, this does not necessarily mean that the recommendations become more reliable. Recommender systems have to be developed further in such a way that users can better assess the credibility of the given recommendations. Furthermore, recommender systems should be able to adapt to their users, taking characteristics of their personality into account. For such systems (as for social networks), psychology can and should inform software development.

Personalized services. Future cloud and big data technologies will allow personalized services for many people, and the automation of large parts of public services and infrastructures. Such services come with service-level agreements (SLA) that must be met. These may concern performance, but also security and privacy. Measures must be taken at runtime if a service violates the SLA. This involves to develop and apply mechanisms to constantly monitor the fulfillment of the SLA and also mechanisms to replace a service with another one when the SLA is violated. Replacing services is a highly nontrivial task that needs powerful component infrastructures.

Rapid market development. Enterprises will be able to react to changing market conditions much faster than today, because there are more data on which decisions can be based on, and these data can be analyzed in almost real time despite of their size. For example, insurance companies will be able to offer personalized contracts to different groups of people, and to adjust such contracts on a daily basis. The contracts are based on proprietary information such as insurance cases of the company, e.g., how many damage adjustments the insurance company had to make for cases of burglary. In the future, such information will be combined with other sources, for example, police data about committed burglaries in a certain area. In this way, insurance companies can assess the risk for damage adjustments in a much more fine-grained and reliable way than this is possible today.

Homeland security. Our society feels more and more threatened by terrorists and criminals. Public video surveillance will become more widespread than is the case today. Video surveillance produces huge amounts of data, and such data will have to be analyzed automatically. For example, if a video shows a person committing a crime, then that person should be recognized automatically as soon as he or she enters the range of another video camera. US Customs are experimenting with IT-based systems to identify people “who wish to do harm or violate U.S. laws” (see https://www.dhs.gov/obim-biometric-identification-services) when they enter the country. In the future, such systems will become more reliable, because of a broader database and the usage of more sophisticated learning algorithms. There are also attempts to predict the occurrence of crimes. For example, to predict where burglaries will be committed in the next few days, or even using demographic data to predict if an individual will commit a crime in the near future. However, such applications bear a risk of false accusations. Society has to find ways to deal with this problem.

19.2 ADVANCED SUPPORTING TECHNOLOGIES

The innovative applications discussed in Section 19.1 can only be developed successfully when there is also innovation in the underlying supporting technologies.

Machine learning. Machine learning will play a major role in aggregating data to information that humans can interpret. The vast amount of available data also supports machine learning algorithms to work successfully. Sophisticated data analytics algorithms need machine learning techniques. The challenge here is that the learning process must take place in a distributed environment [8].

Data visualization. The importance of data visualization will increase, as organizations relying on big data technology require faster access to useable and comprehensible presentations of the results of data analytics. According to Assunção et al. [2], visualization is needed to support descriptive, predictive, and prescriptive data analysis. These different tasks call for different kinds of visualization approaches. Visualization should not be hampered because of transferring the data to be visualized over a network, which leads to performance issues.

System Interfaces. Big data analytics is challenging and time consuming, requiring expensive software, enormous computational infrastructure and effort. Standards will need to be developed along with new APIs (application programming interfaces) to allow practitioners to switch among analytic products and data sources. New languages will have to be developed to describe data in such a way that big data analytics is supported best. Steps in this direction are the Predictive Model Markup Language (PMML) and the Google Prediction API, as discussed by Assunção et al. [2].

Computing platforms. The most important big data computing paradigm is MapReduce and its implementation Hadoop. With this paradigm, big data processing has become possible at all. MapReduce is an instance of the "data at rest" principle, where data are stored and different queries can be processed on that data. Complex event processing, on the other hand, is an instance of the "data in flight" principle, where the query on the data is fixed, and the query is evaluated on the incoming data, without the need to store the data. Further development of novel computing platforms can be expected, combining the "data at rest" and "data in flight" principles.

Cloud platforms and deployments. Due to their enormous storage requirements, big data applications often make use of cloud platforms. Such platforms may be different in nature (e.g., public or private), but usually offer a set of services, concerning, for example, the configuration of applications, data management, and security. We can expect that those services will be enhanced and augmented in the future. However, the diversity of the services makes it difficult to migrate between different cloud platforms. The question how a migration can be achieved smoothly will deserve more attention in the future.

