CoK: A Survey of Privacy Challenges in Relation to Data Meshes

Nikolai J. Podlesny(✉), Anne V. D. M. Kayem, and Christoph Meinel

Hasso Plattner Institute, University of Potsdam, Potsdam, Germany
{Nikolai.Podlesny,Anne.Kayem,Christoph.Meinel}@hpi.de

Abstract. The growing volumes of data that appear on multiple distributed platforms raise the question of how to compose data meshes that can be published and/or shared safely amongst multiple cooperating parties. Data meshes are composed of subsets (or whole sets) of data repositories that are owned by autonomous parties. This raises new challenges in terms of guaranteeing privacy across various data mesh compositions. In this paper, we present a survey of the issues that emerge in guaranteeing the privacy of distributed mesh data. We discuss the limitations of existing solutions in handling personal data privacy with respect to meshed data. Finally, we postulate that identifying personal data in such datasets must be handled with a performance efficient algorithm that can determine (on-the-fly), potential linkages across various data repositories, that could be exploited to subvert privacy.

1 Introduction

Dealing with mesh data from the privacy perspective is important in the IT industry. In fact, data meshes are in reality, a special case of distributed data repositories where the data exist in a flexible ecosystem but with clear user-ownership properties. Unlike standard relational database management systems, a central authority is absent and is instead replaced by separate authorities that co-exist in a "mutually exclusive and collectively exhaustive" environment. That is, data mesh instances can interact with each other and share data across different domains. For instance, an online marketing platform shares data with banking platforms and shopping regulatory services to validate a purchase request from a given customer. In essence, the goal is that there should be no centralised communication orchestrator required under this paradigm to guarantee data privacy across the different domains. While each database instance allows flexibility nuances, they adhere to overarching architecture principles and guarantees service level agreements to each other through data contracts (illustrated in Fig. 1). This paradigm of data meshes can be referred to as micro-service architecture in software engineering, where each service is encapsulated and isolated to allow more flexibility.

Problem Statement. Distributing private information and fragmenting their identifiers significantly impede their tracing and discovery. This may sound good in the first moment, but it exacerbates privacy work to protect the same. To

C. Strauss et al. (Eds.): DEXA 2022, LNCS 13426, pp. 85–102, 2022.
https://doi.org/10.1007/978-3-031-12423-5_7

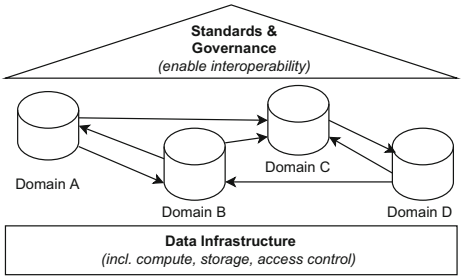


Fig. 1. Illustration of data meshes within an organisation

adhere to a high ethical standard and be compliant with most legislation like GDPR, HIPAA or CCPA, personally identifiable information (PII) even being distributed, must be protected, deleted upon request, and held secure. To do this, their existence and location must be known, even in a fragmented environment. Despite that individual data points might not initially be considered a privacy risk, their combination can be. Such attribute combinations are known as quasi-identifiers (QID). Traditional use cases of QID discovery imply static datasets with a standard relational database model, where standard metrics have to be addressed. In (data) mesh environments, there might be no, or only dynamically changing relational models. With the absence of any centralised layer that can identify, classify, label and alienate PII data records, differential privacy mechanisms by nature cannot help and a different solution is needed.

Contribution. In this work, we review, discuss and analyse the privacy implication of data mesh environments. We consolidate and systematise the state-of-the-art of related privacy work to do so. Based on this systematisation of knowledge (SoK), this work derives privacy fallacies in data mesh settings. Further, it discusses why practically the right of deletion, and other privacy actions are difficult to realise. We then offer experiments on implications for the search of privacy-compromising quasi-identifiers as vanishing points for de-anonymisation activities through comparing data mesh vs traditional RDBMS setups.

Outline. The rest of the paper is structured in the following manner: We assemble, consolidate and systematise latest related work in Sect. 2. This includes research on syntactic data anonymisation in Subsect. 2.3, semantic data anonymisation and differential privacy in Subsect. 2.4, unique column combinations in Subsect. 2.5, high-dimensional data anonymisation in Subsect. 2.6, quasi-identifier discovery in Subsect. 2.7, as well as data mesh databases in Subsect. 2.1 and privacy in data mesh environments in Subsect. 2.2. Section 3 then offers a characterisation of data meshes and quasi-identifiers in their context. Section 4 contributes experiments on discovering quasi-identifiers to avoid private data exposure in data mesh environments. Section 5 finally concludes our results and suggests avenues for future work.

2 State-of-the-Art

Data mesh databases are not a completely new research field, and have been addressed partially in the fields of peer-to-peer databases, distributed databases, data mesh topologies, syntactic-, semantic data anonymisation, high-dimensional data anonymisation and quasi-identifier discovery. The following subsections will summarise the most recent and extraordinary related work.

2.1 Data Mesh Databases

Back in 1997, Beall et al. reviewed systems for a general-purpose mesh database based on a hierarchy of topological entities [7]. Their hierarchical analysis for topology concluded that the hierarchic representation does not add a significant amount of extra storage to a mesh database. Rather, this representation can easily be extended to represent non-manifold models properly. In 2001, Gribble et al. published work on peer-to-peer systems and their behaviour towards the semantics of data [30]. Further, Gribble et al. highlight that P2P databases have unique challenges like the data placement problem where it is necessary to figure out how to distribute data and work so database queries can run at a low cost under resource and bandwidth constraints. As an outlook, new architectural designs are mentioned promising to help P2P databases to implement distributed query answering systems that are more scalable, reliable, and performant.

