A Federated In-Memory Database System For Life Sciences

Matthieu-P. Schapranow[1], Cindy Perscheid[1],
Alf Wachsmann[2], Martin Siegert[2], Cornelius Bock[1], Friedrich Horschig[1],
Franz Liedke[1], Janos Brauer[1], and Hasso Plattner[1]

[1] Hasso Plattner Institute, Enterprise Platform and Integration Concepts,
August-Bebel-Str. 88, 14482 Potsdam, Germany
{schapranow|cindy.perscheid|plattner}@hpi.de
{cornelius.bock|friedrich.horschig|franz.liedke|janos.brauer}@student.hpi.de
[2] Max Delbrück Center, Robert-Rössle-Str. 10, 13125 Berlin, Germany
{martin.siegert|alf.wachsmann}@mdc-berlin.de

Abstract. Cloud computing has become a synonym for elastic provision of shared computing resources operated by a professional service provider. However, data needs to be transferred from local systems to shared resources for processing, which might results in significant process delays and the need to comply with special data privacy acts. Based on the concrete requirements of life sciences research, we share our experience in integrating existing decentralized computing resources to form a federated in-memory database system. Our approach combines advantages of cloud computing, such as efficient use of hardware resources and provisioning of managed software, whilst sensitive data are stored and processed on local hardware only.

Keywords: Federated In-Memory Database, Cloud Computing, Distributed Data Processing

1 Introduction

Cloud computing has been an emerging trend in information technology in the recent years, which abstracts computing power from physical hardware. Entities with limited or no physical hardware were early adapters of cloud computing, e.g. private households and Small and Mid-size Enterprises (SMEs) [29]. However, large companies and research facilities are reservedly moving their core business processes towards cloud computing environments due to legal regulations and various concerns although they would benefit equally from advantages, such as consolidation of hardware and improved use of available resources [16].

In the given work, we introduce a Federated In-Memory Database (FIMDB) system using unique hybrid cloud approach eliminating the need for transferring data to central cloud infrastructures. We focus on the specific requirements of large enterprises and research facilities with the example of a concrete use case taken form life sciences, i.e. data processing and analysis of Next-Generation

Sequencing (NGS) data. We share experiences of providing managed services through our cloud platform we.analyzegenomes.com whilst storing sensitive data on local, on-premise computing resources in our FIMDB. Instead of data, we move algorithms, maintain a single source of truth, eliminate data duplication through data copying, and incorporate existing, local on-premise computing resources for data processing.

Fig. 1 depicts our FIMDB software architecture modeled as Fundamental Modeling Concepts (FMC) block diagram [15]. The sum of all local database instances forms the FIMDB system whilst sensitive data and computing resources reside locally. In the remainder of the work, we use the terms customer and service provider. We refer to a customer as the user of a service provided by a service provider. Examples for customers are hospitals or research sites whilst central computing or bioinformatics centers are examples for service providers.

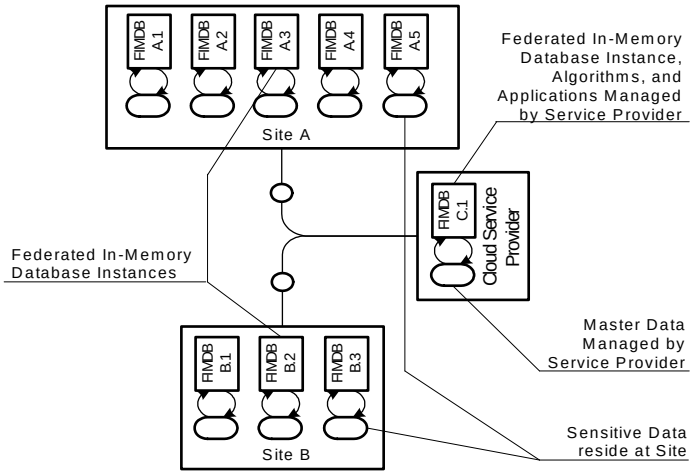


Fig. 1: Data reside at local computing resources whilst the service provider manages algorithms and apps remotely in the FIMDB system.

Our contribution is structured as follows: In Sect. 2 our work is set in the context of related work whilst we define cloud computing methods in Sect. 3. We introduce our FIMDB approach in Sect. 4 and share real-world experiences for a concrete life science application example in Sect. 5. In Sect. 6 we discuss our findings and our work concludes with an outlook in Sect. 7.