NoSQL data bases. In the past, relational databases were the most common ones. With a unified data model and the common structured query language SQL, they offered a functionality that met the demands of many applications involving data management. However, relational data bases are not suitable for treating data that stem from various sources and do not adhere to a unique data model. Therefore, NoSQL databases have been developed. They can cope with diverse data and distribution, and they support hyperscalability (see Section 19.3), but come with proprietary APIs (application programming interfaces), which makes porting applications more difficult. Among the new kinds of NoSQL databases that have already been developed are document databases, key-value databases, column-oriented data bases, and graph databases [3,4]. It can be expected that more such novel kinds of data bases will emerge in the future, revolutionizing the way in which big data can be stored and retrieved.

New Cloud Paradigms. Apart from Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), new cloud paradigms have already emerged. They include Data as a Service, Analytics as a Service, Model as a Service, and Storage as a Service [2]. We can imagine that more kinds of useful cloud services will be developed in the future. They will free cloud customers from many tasks that they have to carry out themselves currently, which will lead to an increase in developer productivity and foster a much larger adoption of cloud computing in the future than is currently the case.

Risk management. Big data and cloud computing come with significant risks. Some of them are mentioned in Section 19.1. First and foremost, privacy and security risks have to be mentioned here. However, there also is a considerable risk to draw false conclusions from big data analytics, which puts the entire goal of the analytics in question. Thus, entire business models may turn out to be invalid. Therefore, risk management must accompany big data and cloud applications during their entire lifetime. Risk is composed of the likelihood of an unwanted incident and the negative consequences of that unwanted incident on an asset. Risk management consists of risk identification, risk evaluation, and risk assessment. If an identified risk is assessed to be unacceptable, appropriate risk reduction measures have to be taken. While risk management is common in the context of safety and security-critical systems, it is not yet common to apply it to big data analytics.

Privacy and Security Mechanisms. To reduce privacy and security risks, appropriate mechanisms have to be applied. Currently, such mechanisms consist mainly of encryption or access control. For the novel applications coming with big data and clouds, these simple mechanisms will not suffice any more. For example, homomorphic encryption is a relatively new mechanism that makes it possible to process encrypted data, without the need to decrypt the data before processing. However, the computations that are possible on the encrypted data today are very simple. To support big data analytics on encrypted data, more research is necessary. There also is a need to develop better mechanisms for user empowerment. An example are sticky policies. Here, a security policy is attached to critical data. Such a policy may demand that the data are deleted after some time. However, to date there are no infrastructures available to enforce such policies.

19.3 ARCHITECTURALLY SIGNIFICANT REQUIREMENTS

The novel applications and the further development of their supporting technologies as discussed in the previous sections have far-reaching consequences for advancing the state for the art in software architecture. They result in a number of architecturally significant requirements, as introduced in Chapter 1 of this book. Such requirements are quality requirements (also called nonfunctional requirements), whose fulfillment have an influence on the software architecture of an IT-based system. The architecturally significant requirements are not new; most of them have been discussed extensively in this book. Here, we highlight the importance of these requirements for future developments.

The five Vs. These characteristics of big data, namely Volume, Velocity, Variety, Veracity, and Value, have been drivers for new technologies in recent years. For example, the Variety property spawned the development of NoSQL data bases. Focusing on such characteristics leads to fruitful new research and also new architectural solutions. In the future, it is to be expected that further crucial characteristics of big data and clouds will be discovered (Assunção et al. [2], for example, name “Viability” as a new V) that will lead to further progress in big data and cloud technologies.

Hyper-scalability. As discussed in Chapter 2 of this book, a crucial property of big data systems is their ability to scale in much more extreme ways than this was necessary (and possible) before big data processing came up. Hyperscalable systems are able to support an exponential growth in computing requests even though the available resources only grow linearly. This is an amazing property, which became possible due to a number of principles, such as automation, optimization, simplification, and observability (see Chapter 2). In the future, more mechanisms will have to be found and applied in order to preserve hyperscalability in the presence of the ever-increasing growth rates of big data.

Distribution. Most of the novel applications mentioned above are distributed and rely on the Internet. The distribution of data – though necessary – entails a number of problems that have to be dealt with and need innovation in the future. In Section 19.2, we already mentioned that learning algorithms need to function on distributed data, and that the visualization of distributed data can cause performance problems. Even though distributed data do not suffer from a single point of failure or attack, media report on successful cyber-attacks, involving data theft, or distributed denial-of-service (DDoS) attacks almost every week. However, the existence of enterprises relies on the availability of their services. Client–server systems as they are common today are not robust enough against failure of parts of the Internet, due to, for example, DDoS attacks or censorship by governments. More resilient ways of distribution are necessary, for example, by using peer-to-peer architectures where appropriate. Such architectures are used only sparsely today, because of the large communication overhead involved. A clever combination of client–server and peer-to-peer architectures could lead to more robust services.