On a different venue, the mappings between peer-to-peer (P2P) databases are typically described to be local with no global schema accordingly to Bernstein et al. [57]. Also, the configurations and mappings between peers are highly dynamic that require semi-automatic solutions. In their work, Bernstein et al. presents Local Relational Model (LRM) as an architecture that can help resolve these issues for modern P2P databases. Franconi et al. [26] proposed a new model for P2P databases where nodes can request data from another node and use the third node for evaluation, but there can be no complex queries across the entire network. In contrast to standard first-order semantics, Franconi et al.'s new model captures the intended semantics of P2P systems. The model also halts the propagation of inconsistencies from node to node, so the database remains consistent, even if some of the nodes have inconsistent data. Remacle et al. [64] offered work on an Algorithm Oriented Mesh Database (AOMD) to manage mesh databases. Due to storage and algorithmic complexity, it is not possible to maintain complete graphs of data meshes according to Remacle et al. [64]. AOMD uses dynamic mesh representation to decrease computer memory use and increase algorithmic efficiency. It results in a light and efficient software implementation for mesh databases. Eunyoung Seegyoung Seol presented in his PhD thesis a mesh that is a piece-wise decomposition of the space/time domain where used by numerical simulation procedures [68]. Flexible distributed mesh database (FMDB) capable of shaping its representation based on the application's specific needs. FMDB embedded in SCOREC simulation packages effectively supporting automated adaptive analyses. Further, Seol et al. [67] published work on flexible distributed Mesh database (FMDB), that is a partition model

and a distributed mesh management system. Seol et al. model has been used to efficiently support parallel automated adaptive analysis processes. The integration of mesh technology with the unified theory of acceptance and use of technology (UTAUT) can help businesses with analytics and technology adoption accordingly to Shirazi et al. [69]. Customised UTAUT models for mesh app, service, and conversational systems adoption that add motivation, innovation, privacy, and AI problem solving to traditional UTAUT can lead to intelligent mesh technology [69]. Rodríguez-Gianolli et al. [65] presented a hyperion prototype that demonstrates the possibility of using Peer-to-Peer (P2P) computing to share data. In their prototype, each peer includes a database with its own schema. The peers can join and leave the network independently. In a Hyperion P2P Database Network, the peer nodes share data by clustering into interest groups and pairing up using acquaintance links. The P2P Layer handles the peer-to-peer data sharing, while the Local Database Layer handles traditional database functions [65].

A P2P database system (PDBS) is a collection of autonomous databases that communicate with each other in a peer-to-peer fashion. Bonifati et al. [63] elaborated on how PDBS can borrow ideas from distributed database systems (DDBS) and multi-database systems (MDBS). For that purpose, Bonifati et al. compared past distributed database systems to PDBS, emphasising the database-centric and P2P-centric features of PDBS [63]. On the same note, Masud et al. investigated transaction processing in a peer-to-peer database network [47]. Their work looked into the problems around the consistent execution of concurrent transactions. Masud et al. also proposed solutions like Merged Transactions and OTM-based propagation to guarantee consistent performance [47].

Various venues broach the issue of data mesh environments, their technical realisation and implication towards distributed datasets. Yet, the fragmentation of data records into distributed databases and the consequences to overarching, traditional central tasks like security and privacy themes remains mostly unresolved.

2.2 Privacy in Mesh Networking

A few privacy questions have been discussed in the context of mesh networks and mesh structures. Wu et al. illustrated privacy attacks on mesh network based on the open medium property of wireless channel [77]. Traditional anonymous routing algorithm cannot be directly applied to Mesh network. In their paper, Wu et al. designed a private routing algorithm that used “Onion”, i.e., layered encryption, to hide routing information [77]. Ganesh et al. proposed a strategy that applies self-organising maps (SOM) algorithm separately in each distributed dataset relative to database horizontal partitions [28]. In the sequence, these representative subsets are sent to a central site, which performs a fusion of partial results and applies K-means algorithms.

While research has been done on privacy in mesh networks, their findings and concepts are not easily transferable to data mesh environments. Data mesh is a special case of databases, while mesh networks originate from network topologies.

A similar paradigm but different application context. As open problems remain the question of how to find distributed describing attributes forming personally identifiable information (PII), how data deletion or data lineage can be realised in fragmented landscapes.

2.3 Syntactic Data Anonymisation

Randomisation [33,45], generalisation [27,72], suppression [27,72], and perturbation [45] are among the data transformation methods used in syntactic data anonymisation. Generalisation restructures the content of a dataset by changing its values according to a pre-defined term replacement taxonomy, whereas suppression simply erases data. As one travels up the ladder in a hierarchy-based taxonomy, each value gradually loses its uniqueness.

The k-anonymity Family. One of the first and best-known is k-anonymity, limiting distinguishability by classifying each tuple in the data set with at least $k-1$ identical data records. Sweeney claims that the $k-1$ closest neighbors are chosen based on similar descriptive features and enforced via generalisation and suppression [72]. The pattern of generalisation is to aggregate data values through a pre-defined hierarchy, such as combining the individual year 2021 into a year range of 2020–2025. Suppression, on the other hand, fully removes the selected data value. The generalisation toolset appears to be sensitive to attacks based on homogeneity and background knowledge [46]. To mitigate this, l-diversity takes the granularity of sensitive data representations into account, ensuring a factor of l diversity for each quasi-identifier within a particular equivalence class (usually a size of k). By evaluating the relative distributions of sensitive values in specific equivalence classes and throughout the entire dataset, t-closeness as an extension handles skewness and background knowledge attacks [42]. k-anonymity is also a privacy metric denoted k-map. If every combination of attribute values for quasi-identifiers appears at least k times in a dataset, it meets the k-map constraint [72]. To protect against symmetric assaults, Nergiz et al. [54] presented δ-presence, which builds on both k-anonymity and k-map. δ-min and δ-max are hidden in the δ-parameter. These two characteristics deal with the fact that no one is present.

Data Transformation Techniques. To support data transformation, the prior anonymisation techniques and their modifications used generalisation and suppression [27,49]. This is useful for theoretical demonstrations, but it quickly reaches its limits when dealing with larger datasets. Syntactic data anonymisation methods like k-anonymity [72], l-diversity [46], and t-closeness [42] are NP-hard, as Meyerson et al. [49] and Bayardo et al. [29] have shown. Because of their iterative and incremental character, the dependent *generalisation* methods are NP-hard in and of themselves. Applying generalisation and suppression to high dimensional data results in considerable information loss, rendering the data worthless for data analytics, according to Aggrawal et al. [4]. This is especially true because generalisation's runtime grows exponentially for several descriptive attributes, making it unfeasible. As a result, suppression persists and obliterates

attribute values, resulting in significant information loss. Given the algorithm's complexity, all variations can only employ heuristics like k-optimise [6] to get improved approximations to perfect privacy, not perfect privacy [5].