2 Related Work

The National Institute of Standards and Technology (NIST) defines cloud computing as "... a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources [...] that can be

rapidly provisioned and released with minimal management effort or service provider interaction" [19]. We follow this definition and believe that cloud computing is the transition from individual physical resources, e.g. servers, CPUs, and cores, to virtually infinitely available, scalable computing resources. Furthermore, the NIST distinguishes between the service models: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). We focus on the service model SaaS as we believe managed services help to reduce Total Costs of Ownership (TCO) of complex applications, e.g. NGS data processing as described in Sect. 5.

Gartner research expects hybrid cloud computing to reach its plateau within the next two to five years [9]. We concentrate on a hybrid cloud-computing approach bridging the gap between the provision of managed cloud apps and data that resides on local on-premise computing resources. It is driven by a) legal requirements restricting the exchange of sensitive data, e.g. patient data, and b) the sheer amount of data, e.g. hundreds of GB, which consume a significant amount of time for data transfer to centralized cloud resources.

A spectrum of grid middleware systems was developed during the peak of the grid-computing era. They were prominently used by researchers but rarely by business users. Meanwhile, only a small fraction of them is still used, e.g. Globus or UNICORE, whilst others were discontinued, e.g. gLite [3,4,27]. For example, the Globus Genomics project is a middleware for the compute-intensive data processing of genomic data on top of Amazon's Elastic Computing Cloud (EC2) incorporating Galaxy as workflow management system [1,10,3]. It addresses researchers with a certain IT and bioinformatics background requiring them to move acquired raw data from local sequencing centers to cloud-computing resources. It is a beneficial alternative if no local computing resources are available as discussed in Sect. 6.2. However, we consider research centers and hospitals with existing on-premise computing resources and motivate the provision of managed algorithms to them.

Tab. 1 compares physical locations of selected infrastructure components per cloud category. Our approach extends the flexible hybrid cloud approach by processing data on existing on-premise infrastructure components whilst enabling provision of managed serviced by a managed service provider. The remaining infrastructure components reside either locally or remotely or both locally and remotely comparable to the hybrid cloud approach.

3 Transferring Data to Computing Resources

Today, cloud computing is often used as a metaphor for consolidation of hardware resources by major public cloud service providers, e.g. Microsoft, Google, Backspace, and Amazon [26,6]. We define the user groups as depicted in Fig. 2 consuming cloud services to understand their different requirements:

A: Large enterprises maintain their existing on-premise server systems and use cloud computing for outsourcing of selected services that typically do not involve sensitive data,

Component	Public	Private	Hybrid	FIMDB
Apps	R	L	L/R	L+R
Data	R	L	L/R	L
Runtime Environment	R	L	L/R	L+R
Middleware	R	L	L/R	L+R
Operating System	R	L	L/R	L+R
Virtualization	R	L	L/R	L+R
Physical Servers	R	L	L/R	L+R
Storage Subsystem	R	L	L/R	L+R
Network	R	L	L/R	L+R

Table 1: Physical location of selected IT components per cloud category (L = Local on premise, R = Remote off premise, / = Either or, + = And).

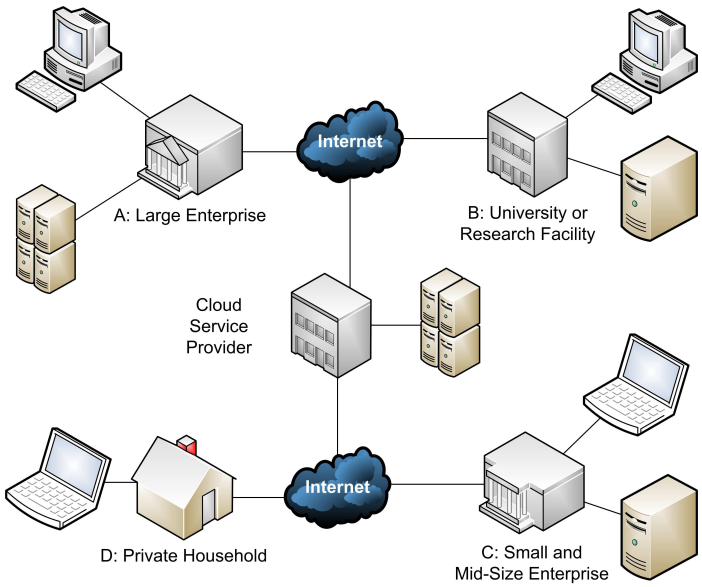


Fig. 2: Categories of cloud service users.