Adaptivity. Almost all of the new or advanced applications will need to be adapted to different users and changing circumstances. MAPE-K [5] is an architectural blueprint for self-adaptive systems. It stands for Monitor-Analyze-Plan-Execute plus Knowledge. A monitor component observes events that happen in the environment of the system. The analysis component interprets these events. This is the basis for planning appropriate actions, which are determined by the planning component. The execute component is responsible for enacting the generated plan. All components make use of a common knowledge base. To instantiate such an architectural blueprint in the context of big data and clouds is challenging because of the amount of data to be taken into account for adaptation and the distributed nature of the data sources.

Resilience. In the future, our society will be entirely dependent on functioning computing power and software. Critical infrastructures will be connected to the Internet, and hence can be subject to attacks or failures. Therefore, future systems need to be much more resilient than this is the case today. This involves the need to identify attacks or other adverse circumstances, and to adapt or otherwise react to those circumstances. Developing resilient software architectures will be one of the major challenges for the future.

Trustworthiness. This property is (among others) related to reliability, resilience, security, and performance. It means that users of a system can justifiably place trust in it. For different systems, different properties may be necessary to achieve trustworthiness. For example, in an ambient-assisted living system, it is important that the system correctly identifies health problems of the elderly person and notifies the health care provider in a timely manner, always preserving the privacy of the person. However, trustworthiness cannot be just reduced to these other properties. We also need components in the software architecture that demonstrate to the users that the trustworthiness properties are indeed fulfilled. This is especially difficult for machine learning systems whose specific behavior cannot be predicted from their design only but relies data fed into the learning process as well.

Privacy and Security. While the development of new technologies allowing better services and more automation is certainly welcome, there are also some risks coming with these new technologies that should be taken into account and treated accordingly by research, industry, and politics right from the beginning. With more data being collected about individuals and better possibilities for analyzing that data, severe privacy issues may arise that have to be addressed. But not only privacy, but also security and safety may be at stake. When important public and industrial facilities are connected to the Internet, they will be subject to attacks. The same is true for data stored in clouds. The results of big data processing are only reliable when the underlying data are correct and reliable, and in particular have not been tampered with by attackers. Apart from compromising security, cyber-attacks can also produce safety risks. Therefore, privacy, security, safety, and trust should receive more attention from software architects and data scientists than has been the case in the past. As far as software architecture is concerned, not only software components implementing cryptographic mechanisms or firewalls are needed. Users will need to be empowered to a much larger extent than today. This will involve specific user interface components allowing users to grant or withdraw consent to collect and process their data, informing users about processing, use, and visibility of their data. Since different users have different needs, such components must be personalized and also should be able to adapt to the user (see Adaptivity). Enterprises, in turn, are threatened in their existence by cyber-attacks. Especially when they use clouds, they are no longer in full control of their assets. Security mechanisms must be incorporated in cloud architectures that give reliable security guarantees for cloud customers.

Standards and Certification. Cloud providers must be able to provide credible assurance that they protect their customers' data according to best practices. Today, the ISO 27000 series of standards requires enterprises to develop and maintain a so-called information security management system (ISMS). Such an ISMS not only covers software, but the entire organization of the enterprise. However, establishing an ISMS also needs to implement security controls, thus influencing the software architecture of the software systems used by the enterprise. Currently, only few enterprises are certified according to the ISO 27000 series or other security standards, such as the Common Criteria (ISO/IEC 15408). In the future, this will change drastically, not only because of compliance requirements arising from legislation (see next point), but also because a growing number of attacks will raise the demand for security guarantees.

Compliance. Today, we see a trend toward new regulations and laws concerning data security and privacy. For example, the European Union has passed a new data protection regulation to better protect the privacy of individuals. Furthermore, it is discussed if enterprises will be obliged to report attacks on their IT. Hence, being compliant to the legislation will oblige enterprises and governmental organizations not only to implement protection mechanisms, but also to demonstrate that they have acted according to best practices. Compliance issues have an influence on software architecture, for example, because logging components will have to be integrated in the architecture.