Perturbation has been proposed as a viable alternative to generalisation [45]. The alternation of the real value to the nearest similar findable value is referred to as perturbation. This includes the effect of introducing an aggregated value or employing a close-by value so that just one value needs to be modified rather than numerous ones to form clusters. Finding such a value can take longer in certain cases due to iteratively rechecking the newly produced value(s), which negatively influences performance.

Optimal k-anonymity has been demonstrated to be an NP-hard task [5,49]. Due to their algorithmic nature, applying generalisation and suppression strategies to high-dimensional data results in a substantial level of information loss, leaving the data essentially unusable for data analytics. Tassa et al. [74] recommend using k-concealment to reduce the information loss caused by generalising database entries. However, in the case of high-dimensional application fields, both contributions degrade the NP-hardness. Fredj et al. [27] provided an in-depth review, categorisation, and advice for selecting generalisation algorithms.

The problem of ensuring k-anonymity with either optimal or holistic techniques to syntactic data anonymisation has been demonstrated to be an NP-hard task [49]. Heuristics can only be used to achieve better approximations to perfect privacy, not perfect privacy, in all types of k-anonymity algorithms [5]. As a result, scaling, particularly generalisation and perturbation in high-dimensional data, produces an impractical runtime [58,62] and a large level of information loss, rendering the data worthless for data analytics. With the help of GPU acceleration [61], it has been proven to shift the time complexity amplitude as runtime explosion from smaller $n < 20$ to larger $n < 150$ for 2^n, yet the nature of the growth remains.

2.4 Semantic Data Anonymisation and Differential Privacy

Semantic data anonymisation approaches sum up the statistical distributions of data values and the semantic meanings drawn from linking (defining patterns) between data points in an attempt to re-define privacy not just as a process of syntactically transforming datasets but also to consider both the statistical distributions of data values and the semantic meanings drawn from linking (defining patterns) between data points. The data veracity is tampered with by deleting significant ties between the data and an individual. Noise injection, permutation, or statistical shifting are commonly used to achieve this [19,33,44]. These algorithms are also known as *differential privacy*, and their statistical approaches are highly optimised for pre-defined use cases and mass data processing. In differential privacy, for example, this is accomplished by deciding how many noise injections to add to the output dataset at query runtime to assure anonymity in each situation [18]. Further, Dwork et al. extend their work with a vast introduction into the algorithmic foundations of differential privacy [22].

Individual contributions to differential privacy include the use of the exponential mechanism to expose statistical information about a dataset while concealing the private specifics of individual data items [48]. By applying controlled random distribution sensitive noise additions, the Laplace method for perturbation facilitates statistical shifting in differential privacy [20,38]. Because both sensitive attributes and quasi-identifiers are evaluated on a per-row basis during anonymisation [41], the discretised version [44] is known as a matrix mechanism. Because these anonymisation are done at runtime and on a case-by-case basis, the anonymisation processing is deferred until query runtime, increasing the risk of data leakage [36]. Leoni introduced "non-interactive" differential privacy [40] by performing statistical adjustments a priori to user searches. Another difficulty with differential privacy is that it is computationally infeasible to apply differential privacy to huge datasets (impractical). Dwork et al. shows that differential privacy is likewise NP-hard [21]. Experts are still debating whether approximation differential privacy algorithms provide adequate privacy assurances. An arbitrary family of attribute sets could be used to link a single data record back to its owner in certain conditions [24]. Abadi et al. offered the application of incorporating differential privacy into the deep learning context [1]. Even the US Census Bureau plans to adopt differential privacy accordingly to John Abowd [3]. But as Lee et al. have highlighted, the concept of differential privacy received considerable attention in the literature, yet little discussion is available on how to apply it in practice [39].

These revelations lead to an unsolved issue. Due to their complexity, anonymising a large dataset using either approximate procedures that may leave data inferences that can be exploited to de-anonymize people or precise counterparts results in exponentially growing runtime.

Randomisation techniques have gained increased attention as a result of the issues surrounding syntactic data anonymisation [33,45]. This semantic data anonymisation technique aims to re-define privacy as a process of considering both statistical distributions of data values and semantic meanings extracted from linking (defining patterns) between data points, rather than simply as a process of syntactically altering datasets. Dwork et al. [23] provide an in-depth survey of past work, in addition to the previous description of relatively recent contributions. Dankbar et al. have provided a comprehensive overview of the current literature on unequal privacy. They also pointed out some important general constraints, such as the theoretical character of the privacy parameter, which limits the ability to quantify the level of anonymity that would be guaranteed to patients [14]. Ji et al. explored the relationship between machine learning and differential privacy [34]. To illustrate both its strong guarantees and limitations, Li et al. focus on empirical accuracy performances of algorithms and semantic implications of differential privacy [43].

Semantic data anonymisation methods, such as differential privacy, have been demonstrated to be NP-hard for big datasets [21]. Given their runtime and use case-specific nature, they are computationally infeasible (impractical performance-wise) when applied to large high-dimensional data.

2.5 Unique Column Combinations

Unique column combinations (UCC) are attribute combinations that generate a unique identifier for the given dataset in data profiling (table). Discovering these unique column combinations (UCC) is a major scientific challenge.

Abedjan et al. [2] compiled and formalised the most recent breakthroughs in the finding of UCCs in their paper. Heise et al. built on their work by presenting a scalable discovery of unique column combinations based on parallelisation and the scale-out concept [32]. Feldmann has done the same thing [25]. Han et al. build on similar ideas [31] and use Hadoop with its MapReduce technology [15] to create a distributed computing environment. Papenbrock et al. [56] offered a comparison of alternative discovery strategies. Papenbrock et al., on the other hand, proposed a hybrid of quick approximation approaches and efficient validation procedures for UCCs [56]. Ruiz et al. published a patent recently that summarised several dataset profiling tools, techniques, and systems, including efficient UCC finding [66].

The search for UCC may be encapsulated in a cyclical dependence on the Hitting-Set issue as a family of W[2]-complete problems [9,17], according to Bläsius et al. [9]. In the worst-case scenario, this implies a super polynomial runtime, rendering its use to huge, high-dimensional data impracticable for the time being.

