Highlights

Terramycelium: A Domain-driven Reference Architecture for Big Data Systems

Author One, Author Two, Author Three

- Research highlight 1
- Research highlight 2

Terramycelium: A Domain-driven Reference Architecture for Big Data Systems

Author One[a], Author Two[b], Author Three[a,b]

[a]*Department One, Address One, City One, 00000, State One, Country One*
[b]*Department Two, Address Two, City Two, 22222, State Two, Country Two*

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Keywords: keyword one, keyword two
PACS: 0000, 1111
2000 MSC: 0000, 1111

1. Introduction

The advent of the internet and widespread use of digital devices have sparked a profound shift in connectivity and data creation, leading to an era marked by a rapid growth in data. This period is characterised by the extensive expansion of data, which presents difficulties for traditional data processing systems and necessitates inventive methods in data architecture [3, 22]. The vast amount, variety, and rapid generation of data in the current digital environment necessitate innovative solutions, particularly in the field of Big Data (BD).

Data needs have dramatically evolved, transitioning from basic business intelligence (BI) functions, like generating reports for risk management and compliance, to incorporating machine learning across various organisational facets [5]. These range from product design with automated assistants to personalised customer service and optimised operations. Also, as machine

learning becomes more popular, application development needs to change from rule-based, deterministic models to more flexible, probabilistic models that can handle a wider range of outcomes and need to be improved all the time with access to the newest data. This evolution underscores the need to reevaluate and simplify our data management strategies to address the growing and diverse expectations placed on data.

Currently, the success rate of BD projects is low. Recent surveys have identified the fact that current approaches to big data do not seem to be effectively addressing these expectations. According to a survey conducted by [11], only 13% of organisations are highly successful in their data strategy. Additionally, a report by NewVantage Partners reveals that only 24% of organisations have successfully converted to being data-driven, and a measly 30% have a well-established big data strategy. These observations, additionally corroborated by research conducted by McKinsey & Company (analytics2016age) and Gartner (Nash), emphasise the difficulties of successfully using big data in the industry. These difficulties include the lack of a clear understanding of how to extract value from data, the challenge of integrating data from multiple sources, data architecture, and the need for skilled data analysts and scientists.

Without a well-established big data strategy, companies may struggle to navigate these challenges and fully leverage the potential of their data. One effective artefact to overcome some of these challenges is Reference Architectures (RAs) [8]. RAs extract the essence of the practice as a series of patterns and architectural constructs and manifest it through high-level semantics. This allows stakeholders to refrain from reinventing the wheel and instead focus on utilising existing knowledge and best practices to harness the full potential of their data. While there are various BD RAs available to help practitioners design their BD systems, these RAs are overly cetnralised, lack attention to cross-cutting concerns such as privacy, security, and metadata, and may not effectively handle the proliferation of data sources and consumers.

To this end, this study presents TerrMycelium, a distributed RA designed specifically for BD systems with a focus on domain-driven design. TerrMycelium seeks to surpass the constraints of current RAs by utilising domain-driven and distributed approaches derived from contemporary software engineering. This method aims to improve the ability of BD systems to scale, be maintained, and evolve, surpassing the constraints of traditional monolithic data architectures.

The paper is structured as follows: Section 2 provides an overview of the foundational concepts and technologies pertinent to BD reference architecture, aiming to forge a conceptual framework that is required for this paper. An overview of the existing research on the topic is presented in Section 3. The significance of reference architectures in the context of big data is explored in Section 4. Section 5 details the software and system requirements necessary for implementing the proposed architecture. Section 6 delves into the theoretical foundation underpinning the challenges in contemporary big data systems. The design and development of the TerrMycelium artifact are described in Section 7. Section 8 examines the evaluation findings, their implications, limitations, and relevance to existing and future research. Finally, Section 9 summarizes the main contributions of the study, its practical implications, and suggests directions for future research.

2. Background

This section provides foundational definitions essential for comprehending the nuances of the research. This chapter aims to create the conceptual framework necessary to understand the terminology used in the thesis.

2.1. What is Big Data?

To define BD within the scope of this research, various academic definitions have been examined. Kaisler et al. [14] define BD as "the amount of data which is beyond technology's capability to store, manage and process efficiently". Srivastava [26] state that BD pertains to "the use of large data sets to handle the collection or reporting of data that serves various recipients in decision making".