B: Universities or healthcare providers have existing on-premise server systems but need to consolidate software and hardware across multiple sites,
C: Small and Mid-size Enterprises (SMEs) might have limited set of local hardware resources or outsource specific software services, such as financials, from the beginning, and
D: Private households do not have any server systems on premise and consume cloud services for their private purposes only.

Tab. 2 compares the aforementioned user groups of cloud services and their requirements, which can be subsumed in the following categories:

I: Users are characterized by existing on-premise hardware resources, high computing power, a variety of local software apps, and high network bandwidth, e.g. large enterprises, universities, and healthcare providers, and
II: Users are characterized by medium or low local hardware resources, a mix of local and cloud-based apps, and low to medium Internet bandwidths, e.g. SMEs and private households.

Currently, cloud service providers mainly focus on customers of Cat. II and customers of Cat. I that aim to reduce local hardware, software, and personnel. Cat. I user with specific legal restrictions regarding exchange and processing of data are not addressed, yet. We focus on the latter group in the remainder of the work focusing on how to enable them to benefit from cloud services, integrating their existing computing resources, and keeping their data on premise.

User Group	Hardware		Software		Band-	Cat.
	Local	Cloud	Local	Cloud	width	
Large enterprises	↗	↘	↗	↘	↗	I
Universities or healthcare providers	↗	↘	↗	→	↗	
Small and mid-size enterprises	→	→	→	→	→	II
Private households	↘	↘	→	→	↘	

Table 2: User characteristics (↘ = Low, → = Medium, ↗ = High).

From an engineering perspective, consolidation of existing hardware resources in data centers, improved use of asset utilization, and near-instantaneous increase and reductions in capacity are obvious benefits of cloud computing [16]. From a business perspective, TCO are optimized by incorporating cloud computing since the service provider is able to deploy processes and tools that enable management of hundreds or thousands of similar systems at the same time, e.g. to perform regular software updates on all systems in parallel. Cloud computing provides a moderate way of scaling for end users, i.e. they can transparently extend their hardware resources for a selected period of time without the need to have the maximum capacity of required hardware permanently available [2]. For example, consider an online shop of a department house, which needs additional resources during seasonal sales.

A major concern of private and enterprise consumers equally to use cloud computing infrastructure is the privacy of their data once they are moved to the cloud, e.g. accidental disclosure or targeted attacks [22]. For private users, it might be acceptable to move photos, videos, or music to cloud storages since they believe these data only have individual value. For enterprises, it would be a fiasco if confidential data stored in cloud storage is exposed to unauthorized persons, e.g. competitors. Although the cloud storages are often maintained at a higher level than local computer resources, it remains still a major concern of end-users [14]. Therefore, large enterprises still prefer to select private cloud solutions to have greater control about asset and data [6]. Though, specific enterprise users are even not allowed to move their data to a data center outside of their premises, e.g. consider hospitals or healthcare providers dealing with patient data, which are highly regulated with respect to data processing [5].

However, does cloud computing necessarily involve the outsourcing of hardware to a service provider or is it valid to keep hardware and data locally whilst benefiting from managed services and improved efficiency of existing resources?

4 Transferring Algorithms to Data

Latest medical devices generate more and more fine-grained diagnostic data in a very short period of time, which motivates our FIMDB approach [21]. Even with increasing network bandwidth, sharing data results in significant delay due to data duplication when following state-of-the-art models as outlined in Sect. 6. Thus, we focus on sharing data between research sites without data duplication.

Today, medical data has becomes more and more available in digital formats, e.g. laboratory results, personal genomic data or Electronic Health Records (EHR) [13]. Sharing of medical data between clinical experts and the integration into clinical software systems is the foundation for discovery of new medical insights [28]. However, personal data requires very specific handling and exchange is limited, e.g. due to legal and privacy regulations, such as the Data Protection Directive of the European Union [8]. The collaboration of international life science research centers and hospitals all over the globe is important to support the finding of new scientific insights, e.g. by sharing of selected knowledge about existing patient cases. However, collaborations face today various IT challenges, e.g. heterogeneous data formats and requirements for data privacy.

4.1 Interconnecting International Genomic Research Centers

What is the meaning of sharing data in the era of cloud computing? To discuss this, we consider the following real-world example. A clinician needs to choose the optimal chemotherapy for a tumor patient once the tumor has been removed. Therefore, the clinician orders whole genome sequencing of the tumor sample to identify driver mutations. The tumor sample is sent to the pathology department to perform the required wet lab work. Data acquired from NGS devices result in up to 750 GB of raw data per single patient, which requires data pre-processing

steps, such as alignment of chunks to a reference and variant calling, prior to its use in context of clinical decision support [25]. The latter requires dedicated bioinformatics expertise, hardware and software expertise, and trained staff on site, which results in additional costs for hospitals. Instead of the local department of pathology, a cloud service provider can provide tools for data analysis, trained staff, and required hardware in a more efficient way by offering the same service to multiple clients.