2.6 High-Dimensional Data

Given past advances in syntactic and semantic data anonymisation, more attention has shifted to hybrid systems that incorporate aspects from the initial syntactic and semantic data anonymisation approaches and provide abstractions from the raw dataset via aggregations or separations. For example, in attribute compartmentation [58,62], privacy is ensured by separating attributes that constitute quasi-identifiers using the notion of maximum partial unique column combinations (mpUCC) from the data profiling domain (mpUCCs). Quasi-identifiers are attribute value combinations that uniquely identify persons in a dataset (QID). By removing those QIDs, the re-identification attack of mixing QIDs with auxiliary data to draw inferences and extract private information is also prevented [76]. However, finding quasi-identifiers is difficult.

The enormous number of rows and columns distinguishes high-dimensional data. While the growing number of rows is seldom a problem, the growing number of columns can fast cause state-space explosions in enumeration issues [8]. The higher the dataset dimensions, the faster it reaches computational infeasibility. As can be seen from the preceding subsections, several disciplinary approaches for obtaining privacy, such as data profiling and mining, anonymisation processing, and differential privacy, eventually run into NP-hard difficulties.

In a few cases, high-dimensional data is being anonymized in great detail. Kohlmayer et al. proposed adaptations based on the Secure Multi-party Computing (SMC) protocol as a flexible approach on top of k-anonymity, l-diversity, and

t-closeness, as well as heuristic optimisation, to anonymize distributed and separated data silos in the medical field [37]. Mohammed et al. propose *LKC-privacy* to achieve privacy in both centralised and distributed scenarios [50], promising scalability for anonymising large datasets *LKC-privacy*, however, restricts the length of quasi-identifier tuples to a pre-determined number of characters that offers a practical approach but does not guarantee the entire absence of privacy-violating identifiers in high-dimensions. Other initiatives, such as Zhang et al. [80], employ a MapReduce approach based on the Hadoop distributed file system (HDFS) to increase compute capacity. On the other hand, the NP-hard nature swiftly beats the economic scalability options. Large numbers of entities defining characteristics (hundreds of attributes) must be handled in a performance-efficient and privacy-preserving way.

There are two reasons why discriminating between sensitive and non-sensitive properties is problematic, according to Manolis Terrovitis' study [75]. First, we can see that sensitive features are not the main reason for the success of de-anonymisation assaults (homogeneity, similarity, and background information). Second, creating an exhaustive collection of sensitive and non-sensitive qualities is problematic for high-dimensional datasets with distinct patterns that expand with the amount of data acquired on an individual. Podlesny et al. proposed modeling the attribute linkage problem for generating privacy-preserving data silos as a Bayesian network [59,60] to reduce the complexity of the compartmentation problem [58,62]. To train a Bayesian network, exact inference learning [53] and approximate inference learning [13] have the same NP-hardness. Recent contributions, however, show that using attribute linkage techniques to compress the network enables for performance-scalable data processing even on huge datasets [60]. Clifton et al. provided a balanced review of outstanding concerns in both syntactic and semantic data anonymisation methods, as well as its benefits, belongings, and summarised critiques [12]. Clifton et al. point out that the differences between different syntactic and semantic anonymisation origin models are less pronounced than previously supposed. Both archetypes, however, will have problems in large-scale data settings. Differential privacy is frequently the best empirical privacy for a fixed (empirical) utility level, however syntactic anonymity models may be preferred for more precise answers.

Regardless of where it came from, data anonymisation is yet to be applied to large-scale, multi-attribute, high-dimensional datasets in a reasonable amount of time and with limited resources. Each solution suffers from considerable complexity restrictions for huge quantities of descriptive characteristics (columns), resulting in massive information loss, calculation demands, and hence runtime, or privacy guarantees through approximation approaches.

2.7 Quasi-Identifier Discovery

Byun et al. addressed the lack of diversity through equivalence classes and their information-loss by transforming the k-anonymity problem to a k-member clustering problem [11], based on Sweeneys work on the family of k-anonymity techniques [72,73]. While Byun et al. technique uses distance and cost functions

works for numeric and categorical data, it does not guarantee approximation factors. For clustering purposes, the projection of quasi-identifier similarity remains data-specific.

Xiao et al. published anatomy, a novel approach that immediately releases all quasi-identifiers and sensitive values in two independent tables [78]. This, in conjunction with grouping operations, should allow for the capture of correlation while minimising reconstruction error. Zhang et al. investigated the scalability benefits of horizontal scaling in cloud computing environments, as well as the use of a quasi-identifier index-based technique to speed up data querying on huge datasets [79]. Statistical de-anonymisation attacks on high-dimensional datasets were proven by Narayanan et al. for re-identifying people in the Netflix Prize dataset with tolerance for certain inaccuracies in the adversary's prior information [51]. Soria-Comas et al. summarised the topic of re-linkage using quasi-identifiers. They explored data governance issues like user permission, purpose limitation, transparency, individual rights of access, correction, and deletion. When deleting specified qualities against extra personally identifiable information (PII), Narayanan et al. expounded on the PII fallacy of the HIPAA privacy law [52], as the eradication of all quasi-identifiers is not assured. Soria-Comas et al. work also highlighted the need for new privacy models built from the ground up with big data requirements in mind, such as continuous and vast data collected from numerous source systems, resulting in multi-attribute and high-dimensional datasets [70]. Braghin et al. have submitted an optimised quasi-identifier strategy that uses parallelisation for efficient QID discovery [10], even though parallelisation is not a novel concept. Braghin et al. study can serve as a comparative baseline for our research due to its extensive description and encouraging outcomes.

The discovery of quasi-identifiers summarised as *Find-QID* problem [61] remains NP-hard and W[2]-complete [9,61]. Heuristic and greedy approach exist, they even weaken the exponential implication of the same *Find-QID* problem, yet particularly in high-dimensional spaces a lasting solution remains open unless the W-hierarchy collapses [9]. This assumes an already pre-compiled, static dataset. Adding now a distributed factor in, like in the case of data meshes, the search and identification of QIDs become even more complex.