19.4 CHALLENGES FOR THE ARCHITECTING PROCESS

Having identified the architecturally significant requirements that play a role in big data and cloud applications in the future, we now consider the challenges architecting processes will need to cope with.

Systems of Systems and their emergent properties. The use of clouds and big data will go hand in hand with the development of Systems of Systems (SoS) that combine independently developed smaller systems to bigger and more powerful ones with increased complexity and emergent behavior. Such SoS have a high demand of flexibility and adaptability, depending on large amounts of data which are stored in a distributed way. Future applications will be much more complex than they are today. This will result in the need of more advanced engineering methods. For example, the architecting process must take into account how the different subsystems can interoperate to achieve a common goal ("glue architecture"). Furthermore, the emergent properties of the composition must be determined, and it must be decided if they are desirable or not.

Balancing and integrating different architecturally significant requirements. Even though most of the above-mentioned requirements will be relevant for the majority of the novel applications coming with big data and clouds, not all of them will be of equal importance, and some of them might be negligible. Hence, the architecting process must balance the architecturally significant requirements to arrive at an optimal architecture. Performance, for example, conflicts with most other architecturally significant requirements. Therefore, it will not be possible just to add components to the architecture, addressing the different architecturally significant requirements. Instead, clever strategies for reconciling conflicting requirements and approaches for optimizing architectures will be needed.

Evolutionary development. In the future, more applications will make use of big data and clouds. This will involve problems of migration to a different database technology or between different computing and cloud platforms. Hence, software architects have to address the challenge of supporting the migration process. Evolutionary development could be an approach to do so. Here, iterative and incremental development processes are applied. It is to be investigated which aspects of an application are likely to change and which ones are not. The software architecture then should reflect this assessment, because evolutionary development is only successful when changes are cheap and easy to accomplish.

Architecting for hyper and unbounded scale for cloud and big data architecture. Cloud infrastructure, which is essentially as a high-performance service-oriented computing paradigm, is continuously evolving. New computing paradigms which leverage the cloud's capabilities such as Edge, Fog, distributed and federation models, microservices, and the Internet of Things (IoT) have been emerging. As described in Chapter 1 of this book, new architecture significant requirements that are cloud related result from the shared environment, its hyperconnectivity, and continuous evolution. Cloud architecture significant requirements call researchers and practitioners to rethink their practices, which should be aware of cloud fundamentals that relate to elastic computing, dynamism and autoscaling, multitenancies of the cloud, value-driven design for utilizing economies of scale, SLA-centric design in support for multitenancies, to name but a few. The awareness of these requirements challenges the way we systematically architect software systems that operate on the cloud – partially or in whole; interface with other cloud-based services and/or part of the evolving cloud ecosystem. These requirements essentially affect the way we define processes which are cloud-specific; where design and architecture processes should be steered by environment uncertainties, risks mitigation strategies, trade-offs, likely evolution of the services and application ecosystem itself, and the inviting unbounded scale. This transforms the architecting design process into a globally distributed and decentralized exercise, where architecture knowledge and unified design decisions are difficult to incept, elaborate, evaluate, trace and negotiate for conflicts, risks and trade-offs. The challenge calls for cloud-centric architecting processes and tooling support that facilitate architecting in the "wild" and leverage on diversity and wealth of inputs to reach sound and efficient architecture solutions for cloud-based applications and services. In particular, we need to

- define processes which are cloud-aware and embrace decentralization and distribution of the architecting process;
- evaluate architecture design decisions, where cloud-specific properties are core determinants for solutions;
- define cloud-specific architecture knowledge and tooling support which leverage the benefits of diversity, multiperspectives of inputs, etc.;
- provide tooling support for managing and tracing design decisions in the “wild”;
- define styles and family of patterns which are suited for given distribution and decentralization contexts.