When considering an exclusively available 1 Gb network connection between the local site and the cloud service provider and a payload bandwidth ratio of 75 %, it consumes 8,000 s or approx. $2h14m$ to transfer the data from the local site to the cloud service before any processing can start.

Our FIMDB system builds on the observation that a significant amount of time is consumed by transferring big medical data sets to computing resources before the actual processing can start. If the time for data transfer exceeds the execution time of the algorithms by far, it might be more efficient to transfer the algorithms to the local site instead. Thus, only the algorithms need to be transferred, which is typically orders of magnitudes smaller in size than the data to be processed. Examples for algorithms are alignment and variant calling tools in the size of some MB instead of hundreds of GB of raw data per tumor sample. For example, 14 MB for GATK 3.1.1 variant calling algorithm is exchanged within approx. 0.15 s in our real-world example.

However, does the idea of transferring algorithms instead of data conflicts with sharing data? We define "data sharing" as the process of granting access to a data set without the need to replicate it, i.e. sharing does not involve copying. Eliminating the need for data replication results in the following advantages when dealing with big medical data sets:

- No latency for data processing as big medical data resides locally,
- Single source of truth eliminating data redundancy, and
- Changes are applied directly to original data eliminating sync conflicts.

4.2 Patient Data vs. Master Data

In the following, we distinguish data in regard to their privacy as patient and master data. Patient data subsumes all kinds of personal data referring to an individual, e.g. name, birth date or personal genome, which involve specific steps to guarantee privacy. In contrast, master data refers to all kinds of sharable data required for various data operations, e.g. disease classifications, publications, or genomic annotations. As master data is rarely updated, existing caching approaches are efficient, which keeps master data transfer at a minimum, e.g. local file system cache or database result caches.

Based on this classification, patient data is considered as sensitive data that needs to reside at their current locations, e.g. at the local hospital site, to comply with privacy regulations. In contrast, master data will be managed at the central site to optimize maintenance, but can also be shared across and accessed by various sites, e.g. the genome annotations that can be incorporated to perform local genome data analysis.

4.3 Managed Services

Cloud-based infrastructures also offer managed services, e.g. which are hosted, configured and maintained by the service provider keeping the costs for operation minimal from the customers perspective. This approach is also referred to as SaaS or hosted software, where the complete maintenance effort for the software is handled by the service provider [2]. Providing access to managed services requires the isolation of customer-specific data, e.g. the execution details of specific genome data analyses. As patient data are not copied to the service provider there is no need for additional data privacy and security measures.

With regards to complexity of genome data processing and analysis, the advantages of hosted services should also be available for a federated system. As a result, we consider the provision of managed services as an integral aspect for a next-generation genome data processing and analysis platform [21]. The provider of the managed services has the required expertise in bioinformatics, software engineering as well as access to infrastructure and hosting structures. It manages algorithms for processing of medical data and maintains all master data incorporated by the algorithms at the central site.

Consuming Managed Services A customer accesses the managed service of the service provider, e.g. a web application provided via an internal web page. The customer uses her/his credentials for authentication and to protect customer-specific data and results. The service provider manages the web application and the logon centrally. The customer needs to specify additional input parameters, e.g. local patient data, depending on the application's purpose. In the aforementioned NGS use case, a clinician wants to process the raw NGS data acquired by a sequencing device in the department of pathology. The clinician creates a new task using the web application. Reference genome and genetic annotation data are considered as master data whilst patient-specific NGS data obtained from the tumor sample are considered as patient data. The service provider manages the master data and the web application whilst patient data must reside at the local site within the department of pathology.

Data Processing We refer to hardware at the customer site as local computing resources whilst we refer to remote computing resources for infrastructure physically hosted at the service provider. Processing of sensitive, patient-specific data is performed on the local computing resources only. All local and remote computing resources together form the FIMDB system, which might also involve multiple distributed sites connected to a single infrastructure provided by a service provider. Local hard- and software needs to be configured once to connect to the existing FIMDB system using existing infrastructure components, e.g. Virtual Private Network (VPN). Furthermore, apps need to be configured, e.g. to incorporate master data provided by the service provider. Master data are shared using established network protocols, e.g. Common Internet File System (CIFS), Server Message Block (SMB), or Network File System (NFS).