In summary, the community has done a lot of research on peer-to-peer database, mesh network and anonymisation techniques individually. Yet, to the best of our understanding, the paradigm of data mesh in databases and its side effects with, against and towards privacy is largely unexplored. In particular, this includes the topics around data deletion, quasi-identifier discover and data lineage under the constraint of distributed, highly fragmented data records across multiple data mesh instances. To emphasis the underlying complexity, we demonstrate the differences of data mesh to more traditional database approaches in the experiments of the following Sect. 4.

3 Data Meshes

To recapitulate on essential terminologies, we briefly summarise the current understanding and state of data mesh in database and quasi-identifiers in the same domain. The concept of data mesh centers around the democratisation and decentralisation of development activities. Instead of a central and predominating database with strict governance, a distributed setup build the basis of data meshes. Each data repository is somehow coupled, can have upstream and downstream dependencies guaranteed through data contracts defining their usage, availability, quality and content. This structural paradigm offers flexibility in its configuration. Still, the same gained flexibility introduces looser governance challenges like the absence of data lineage, which we will describe in the following more profoundly. A similarity can be found in software engineering, where a trend from monolith- towards microservices as architecture patterns has been observed [35,55].

Characteristics of a Data Mesh. Given the decoupled nature of data meshes [16], different data records might be split or even duplicated across multiple data repositories. Traditionally, each data mesh instance is dedicated to a certain data domain, with a clear owned business entity and corresponding dependencies, inputs and outputs objective. While each data mesh instance is somehow autarkic, it may directly consume each other. Figure 1 illustrates this setup on a high level perspective. Data between each instance can be linked through identifiers, but this is not guaranteed. Such a fractured landscape brings value through its flexibility. Each data domain can act and scale independently, yet learnings from different sectors include that the same paradigm re-balances the weight against arbitrary governance structures. As seen in the healthcare domain, the archetype of various detached data repositories introduces a challenge for overarching topics like data privacy, common interfaces and standardisation.

In the case of a central place, the same overarching objectives can be easily monitored, traced and supported like in the case of the implementation of GDPRs *data deletion* right. A simple act like deleting personally identifiable information (PII) sounds trivial, but imagine there are hundreds of data mesh instances across hundreds of teams and each acts on its own. In various decoupled data repositories, tracking down distributed user attributes can only work with thoroughly conducting *data lineage* which requires a lot of dedication and documentation work for each development team as cross-linkages may be possible. Figure 2 depicts such perspective, where each domain holds a subset of user data. Each subset individually may not look concerning from a privacy perspective, but joining these through existing identifiers they can become concerning.

Quasi-identifiers in a Data Mesh. Quasi-identifiers (QID) are attribute combinations that jointly form identifiers while independently might seem unsuspicious. A quasi-identifier does not have to identify all individuals, but serves at least one individual to be exposed and cause harm to their privacy. Formally, QIDs are defined as

Definition 1. *Quasi-identifier*
Let $F = \{f_1, ..., f_n\}$ *be a set of all features and* $B := \mathcal{P}(F) = \{B_1, ..., B_k\}$ *its power set, i.e. the set of all possible feature combinations. A set of selected features* $B_i \in B$ *is called a quasi-identifier, if* B_i *identifies at least one entity uniquely and all features* $f_j \in B_i$ *are not standalone identifiers.*

To make this tangible, the readers attention is pointed towards Fig. 2 one more time. Here, one can see that Domain A holds a *ZIP code* information, Domain B *age* and *gender* and both are linked through the *Call Center ID*. Further, Domain D holds analytical results like the *disease prediction* or *medical adherence*. When following all identifiers, one can easily build a data profile including *age*, *gender*, *ZIP code*, *disease prediction* and *medical adherence* without touching the Domain C. Now, as Sweeney et al. showed that 87% of the entire population are identifiable through the combination of age, gender and zip [71], an attacker may infer *disease prediction* and *medical adherence* to those 87%.

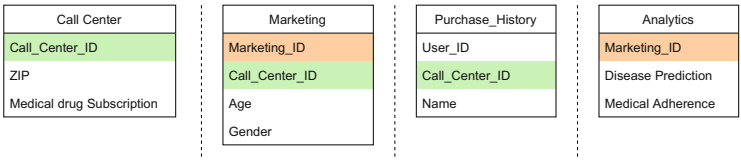


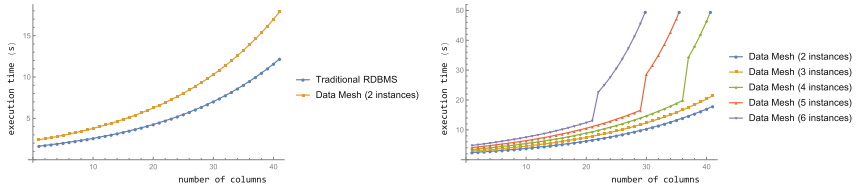
Fig. 2. Indirect linkage of quasi-identifiers in a data mesh

4 Experiments

To fortify the novolum that the data mesh paradigm creates towards data privacy topics, we will build on the prior knowledge and characteristic summary and outline through a series of experiments the same theses and raised challenges. For that purpose, we leverage a semi-synthetic dataset and state-of-the-art hardware to compare different database archetypes and their runtime implications on finding PII compromising quasi-identifiers.

Hardware. Our examination runs on a GPU-accelerated high-performance compute cluster, housing 64 vCPU cores (E5-4650), 240 GB RAM, and 8x NVIDIA GeForce 3060 with 3584 CUDA cores each and a combined Tensor performance of 816 Tensor TFLOPs. GPU-related experiments' execution environment will be restricted to one dedicated CPU core and a single, dedicated Tesla V100 GPU.

Dataset. For the purpose of evaluation, a semi-synthetic health dataset has been compiled based on publicly available contributions, previous work and publications. The dataset consists of genomic data, fake but consistent names, addresses, SSN, passwords and telephone numbers, as well as medical records randomly



(a) .. in traditional RDBMS to a data mesh (b) in different data mesh settings

Fig. 3. Projected runtime growth of discovered QID over increasing columns.

assigned but adhering to known statistical distributions. For transparency, the full dataset can be downloaded from github.com[1].