Sagiroglu and Sinanc [25] describe BD as "a term for massive data sets having large, more varied and complex structure with the difficulties of storing, analyzing and visualizing for further processes or results".

Drawing from these definitions, BD in this research is conceptualised as the endeavour to discern patterns from vast amounts of data for the objectives of advancement, governance, and predictive analysis in domain-specific applications.

2.2. The Value of Big Data

The significance and value derived from BD remain pronounced [4]. Extensive discussions on the concept permeate reports, statistics, researches,

and conferences [7]. Notably, prominent companies like Google, Facebook, Netflix, and Amazon have propelled this momentum with substantial investments in BD initiatives [23].

A compelling illustration of the tangible benefits that BD offers can be seen in the Netflix Prize recommender system. This system capitalized on a diverse array of data sources, including user queries, ratings, search terms, and various demographic indicators [1]. By implementing BD-powered recommendation algorithms, Netflix not only achieved a considerable increase in TV series consumption but also observed certain series experiencing up to a fourfold surge in viewership [1].

In a healthcare context, the Taiwanese government adeptly merged its national health insurance database with customs and immigration datasets as part of a BD strategy [28]. The resulting real-time alerts during clinical visits, informed by clinical symptoms and travel history among other factors, facilitated proactive identification of potential COVID-19 cases. Such strategic data-driven initiatives significantly bolstered Taiwan's effectiveness in managing the epidemic.

In the realm of energy exploration, Shell harnesses BD to optimise the decision-making process and reduce exploration costs [17]. By uploading and comparing data from various drilling sites globally, decisions pivot towards locations that mirror those with confirmed abundant resources. Prior to BD's integration, identifying energy resources presented formidable challenges. Traditional exploration methods, which relied heavily on deciphering waves of energy travelling through the earth's crust, were not only error-prone but also exorbitantly expensive and time-intensive.

Similarly, Rolls Royce capitalises on BD's potential by collecting intricate performance data from sensors fitted on its aircraft products [17]. Such data, transmitted wirelessly, provides insights into key operational phases, from take-off to maintenance. Leveraging this wealth of information, Rolls Royce can more accurately detect degradation, enhance diagnostic and prognostic accuracy, and effectively reduce false positives.

2.3. Reference Architectures

RAs have emerged as pivotal elements in contemporary system development, guiding the construction, maintenance, and evolution of increasingly complex systems [8]. They offer a clear depiction of the essential components of a system and the interactions necessary to realize overarching objectives.

This clarity fosters the creation of manageable modules, each addressing distinct aspects of complex problems, and provides a high-level platform for stakeholders to engage, contribute, and collaborate.

The significance of RAs in IT is underscored by the success of widely adopted technologies like OAuth [19] and ANSI-SPARC architecture [2], which have their origins in well-structured RAs. These RAs not only define the qualities of a system but also shape its evolution. While every system inherently possesses an architecture, RAs distinguish themselves by focusing on more abstract qualities and higher levels of abstraction. They aim to capture the essence of practice and integrate well-established patterns into cohesive frameworks, encompassing elements, properties, and interrelationships.

The significance of RAs in BD is multifaceted, encompassing aspects like communication, complexity control, knowledge management, risk mitigation, fostering future architectural visions, defining common ground, enhancing understanding of BD systems, and facilitating further analysis.

2.4. Microservices and Decentralised, Distributed Architectures

Microservices architecture, representing an evolution in software engineering, involves structuring applications as a collection of loosely coupled services [6]. This approach, emerging from the broader concept of Service Oriented Architectures (SOA), focuses on developing small, independently deployable modules that collaborate to form a comprehensive application. As Newman [18] elucidates, microservices enhance scalability, facilitate continuous deployment, and foster a more agile development environment. They enable teams to develop, deploy, and scale parts of a system independently, thus improving overall system resilience and facilitating rapid adaptation to changing demands.

Decentralised and distributed architectures are integral to the modern computing landscape, characterised by systems spread across multiple nodes, often in different geographic locations. This architectural style, as highlighted by Richards [24], mitigates the limitations of traditional monolithic structures, offering enhanced scalability, fault tolerance, and flexibility. In distributed systems, data and processing are dispersed across multiple nodes, which interact with each other to perform tasks, as discussed by Coulouris et al. [10]. Decentralisation in this context implies the lack of a single controlling node, instead opting for a more democratic and resilient network structure.