Setup and Configuration to Access Managed Service The site administrator to connect a local site to the FIMDB system performs the following steps once during setup phase:

- Establish site-to-site VPN connection with the service provider [20],
- Configure services directory exposed by the services provider locally, which contains managed algorithms for execution on local data,
- Install local IMDB instances and join them to the FIDMB system,
- Subscribe to selected app, and
- Configure app, e.g. user accounts, home directory, and access rights.

The user performs the following steps:

- Log in to managed app with personal credentials,
- Application-specific configuration, e.g. specify FASTQ files for processing,
- Trigger service execution, e.g. submit request to process local files, and
- Investigate results, e.g. use genome browser to analyze genetic variants.

5 Use Case: Processing and Analysis of Genome Data

In the following, required steps for setup and configuration of a managed service as defined in Sect. 4.3 are outlined. The given application scenario focuses on the processing and analysis of genome data as managed service. NGS data is created at decentralized research sites or sequencing centers and must reside locally due to legal restrictions for patient-specific data. However, to assess treatment alternatives, the comparison of the concrete patient case with similar patient cases stored at individual partner sites is required. Both, the research sites and the managed services sites, can consist of multiple computing nodes. In our application scenario, the research site was equipped with 150 computing nodes and the service provider with 25 high-end computing nodes.

5.1 VPN Connection

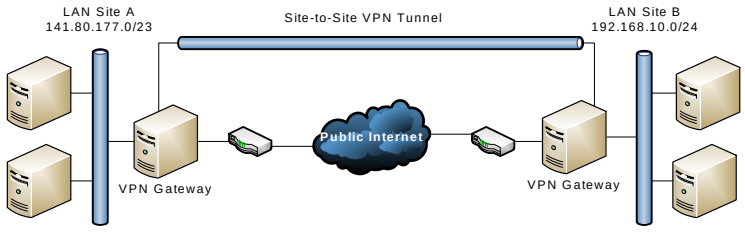


Fig. 3: Site-to-site VPN interconnecting LANs of site A and site B.

The network team of the local IT department needs to install and configure the VPN client. For the application scenario, OpenVPN version 2.3.5 was used

and configured to establishing a secured bidirectional site-to-site VPN tunnel. The VPN tunnel connects the Local Area Networks (LANs) at site A and site B via the public Internet as depicted in Fig. 3. In the typical VPN setup multiple VPN clients connect to a corporate network via a single VPN server, i.e. the corporate network is extended and clients consume corporate services in the same way they would access them being physically connected to the corporate network. In contrast, the site-to-site setup connects multiple LANs with each other, i.e. multiple LANs are connected forming a dedicated virtual network across all network topologies being able to create any kind of point-to-point connections. In the given application scenario, the local research site configured a single system as gateway system for the VPN connection while the updated network routes were pushed to individual computing nodes. Thus, the configuration efforts were minimized while a single point of maintenance was established.

5.2 Configure Remote Services Directory

The service provider grants access to her/his managed algorithms. In the application scenario, services are either file- or database-based. File-based services, e.g. the alignment algorithms Burrows Wheeler Aligner (BWA) or Bowtie, are exposed as runtime binaries using a Network File System (NFS) [18,17]. Therefore, the service consumer needs to create a local mount point and add the configuration for automatically mounting the remote service directory to all local computing nodes. In our given scenario, local sites integrated the configuration in their puppet scripts to deploy configuration to all involved computing nodes. Database-based services, e.g. stored procedures or analytical queries, are deployed through the FIMDB system. They are available once local database instances are connected to the FIMDB landscape without additional configuration.