Evaluation. To demonstrate the differences in time complexity when different database archetypes are being introduced, these experiments build on Sweeney's k-anonymity approach of finding quasi-identifiers [73]. A GPU-accelerated search schema without heuristics purely based on *groupby* and *count* statements developed by Podlesny et al. [61] is being utilised in the following. Figure 3a delineates the runtime growth for discovering the quasi-identifiers. The Y-axis represents the execution time to find all QIDs in an exact manner (not heuristic) while the x-axis the increasing number of describing attributes being stored in the associated database archetype. The different database archetypes of traditional central RDBMS and data mesh are clearly visible. Both runtime portray an exponential increase, while the growth of the data mesh answers to a higher factor (see Fig. 3a). While both, a traditional central RDBMS and a data mesh can be scaled horizontally and vertically in number of nodes and hardware used, the data mesh suffers a fragmentation of describing data attributes that can form quasi-identifiers. This fragmentation needs to be first compensated which essentially answers to more network I/O and therefore longer processing time. The larger the fragmentation, the higher the network I/O and the longer the compute.

Following the same line of thoughts, Fig. 3b depicts the evolution of the same metrics over different data mesh sizes. The data mesh size answers to the number of instances involved with equally distributed data attributes, starting from two and increasing. Given the nature of the search, the complexity is exponential already. Yet, two things stand out. First, the more data mesh instances exist with equivalent data distribution, the sooner runtime increases due to the higher degree of fragmentation and therefore, more data shifting and joining is required. Second, the more data meshes exist, the earlier one experiences an uncontrolled explosion of execution time as, given the hardware constraint, the capacities of main memory and GPU memory are exceeded.

[1] https://github.com/jaSunny/synthetic_genome_data.

5 Conclusion and Future Directions

The previous sections offered a systematisation of knowledge and clarified characteristics of data mesh and how quasi-identifiers potentially exposing PII. Further, the summarised state-of-the-art delineates gaps for privacy and anonymisation concepts in distributed data mesh environments. To demonstrate the uniqueness and scalability of this problem, we have offered a variety of experiments to discover quasi-identifier exposing PII in a traditional RDBMS setup and compared these metrics against same algorithms running in a data mesh setup. The increase of complexity and runtime is clearly visible.

Based on this understanding, we formulate the open distributed Quasi-identifiers problem: To find usage of PII data within a data mesh, elements of one quasi-identifiers (QIDs) might be distributed and linked across more than one database instance. To find these distributed QIDs, all describing attribute combination of any length that can be cross-linked through arbitrary identifiers need to be considered. Due to its distributed nature, this represents a special case of the W[2]-complete *Find-QID* problem [61].

References

1. Abadi, M., et al.: Deep learning with differential privacy. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 308–318. ACM (2016)
2. Abedjan, Z., Naumann, F.: Advancing the discovery of unique column combinations. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, pp. 1565–1570 (2011)
3. Abowd, J.M.: The US census bureau adopts differential privacy. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, p. 2867 (2018)
4. Aggarwal, C.C.: On k-anonymity and the curse of dimensionality. In: Proceedings of the 31st International Conference on Very Large Data Bases, VLDB 2005, VLDB Endowment, pp. 901–909 (2005)
5. Barth-Jones, D.: The 're-identification' of governor William Weld's medical information: a critical re-examination of health data identification risks and privacy protections, then and now (July 2012)
6. Bayardo, R.J., Agrawal, R.: Data privacy through optimal k-anonymization. In: 2005 Proceedings of the 21st International Conference on Data Engineering, ICDE 2005, pp. 217–228. IEEE (2005)
7. Beall, M.W., Shephard, M.S.: A general topology-based mesh data structure. Int. J. Numer. Meth. Eng. **40**(9), 1573–1596 (1997)
8. Birnick, J., Bläsius, T., Friedrich, T., Naumann, F., Papenbrock, T., Schirneck, M.: Hitting set enumeration with partial information for unique column combination discovery. Proc. VLDB Endow. **13**(12), 2270–2283 (2020)
9. Bläsius, T., Friedrich, T., Schirneck, M.: The parameterized complexity of dependency detection in relational databases. In: 11th International Symposium on Parameterized and Exact Computation, Dagstuhl, Germany, vol. 63, pp. 6:1–6:13 (2017)

10. Braghin, S., Gkoulalas-Divanis, A., Wurst, M.: Detecting quasi-identifiers in datasets (16 January 2018). US Patent 9,870,381
11. Byun, J.-W., Kamra, A., Bertino, E., Li, N.: Efficient k-anonymization using clustering techniques. In: Kotagiri, R., Krishna, P.R., Mohania, M., Nantajeewarawat, E. (eds.) DASFAA 2007. LNCS, vol. 4443, pp. 188–200. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-71703-4_18
12. Clifton, C., Tassa, T.: On syntactic anonymity and differential privacy. In: 2013 IEEE 29th International Conference on Data Engineering Workshops (ICDEW), pp. 88–93. IEEE (2013)
13. Dagum, P., Luby, M.: Approximating probabilistic inference in Bayesian belief networks is NP-hard. Artif. Intell. **60**(1), 141–153 (1993)
14. Dankar, F.K., El Emam, K.: Practicing differential privacy in health care: a review. Trans. Data Priv. **6**(1), 35–67 (2013)
15. Dean, J., Ghemawat, S.: MapReduce: simplified data processing on large clusters. Commun. ACM **51**(1), 107–113 (2008)
16. Dehghani, Z.: Data mesh principles and logical architecture. martinfowler.com (2020)
17. Downey, R.G., Fellows, M.R.: Fundamentals of Parameterized Complexity. TCS, vol. 4. Springer, London (2013). https://doi.org/10.1007/978-1-4471-5559-1
18. Dwork, C.: Differential privacy: a survey of results. In: Agrawal, M., Du, D., Duan, Z., Li, A. (eds.) TAMC 2008. LNCS, vol. 4978, pp. 1–19. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-79228-4_1
19. Dwork, C.: Differential privacy. In: van Tilborg, H.C.A., Jajodia, S. (eds.) Encyclopedia of Cryptography and Security, pp. 338–340. Springer, Boston (2011). https://doi.org/10.1007/978-1-4419-5906-5_752
20. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 265–284. Springer, Heidelberg (2006). https://doi.org/10.1007/11681878_14
21. Dwork, C., Naor, M., Reingold, O., Rothblum, G.N., Vadhan, S.: On the complexity of differentially private data release: efficient algorithms and hardness results. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, pp. 381–390. ACM, New York, NY, USA (2009)
22. Dwork, C., Roth, A.: The algorithmic foundations of differential privacy. Found. Trends® Theor. Compu. Sci. **9**(3–4), 211–407 (2013)
23. Dwork, C., Smith, A.: Differential privacy for statistics: what we know and what we want to learn. J. Priv. Confid. **1**(2), 135–154 (2010)
24. European Commission: Opinion 05/2014 on anonymisation techniques (April 2014)
25. Feldmann, B.: Distributed Unique Column Combinations Discovery. Hasso-Plattner-Institute, January 2020. https://hpi.de/fileadmin/user_upload/fachgebiete/friedrich/documents/Schirneck/Feldmann_masters_thesis.pdf
26. Franconi, E., Kuper, G., Lopatenko, A., Serafini, L.: A robust logical and computational characterisation of peer-to-peer database systems. In: Aberer, K., Koubarakis, M., Kalogeraki, V. (eds.) DBISP2P 2003. LNCS, vol. 2944, pp. 64–76. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24629-9_6
27. Fredj, F.B., Lammari, N., Comyn-Wattiau, I.: Abstracting anonymization techniques: a prerequisite for selecting a generalization algorithm. Procedia Comput. Sci. **60**, 206–215 (2015)
28. Ganesh, P., KamalRaj, R., Karthik, S.: Protection of privacy in distributed databases using clustering. Int. J. Mod. Eng. Res. **2**, 1955–1957 (2012)