The convergence of microservices within these architectures represents a progressive step in software engineering. It reflects a move towards systems that are not only distributed in nature but also modular and adaptable. This architectural approach aligns well with contemporary demands for systems that are scalable, resilient, and capable of leveraging the distributed nature of modern computing environments. The adoption of microservices in decentralised, distributed architectures heralds a new era in software development, where flexibility, scalability, and resilience are paramount.

3. Related Work

This section reviews seminal works in BD RAs, delineating their scope, methodologies, and inherent limitations, thereby justifying the novel approach of this study's domain-driven distributed RA, *Terramycelium*.

Lambda architecture [15] and Kappa architecture [16] represent pivotal industry contributions to BD RAs, introducing foundational paradigms for data processing. However, these architectures have faced criticism for their lack of comprehensive data management strategies, particularly regarding data quality, security, and metadata [3]. This gap is further evidenced in domain-specific RAs like those proposed by [21] for healthcare, which, while addressing domain-specific needs, often overlook cross-cutting concerns such as privacy and interoperability.

Academic efforts, as in [27] and [20], have aimed to broaden the conceptual understanding of BD systems, proposing RAs that attempt to encapsulate more holistic views of data analytics ecosystems. Yet, these proposals frequently fall short in addressing the dynamic and distributed nature of modern data landscapes, particularly in terms of scalability and modifiability.

The limitations of current BD RAs, as summarized in the works of Ataei and Litchfield, highlight a common trend: a pronounced reliance on monolithic data pipeline architectures. This reliance is manifest in the inadequacy of existing RAs to effectively manage data quality, security, privacy, and metadata. Furthermore, the monolithic nature of these architectures often results in scalability and modifiability issues, as they struggle to adapt to the evolving data and technology landscapes.

Against this backdrop, *Terramycelium* emerges as a novel RA for BD systems. By embracing a domain-driven, distributed approach, *Terramycelium*

addresses the critical limitations identified in existing RAs. Unlike its predecessors, which often encapsulate data management within rigid, monolithic structures, *Terramycelium* advocates for decentralized data stewardship and a modular architecture. This approach not only enhances scalability and flexibility but also ensures that cross-cutting concerns such as security, privacy, and data quality are inherently integrated into the system's design.

In essence, *Terramycelium* represents a significant departure from traditional BD RAs. By prioritizing domain-driven design and distributed processing, it offers a scalable and adaptable framework that aligns more closely with contemporary data management needs and the principles of modern software architecture. In doing so, *Terramycelium* not only addresses the limitations of existing RAs but also positions itself as a vanguard in the evolution of BD system design, paving the way for future research and development in this critical field.

4. Why Reference Architectures

Conceptualisation of the system as an RA, helps with understanding of the system's key components, behaviour, composition and evolution of it, which in turn affect quality attributes such as maintainability, scalability and performance [9].

Therefore, RAs can be a good standardisation artefact and a communication medium that not only results in concrete architectures for BD systems, but also provide stakeholders with unified elements and symbols to discuss and progress BD projects.

The practice of leveraging RAs for both system conceptualisation and as a standardisation artefact is not new to practitioners of complex systems. In Software Product Line (SPL) development, RAs are utilised as generic artifacts that are instantiated and configured for a particular domain of systems [12].

In software engineering, renowned IT corporations such as IBM have consistently advocated for RAs, considering them exemplary practices in addressing intricate system design challenges [8]. Similarly, in the realm of international standards, RAs frequently serve as tools to standardize emerging domains.

Morover, the BS ISO/IEC 18384-1 RA for service-oriented architectures [13] demonstrates the utility of RAs in creating standardized frameworks in specific fields.

5. Software and System Requirements

6. Theory

7. Artifact

8. Discussion

9. Conclusion

References

[1] Amatriain, X., 2013. Beyond data: from user information to business value through personalized recommendations and consumer science, ACM. pp. 2201–2208. doi:10.1145/2505515.2514691.

[2] ANSI, A., 1975. X3/sparc study group on dbms, interim report. SIGMOD FDT Bull 7.

[3] Ataei, P., Litchfield, A., 2020. Big data reference architectures: A systematic literature review, in: 2020 31st Australasian Conference on Information Systems (ACIS), IEEE. pp. 1–11. doi:10.5130/acis2020.bf.

[4] Ataei, P., Litchfield, A., 2022. The state of big data reference architectures: a systematic literature review. IEEE Access .

[5] Ataei, P., Litchfield, A., 2023. Towards a domain-driven distributed reference architecture for big data systems, in: AMCIS 2023.