5.3 Install Local IMDB Database Instance

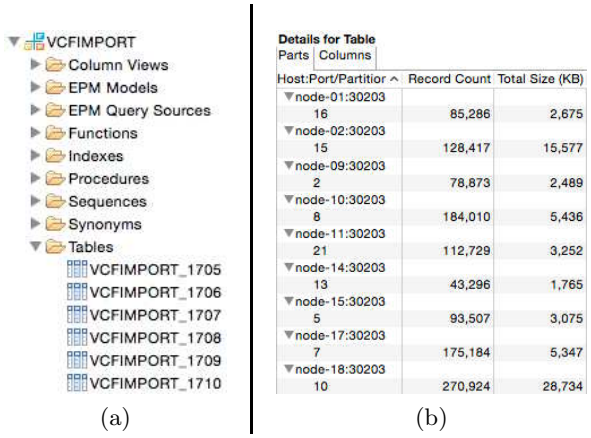
For each local computing node, a dedicated database instance needs to be installed and configured to connect to the FIMDB landscape. We incorporated SAP HANA version 1.00.82.394270 as our in-memory database system in landscape mode to form a distributed database [7]. The required database software is provided via the dedicated remote services directory. Thus, after mounting the services directory, the installation of the local database instance needs to be performed. For minimizing the efforts of installing the database instances, we incorporated the parameter-based installation, i.e. all parameters for the installation were predefined and provided as command parameters, which was executed in parallel across all nodes at the same time using the Linux tool Parallel Distributed SHell (PDSH) version 2.29 [11]. As a result, the required binaries were copied to the local database nodes, the local instances started, and registered online with the SAP HANA master server without any configuration downtime of the FIMDB system. In the concrete use case, the service provider invokes the SAP HANA Database Lifecycle Manager with the `addhosts` command [23]:

```
./hdblcm --action=add_hosts --addhosts=node-01,...,node-25
```

5.4 Subscribe to Managed Service

In the given application scenario, the managed service for processing and analysis of genome data was provided as a web application, which was accessible via any Internet browser. The web application was hosted at the site of the service provider and can be accessed by users using the URL of the application. The service provider supports the use of local user accounts and the integration of existing authentication providers, e.g. OAuth 2.0, for authentication [12].

Customer The application administrator of the research site subscribes to the managed services for the entire site or research department and access is granted to administer the application and settings. She/he is responsible to maintain user groups and access rights for users of the research site within the application. The application administrator performs the mapping of application users to corresponding database users and roles while the service provider maintains users and roles in the database.



(a) (b)

Fig. 4: Screenshots: (a) Database schema of the FIMDB grouping tables, functions, and stored procedures; (b) partitioning details of a FIMDB table.

Service Provider The service provider defines a dedicated database schema per site. A database schema is a container for a set of database tables, functions, and stored procedures as depicted in Fig. 4 on the left. Each database schema is kept isolated, i.e. tenant-specific data is separated to ensure data privacy [24]. The database administrator maintains specific user roles per tenant and grants them access to their tenant-specific database schemes. Each database schema is partitioned across a tenant-specific resource set, i.e. a local subset of the overall computing nodes, which are used for storing and processing the data. The database

administrator can update the list of computing nodes online without interfering running operations, i.e. data is repartitioned without any database downtime. Furthermore, the FIMDB administrator can assign additional resources to a resource set, e.g. to ensure scalability by adding resources of the service provider. Fig. 4 on the right depicts selected details of a database table partitioned across the FIMDB. For example, the first line describes that database table chunk 16, which has a cardinality of 85,286 entries and a size of 2.6 MB, is stored on the computing resource named `node-01`.

5.5 Configure Selected Service

The user of the research site accesses the managed service using the URL of the web application. The web application is accessed via the VPN connection, i.e. all data is exchanged via the secured tunnel. The end user is able to maintain her/his personal profile and configure application settings. In the application scenario, each end user was able to define her/his local home directory, which contains all genome data they were working on.

6 Evaluation and Discussion

Description	6.1	6.2	6.3	6.4	6.5
Export data from local system at site A	✓	✓	✗	✗	✗
Upload local data to shared cloud storage or cloud app resp.	✓	✓	✓	✓	✗
Sync data between shared cloud services or cloud apps resp.	✓	✗	✓	✗	✗
Sync data from cloud service B to local site B	✓	✓	✗	✗	✗
Import local data at site B to local system B	✓	✓	✗	✗	✗

Table 3: Comparison of cloud setups and involved process steps for sharing data.

In the following, selected state-of-the-art cloud service approaches and their applicability to implement data sharing are compared. We are focusing on data sharing as it results in isolated copied data artifacts having issues, e.g. conflict management and data redundancy. Our FIMDB system eliminates redundancy and the need for synchronization of changes by acting as the only source of data where all operations are performed on. Tab. 3 compares individual cloud setups and the involved amount of data duplication resp. data transfer. Compared to all other setups, our FIMDB approach does not require any upload of data, which is shown in the application scenario described in Sect. 5.