29. Ghinita, G., Karras, P., Kalnis, P., Mamoulis, N.: Fast data anonymization with low information loss. In: Proceedings of the 33rd International Conference on Very Large Data Bases, VLDB Endowment, pp. 758–769 (2007)
30. Gribble, S.D., Halevy, A.Y., Ives, Z.G., Rodrig, M., Suciu, D.: What can database do for peer-to-peer? In: WebDB, vol. 1, pp. 31–36 (2001)
31. Han, S., Cai, X., Wang, C., Zhang, H., Wen, Y.: Discovery of unique column combinations with hadoop. In: Chen, L., Jia, Y., Sellis, T., Liu, G. (eds.) APWeb 2014. LNCS, vol. 8709, pp. 533–541. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11116-2_49
32. Heise, A., Quiané-Ruiz, J.A., Abedjan, Z., Jentzsch, A., Naumann, F.: Scalable discovery of unique column combinations. Proc. VLDB Endow. **7**(4), 301–312 (2013)
33. Islam, M.Z., Brankovic, L.: Privacy preserving data mining: a noise addition framework using a novel clustering technique. Knowl. Based Syst. **24**(8), 1214–1223 (2011)
34. Ji, Z., Lipton, Z.C., Elkan, C.: Differential privacy and machine learning: a survey and review (2014)
35. Kalske, M., Mäkitalo, N., Mikkonen, T.: Challenges when moving from Monolith to microservice architecture. In: Garrigós, I., Wimmer, M. (eds.) ICWE 2017. LNCS, vol. 10544, pp. 32–47. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-74433-9_3
36. Kifer, D., Machanavajjhala, A.: No free lunch in data privacy. In: Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data, SIGMOD 2011, pp. 193–204. ACM, New York (2011)
37. Kohlmayer, F., Prasser, F., Eckert, C., Kuhn, K.A.: A flexible approach to distributed data anonymization. J. Biomed. Inform. **50**, 62–76 (2014)
38. Koufogiannis, F., Han, S., Pappas, G.J.: Optimality of the Laplace mechanism in differential privacy. arXiv preprint arXiv:1504.00065 (2015)
39. Lee, J., Clifton, C.: How much is enough? Choosing ε for differential privacy. In: Lai, X., Zhou, J., Li, H. (eds.) ISC 2011. LNCS, vol. 7001, pp. 325–340. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-24861-0_22
40. Leoni, D.: Non-interactive differential privacy: a survey. In: Proceedings of the 1st International Workshop on Open Data, pp. 40–52. ACM (2012)
41. Li, C., Miklau, G., Hay, M., McGregor, A., Rastogi, V.: The matrix mechanism: optimizing linear counting queries under differential privacy. VLDB J. **24**(6), 757–781 (2015). https://doi.org/10.1007/s00778-015-0398-x
42. Li, N., Li, T., Venkatasubramanian, S.: t-closeness: privacy beyond k-anonymity and l-diversity. In: 2007 IEEE 23rd International Conference on Data Engineering, pp. 106–115 (April 2007)
43. Li, N., Lyu, M., Su, D., Yang, W.: Differential privacy: from theory to practice. Synth. Lect. Inf. Secur. Priv. Trust **8**(4), 1–138 (2016)
44. Liu, F.: Generalized gaussian mechanism for differential privacy. arXiv preprint arXiv:1602.06028 (2016)
45. Liu, K., Kargupta, H., Ryan, J.: Random projection-based multiplicative data perturbation for privacy preserving distributed data mining. IEEE Trans. Knowl. Data Eng. **18**(1), 92–106 (2006)
46. Machanavajjhala, A., Kifer, D., Gehrke, J., Venkitasubramaniam, M.: L-diversity: privacy beyond k-anonymity. ACM Trans. Knowl. Discov. Data (TKDD) **1**(1), 3 (2007)
47. Masud, M., Kiringa, I.: Transaction processing in a peer to peer database network. Data Knowl. Eng. **70**(4), 307–334 (2011)