Multitenancy and service-level agreement-centric architecture solutions. As discussed in Chapter 1, cloud architecturally significant requirements do have measurable effects and can be observed on Services Level Agreements (SLA) compliance and violations. The choice of architecture design decisions should be essentially SLA-aware; where SLA promises should mandate design decisions and choices. Conversely, architecture design decisions and choices need to inform, refine and elaborate SLAs. The process is continuous and is not restricted to the development and predeployment phases. It should be “live” during the operation and evolution of the cloud-software system. The process is intertwined and interleaved and needs to be informed by requirements of various multitenants. The challenge is that the process has to reconcile various multitenant requirements and their constraints, while not compromising the overall “welfare” of the shared environment. This makes architecting rather complex exercise and calls for new approaches for architecting for multitenancy taking into account multiple users in a given tenancy and across multiple ones. Architects need to consider the diverse needs, wishes, context, requirements, and quality-of-services constraints within and across tenants. Architects need to predict what may seem unpredictable from likely changes in requirements, within and across tenants. They also need to consider how changes that relate to adapting to new environments that relate to the cloud, such as mobility, Fog, Edge, Internet of Things (IoT), and federation, can affect design decisions and their evolution over time. Architects need to formulate design decisions and tactics that are flexible enough to cater for continuous changes in users’ requirements in a single tenancy and across multitenancy. This calls for novel architecture-centric frameworks that elicit, define, model, evaluate, and realize the commonality, variability, veracity, diversity, and scale of both functional and nonfunctional requirements supporting the individuals, tenants, and the operating environment. These solutions should also provide mechanisms for ensuring fairness and preventing “greed” in situations where providers stretch the economies of scale, by accommodating more than what the environment can normally handle.

Architecting for the unbounded data scale. As explained in Chapter 1, the unbounded scalability of the cloud and its various service layers have “accidentally” lead to “data farming”, due to the volume, veracity, and variety of data accumulated and/or assimilated across various service cloud layers and constituent architectural components. There are two fundamental aspects that have changed the way data needs to be processed and thus needs a complete paradigm shift: the size of data has evolved to the amount that it has become intractable by existing data management systems, and the rate of change in data is so rapid that processing also needs to happen in real-time. With analysts estimating data storage growth at 30% to 60% per year, organizations must develop a long-term strategy to address the challenges of managing projects that analyze exponentially growing data sets with predictable linear costs [Ian Gorton, March 7, 2016 – SEI, Pittsburgh]. Henceforth, architecture processes and design decisions shall have managing data, its growth evolution and decay in the heart of the inception, elaboration, evaluation, and operations processes. The process shall provide systematic treatment for architecturally significant requirements that are data related. The treatment shall align the organization's strategies, their long-term business objectives and priorities with the technical decisions for the way data management is designed as a first-class architecture entity. In particular, the architecting process shall explicitly look at effective, efficient and scalable solutions for the way data are assimilated, aggregated, classified and at their intelligent data analysis. Concerns which relate to data security, privacy, scalability, performance, availability, and integrity and associated trade-offs are among the architecture significant requirements that influence architecture design decisions. The choice of data models, their portability, and expressiveness become another first-class concern in the architecting process. The choice is particularly important not only for managing data but for unlocking its potentials in enabling new applications.

Architecting for ubiquitous cloud-based and big data driven systems. The volume of data, especially machine-generated data, is exploding; it is remarkable how fast that data is growing every year, with new sources of data that are emerging. For example, in the year 2000, 800,000 petabytes of data were stored in the world, and it is expected to reach 35 zettabytes by 2020 (according to IBM). Social media plays a key role. Twitter generates 7+ terabytes of data every day. Facebook, 10 terabytes. Mobile devices play a key role as well, as there were estimated 6 billion mobile phones in 2011. Driven by the needs to handle ever growing data collection and integrate diverse applications into systems of systems, scalability is the dominant quality attribute factor for many contemporary software architectures [4]. Architecting for scalability, while ensuring ubiquity and privacy, continues to be one of the challenges that confront architects. This calls for defining and/or extrapolating architecture styles and patterns which have been successful in embracing ubiquity while promising to deliver on other nonfunctional trade-offs.

Ethical-centric architecting for privacy and security. With more data being collected about individuals and better possibilities for analyzing that data, severe privacy issues may arise that have to be addressed as of the architecting phase. The widespread (and often unauthorized) collection of data poses several ethical concerns. Software architects need to become more concerned about how data will be used within the solution and/or within the cloud ecosystem. The architecting inception, elaboration and construction processes should give explicit attention to the benefits and risks of exploiting data and its modalities in the connected world. Architects need to weigh the risks and benefits of exploiting the data; their practices and architecture design choices should seriously look at the potentials of big data practices to further increase the divide between technology haves and have nots. As already mentioned, cyber-attacks can also produce safety risks. Therefore, privacy, security, safety, and trust should require greater attention from software architects and data scientists than it has in the past. Academic computing programs will be pressured to raise the awareness for issues that relate to ethical-centric architecting, architecting for privacy and security and related trade-offs. This cannot be underestimated as a strategy in educating future architects and software engineers for modern distribution paradigms.