6.1 Local Systems and Multiple Cloud Service Providers

Sharing data from a local system, e.g. a Hospital Information System (HIS), via a single cloud service involves individual cloud service provider per local site as

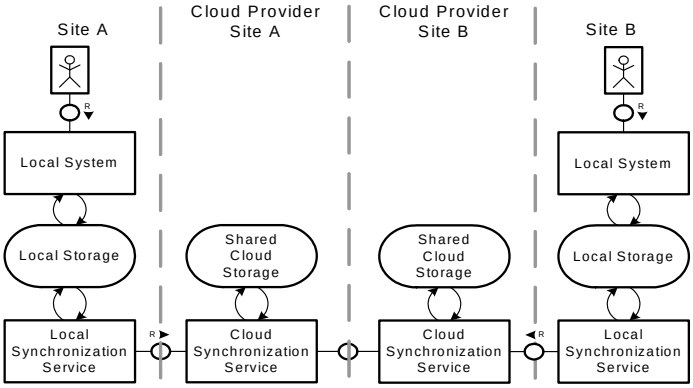


Fig. 5: Multiple sites interconnected via individual cloud providers.

depicted in Fig. 5. Local site A uses cloud provider A whilst local site B uses cloud provider B. Synchronization between local sites and their cloud service provider as well as between multiple cloud service providers is required.

6.2 Local Systems and Single Cloud Service Provider

The current configuration incorporates a single cloud provider used by all involved local sites instead of multiple cloud service provides as given in Sect. 6.1. Regular data synchronization between local sites A, B, and the shared cloud storage of the cloud provider occurs. It involves data ex- and import from local systems, i.e. data duplication requiring conflict management. A single cloud service provider is usually in place when you need to rely on locally installed software at various sites. It can be considered as a first transition towards a cloud-based software deployment model [19].

6.3 Cloud-based Software of Different Cloud Service Providers

Sharing data between apps of different cloud providers requires data synchronization as described in Sect. 6.2. Fig. 6 on the left depicts two sites A and B interconnected via individual cloud service providers hosting individual apps. Data synchronization is performed transparently between cloud service centers of individual providers using high-speed interconnections. Due to higher network bandwidth, data exchange requires less time than data exchanged between cloud provider and local site. This approach offers full flexibility in terms of incorporated cloud software and local IT optimization. However, it requires high-speed interconnections between cloud providers and still results in data duplication.

6.4 Cloud-based Software of A Single Service Provider

Fig. 6 on the right depicts two sites A and B using one app provided by a single cloud service provider. Working with a cloud-based application of a single

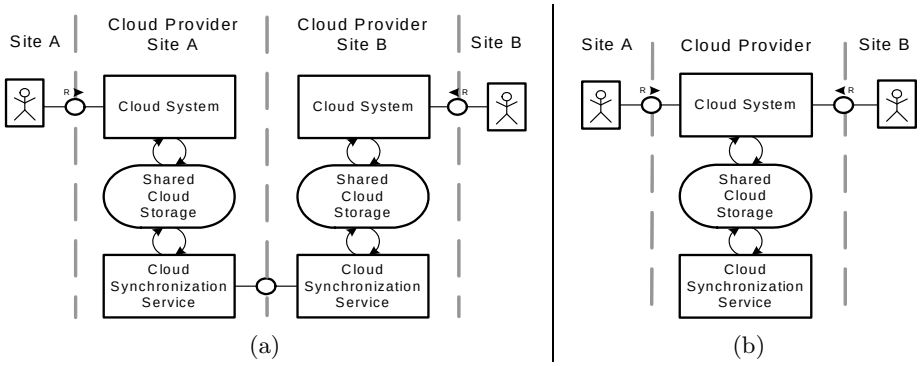


Fig. 6: Comparison: (a) multiple applications hosted by multiple cloud service providers; (b) single application hosted by the same cloud service provider.

provider does not eliminate the initial upload and import of data from the local site. However, the need for synchronization and conflict management are eliminated. Thus, only a single data transfer from all participating local sites to the cloud service provider is required.

6.5 Federated In-Memory Database Systems

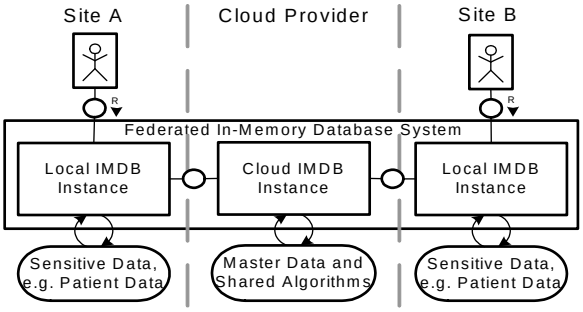


Fig. 7: Multiple IMDB instances forming the FIMDB.