48. McSherry, F., Talwar, K.: Mechanism design via differential privacy. In: 2007 48th Annual IEEE Symposium on Foundations of Computer Science, pp. 94–103. IEEE (2007)
49. Meyerson, A., Williams, R.: On the complexity of optimal k-anonymity. In: Proceedings of the 23rd ACM SIGMOD-SIGACT-SIGART Symposium, pp. 223–228. ACM (2004)
50. Mohammed, N., Fung, B., Hung, P.C., Lee, C.K.: Centralized and distributed anonymization for high-dimensional healthcare data. ACM Trans. Knowl. Discov. Data (TKDD) **4**(4), 18 (2010)
51. Narayanan, A., Shmatikov, V.: Robust de-anonymization of large sparse datasets. In: 2008 IEEE Symposium on Security and Privacy, SP 2008, pp. 111–125. IEEE (2008)
52. Narayanan, A., Shmatikov, V.: Myths and fallacies of "personally identifiable information". Commun. ACM **53**(6), 24–26 (2010)
53. Neapolitan, R.E.: Probabilistic reasoning in expert systems: theory and algorithms. CreateSpace Independent Publishing Platform (2012)
54. Nergiz, M.E., Atzori, M., Clifton, C.: Hiding the presence of individuals from shared databases. In: Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data, pp. 665–676 (2007)
55. Newman, S.: Monolith to Microservices: Evolutionary Patterns to Transform Your Monolith. O'Reilly Media (2019)
56. Papenbrock, T., Naumann, F.: A hybrid approach for efficient unique column combination discovery. Proc. der Fachtagung Business, Technologie und Web (BTW). GI, Bonn, Deutschland (accepted) Google Scholar (2017)
57. Phil, B., Giunchiglia, F., Kementsietsidis, A., Mylopoulos, J., Serafini, L., Zaihrayeu, I.: Data management for peer-to-peer computing: a vision. In: 5th International Workshop on the Web and Databases, WebDB 2002 (2002)
58. Podlesny, N.J., Kayem, A.V., Meinel, C.: Attribute compartmentation and greedy UCC discovery for high-dimensional data anonymization. In: Proceedings of the 9th ACM Conference on Data and Application Security and Privacy, pp. 109–119 (2019)
59. Podlesny, N.J., Kayem, A.V., Meinel, C.: Identifying data exposure across high-dimensional health data silos through Bayesian networks optimised by multigrid and manifold. In: 2019 IEEE 17th International Conference on Dependable, Autonomic and Secure Computing (DASC). IEEE (2019)
60. Podlesny, N.J., Kayem, A.V.D.M., Meinel, C.: Towards identifying de-anonymisation risks in distributed health data silos. In: Hartmann, S., Küng, J., Chakravarthy, S., Anderst-Kotsis, G., Tjoa, A.M., Khalil, I. (eds.) DEXA 2019. LNCS, vol. 11706, pp. 33–43. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-27615-7_3
61. Podlesny, N.J., Kayem, A.V.D.M., Meinel, C.: A parallel quasi-identifier discovery scheme for dependable data anonymisation. In: Hameurlain, A., Tjoa, A.M. (eds.) Transactions on Large-Scale Data- and Knowledge-Centered Systems L. LNCS, vol. 12930, pp. 1–24. Springer, Heidelberg (2021). https://doi.org/10.1007/978-3-662-64553-6_1
62. Podlesny, N.J., Kayem, A.V.D.M., von Schorlemer, S., Uflacker, M.: Minimising information loss on anonymised high dimensional data with greedy in-memory processing. In: Hartmann, S., Ma, H., Hameurlain, A., Pernul, G., Wagner, R.R. (eds.) DEXA 2018. LNCS, vol. 11029, pp. 85–100. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-98809-2_6

63. Record, A.S.: Distributed databases and peer-to-peer databases. SIGMOD Rec. **37**(1), 5 (2008)
64. Remacle, J.F., Shephard, M.S.: An algorithm oriented mesh database. Int. J. Numer. Meth. Eng. **58**(2), 349–374 (2003)
65. Rodríguez-Gianolli, P., et al.: Data sharing in the hyperion peer database system. In: Proceedings of the 31st International Conference on Very Large Data Bases, pp. 1291–1294. Citeseer (2005)
66. Ruiz, J.A.Q., Naumann, F., Abedjan, Z.: Datasets profiling tools, methods, and systems (11 June 2019). US Patent 10,318,388
67. Seol, E.S., Shephard, M.S.: Efficient distributed mesh data structure for parallel automated adaptive analysis. Eng. Comput. **22**(3–4), 197–213 (2006)
68. Seol, E.S.: FMDB: flexible distributed mesh database for parallel automated adaptive analysis. Rensselaer Polytechnic Institute Troy, NY (2005)
69. Shirazi, F., Keramati, A.: Intelligent digital mesh adoption for big data (2019)
70. Soria-Comas, J., Domingo-Ferrer, J.: Big data privacy: challenges to privacy principles and models. Data Sci. Eng. **1**(1), 21–28 (2016)
71. Sweeney, L.: Simple demographics often identify people uniquely. Health (San Francisco) **671**(2000), 1–34 (2000)
72. Sweeney, L.: Achieving k-anonymity privacy protection using generalization and suppression. Int. J. Uncertain. Fuzziness Knowl. Based Syst. **10**(05), 571–588 (2002)
73. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertain. Fuzziness Knowl. Based Syst. **10**(05), 557–570 (2002)
74. Tassa, T., Mazza, A., Gionis, A.: k-concealment: an alternative model of k-type anonymity. Trans. Data Priv. **5**(1), 189–222 (2012)
75. Terrovitis, M., Mamoulis, N., Kalnis, P.: Privacy-preserving anonymization of set-valued data. Proc. VLDB Endow. **1**(1), 115–125 (2008)
76. Wong, R.C.-W., Fu, A.W.-C., Wang, K., Pei, J.: Anonymization-based attacks in privacy-preserving data publishing. ACM Trans. Database Syst. **34**(2), 1–46 (2009)
77. Wu, X., Li, N.: Achieving privacy in mesh networks. In: Proceedings of the 4th ACM Workshop on Security of Ad Hoc and Sensor Networks, pp. 13–22 (2006)
78. Xiao, X., Tao, Y.: Anatomy: simple and effective privacy preservation. In: Proceedings of the 32nd International Conference on Very Large Data Bases, pp. 139–150. VLDB Endowment (2006)
79. Zhang, X., Liu, C., Nepal, S., Chen, J.: An efficient quasi-identifier index based approach for privacy preservation over incremental data sets on cloud. J. Comput. Syst. Sci. **79**(5), 542–555 (2013)
80. Zhang, X., Yang, L.T., Liu, C., Chen, J.: A scalable two-phase top-down specialization approach for data anonymization using MapReduce on cloud. IEEE Trans. Parallel Distrib. Syst. **25**(2), 363–373 (2014)