[6] Bucchiarone, A., Dragoni, N., Dustdar, S., Lago, P., Mazzara, M., Rivera, V., Sadovykh, A., 2020. Microservices. Science and Engineering. Springer .

[7] Chen, H., Chiang, R.H., Storey, V.C., 2012. Business intelligence and analytics: From big data to big impact. MIS quarterly 36, 1165. doi:10.2307/41703503.

[8] Cloutier, R., Muller, G., Verma, D., Nilchiani, R., Hole, E., Bone, M., 2010a. The concept of reference architectures. Systems Engineering 13, 14–27. doi:10.2514/6.2017-5118.

[9] Cloutier, R., Muller, G., Verma, D., Nilchiani, R., Hole, E., Bone, M., 2010b. The concept of reference architectures. Systems Engineering 13, 14–27.

[10] Coulouris, G., Dollimore, J., Kindberg, T., 2005. Distributed Systems: Concepts and Design. 4 ed., Addison-Wesley.

[11] technology review insights in partnership with Databricks, M., 2021. Building a high-performance data organization. URL: https://databricks.com/p/whitepaper/mit-technology-review-insights-report.

[12] Derras, M., Deruelle, L., Douin, J.M., Levy, N., Losavio, F., Pollet, Y., Reiner, V., 2018. Reference architecture design: a practical approach, in: 13th International Conference on Software Technologies (ICSOFT), SciTePress-Science and Technology Publications. pp. 633–640.

[13] ISO, I., 2016. Information technology — reference architecture for service oriented architecture (soa ra) — part 1: Terminology and concepts for soa. International Organization for Standardization , 51URL: https://www.iso.org/standard/63104.html.

[14] Kaisler, S., Armour, F., Espinosa, J.A., Money, W., 2013. Big data: Issues and challenges moving forward, in: 2013 46th Hawaii International Conference on System Sciences, IEEE. pp. 995–1004. doi:10.1109/hicss.2013.645.

[15] Kiran, M., Murphy, P., Monga, I., Dugan, J., Baveja, S.S., 2015. Lambda architecture for cost-effective batch and speed big data processing, in: 2015 IEEE International Conference on Big Data (Big Data), IEEE. pp. 2785–2792.

[16] Kreps, J., 2014. Questioning the lambda architecture. Online article, July 205. URL: https://www.oreilly.com/radar/questioning-the-lambda-architecture/.

[17] Marr, B., 2016. Big data in practice: how 45 successful companies used big data analytics to deliver extraordinary results. John Wiley and Sons. doi:10.1109/bigdata.2018.8622333.

[18] Newman, S., 2015. Building Microservices: Designing Fine-Grained Systems. O'Reilly Media, Inc.

[19] OATH, 2007. Oath reference architecture, release 2.0 initiative for open authentication. OATH URL: https://openauthentication.org/wp-content/uploads/2015/09/ReferenceArchitectur

[20] Pääkkönen, P., Pakkala, D., 2015. Reference architecture and classification of technologies, products and services for big data systems. Big data research 2, 166–186.

[21] Quintero, D., Lee, F.N., et al., 2019. IBM reference architecture for high performance data and AI in healthcare and life sciences. IBM Redbooks.

[22] Rad, B.B., Ataei, P., 2017. The big data ecosystem and its environs. International Journal of Computer Science and Network Security (IJCSNS) 17, 38.

[23] Rada, B.B., Ataeib, P., Khakbizc, Y., Akbarzadehd, N., 2017. The hype of emerging technologies: Big data as a service. Int. J. Control Theory Appl 9, 1–18.

[24] Richards, M., 2015. Microservices vs. Service-Oriented Architecture. O'Reilly Media, Inc.

[25] Sagiroglu, S., Sinanc, D., 2013. Big data: A review, in: 2013 International Conference on Collaboration Technologies and Systems (CTS), IEEE. pp. 42–47. doi:10.1109/cts.2013.6567202.

[26] Srivastava, R., 2018. Big data: Issues and challenges. International Journal Of Scientific And Innovative Research .

[27] Viana, P., Sato, L., 2014. A proposal for a reference architecture for long-term archiving, preservation, and retrieval of big data, in: 2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications, IEEE. pp. 622–629.

[28] Wang, C.J., Ng, C.Y., Brook, R.H., 2020. Response to covid-19 in taiwan: big data analytics, new technology, and proactive testing. Jama 323, 1341–1342.