19.5 FURTHER READING

Recently, a number of publications discussing future trends and challenges for big data and clouds have appeared. Not all of them focus on software architectures. We summarize some of these papers to give the interested reader hints for further information on this topic.

Assunção et al. [2] discuss trends and future directions for big data computing and clouds. They give a thorough overview of data management techniques, including the different "Vs", data storage, integration, processing, and resource management. In the context of model building and scoring, they mention "Data as a Service" as a new cloud computing paradigm, and new languages and APIs to support data analytics. Furthermore, they discuss the role of visualization and user interaction, as well as potential new business models from "Analytics as a Service."

Anagnostopoulos et al. [1] focus on big data, taking a more technical point of view. They describe and compare several Hadoop-based platforms according to the possibility to perform real-time analytics, support for data integration, and supported application domains. Then they systematically analyze the challenges that are relevant for the different tasks in big data management, namely data cleansing, acquisition and capture; storage, sharing and transfer; analysis and collection of results. Ethical considerations are also taken into account.

Nasser and Tariq [6] discuss data challenges (with respect to the different "Vs" and other properties, namely quality, discovery and dogmatism), process challenges, (with respect to the different big data management activities) and management challenges (privacy, security and governance). Furthermore, they discuss possible solutions based on a layered reference architecture called big data technology stack, consisting of seven layers. They contrast different ways to introduce big data technologies in an enterprise, namely the revolutionary, evolutionary, and hybrid approaches.

Gorton and Klein [4] address big data from a software architectural point of view. After discussing NoSQL data bases and the importance of scalability, they enumerate common requirements for big data systems, namely write-heavy workloads, variable request loads, computation-intensive analytics, and high availability. Finally, they show how existing architectural tactics can be adjusted for big data systems. This is achieved by considering three different architectural levels, namely data, distribution, and deployment.

Yang et al. [8] give a comprehensive description of big data and cloud systems, and the corresponding innovation opportunities and challenges. They show how the five "Vs" can be addressed with cloud computing techniques. They identify different sources of big data. They discuss big data technology challenges including data storage, transmission, management, processing, analysis, visualization, integration, architecture, security, privacy challenges, and quality. They sketch the relevant technology landscape, and relate the different technologies to the challenges described before. An elaborate research agenda covers many aspects, from technical ones such as distributed data storage to more cross-disciplinary ones, such as interdisciplinary collaboration. Spatiotemporal aspects play an important role throughout.

Sasi Kiran et al. [7] discuss issues and challenges on big data in cloud computing, based on a multitenant system model with different levels of resource sharing. As future challenges, they identify the problems to find an optimal architecture for an analytics system, guaranteeing the statistical relevance of the analytics results, and distributed data mining.

REFERENCES

[1] I. Anagnostopoulos, S. Zeadally, E. Exposito, Handling big data: research challenges and future directions, J. Supercomput. 72 (2016) 1494, http://dx.doi.org/10.1007/s11227-016-1677-z.

[2] M.D. Assunção, R.N. Calheiros, S. Bianchi, M.A.S. Netto, R. Buyya, Big data computing and clouds: trends and future directions, J. Parallel Distrib. Comput. 79–80 (2015) 3–15.

[3] I. Gorton, J. Klein, Designing scalable software and data architectures, Tutorial at ICSE 2014, 2014.

[4] I. Gorton, J. Klein, Distribution, Data, Deployment: Software Architecture Convergence in Big Data Systems, Technical Report, SEI, May 2014.

[5] J.O. Kephart, D.M. Chess, The vision of autonomic computing, Computer (January 2003).

[6] T. Nasser, R.S. Tariq, Big data challenges, Comput. Eng. Inform. Technol. 4 (3) (2015).

[7] J. Sasi Kiran, M. Sravanthi, K. Preethi, M. Anusha, Recent issues and challenges on big data in cloud computing, IJCST 6 (2) (April–June 2015).

[8] C. Yang, Q. Huang, Z. Li, K. Liu, F. Hu, Big data and cloud computing: innovation opportunities and challenges, Int. J. Digital Earth 10 (1) (2017) 13–53, http://dx.doi.org/10.1080/17538947.2016.1239771.