Our FIMDB system eliminates even the initial data transfer to the cloud as discussed in Sect. 6.4 by importing data directly into the local database instance as depicted in Fig. 7. When incorporating our FIMDB system, there is no need to upload data via the Internet connection to a central cloud provider. Required algorithms and methods are either downloaded from the service directory, e.g. software executed by the OS, or accessed via the FIMDB system, e.g. stored database procedures. For distributed execution of the algorithms and management of results, we incorporate our worker framework [21]. It is a dedicated

runtime environment for executing arbitrary programs on either the operating system or database level. Thus, the FIMDB system eliminates processing latency due to data transfer while offering full access control for sensitive data by keeping them always on local hardware.

7 Conclusion and Outlook

In the given work, we presented details about our unique hybrid cloud computing approach enabling a) the use of cloud service even if legal requirements do not allow exchange of sensitive data with traditional cloud apps and b) the processing of huge data sets locally when their exchange would significantly delay the processing of data even with latest network bandwidths.

Based on a real-world scenario, we shared experiences implementing our cloud approach in a concrete life sciences use case. As a result, we were able to provide managed apps for research without moving high volume NGS data to central computing resources. For that, all local sites were interconnected via the Internet to our service infrastructure using secured VPN connection. Data was processed decentralized and results stored only in local database systems, which were configured to form our distributed FIMDB system.

We are convinced that sharing knowledge is the foundation to support research cooperation and to discover new insights cooperatively. Thus, our future work will focus on how to use our FIMDB system to encourage sharing of huge dataset in life sciences between research facilities without the need to create duplicates of sensitive data at partner sites.

References

1. Amazon Web Services, Inc.: Amazon Elastic Computing Cloud (EC2). http://aws.amazon.com/ec2/[1] (Jul 2015)
2. Armbrust, M., et al.: A View of Cloud Computing. Commun. ACM 53(4), 50–58 (Apr 2010)
3. Bhuvaneshwar, K., et al.: A Case Study for Cloud-based High-throughput Analysis of NGS Data using the Globus Genomics System. Computational and Structural Biotechnology Journal 13, 64–74 (2015)
4. CERN: gLite - Lightweight Middleware for Grid Computing. http://grid-deployment.web.cern.ch/grid-deployment/glite-web/introduction[1] (Apr 2014)
5. Dzombeta, S., Goldstein, O.: Patientendaten und Cloud Computing. http://www.persicon.com/news/items/patientendaten-und-cloud-computing.html[1] (Jul 2013)
6. Everest Global, Inc.: Enterprise Cloud Adoption Survey. http://www.everestgrp.com/wp-content/uploads/2014/03/2014-Enterprise-Cloud-Adoption-Survey.pdf[1] (Mar 2014)
7. Färber, F., et al.: SAP HANA Database: Data Management for Modern Business Applications. SIGMOD Rec. 40(4), 45–51 (Jan 2012)
8. Fears, R., et al.: Data Protection Regulation and the Promotion of Health Research: Getting the Balance Right. QJM 107(1), 3–5 (Jan 2013)

9. Gartner, Inc.: Gartner's 2014 Hype Cycle for Emerging Technologies Maps the Journey to Digital Business. http://www.gartner.com/newsroom/id/2819918[1] (Aug 2014)
10. Goecks, J., Nekrutenko, A., The Galaxy Team, J.T.: Galaxy: A Comprehensive Approach for Supporting Accessible, Reproducible, and Transparent Computational Research in the Life Sciences. Genome Biology 11(8), R86 (Aug 2010)
11. Grondona, M.A.: Parallel Distributed Shell (PDSH). https://code.google.com/p/pdsh/wiki/UsingPDSH[1] (Aug 2011)
12. Hardt, D.: RFC6749: The OAuth 2.0 Authorization Framework. http://tools.ietf.org/html/rfc6749/[1] (Oct 2012)
13. Jensen, P.B., Jensen, L.J., Brunak, S.: Mining Electronic Health Records: Towards Better Research Applications and Clinical Care. Nature Rev Genet 13(6), 395–405 (2012)
14. Kalloniatis, C., et al.: Migrating Into the Cloud: Identifying the Major Security and Privacy Concerns. IFIP Adv in Inform and Commun Tech 399, 73–87 (2013)
15. Knöpfel, A., Grone, B., Tabeling, P.: Fundamental Modeling Concepts: Effective Communication of IT Systems. John Wiley & Sons (2006)
16. Kundra, V.: Federal Cloud Computing Strategy. http://www.whitehouse.gov/sites/default/files/omb/assets/egov_docs/federal-cloud-computing-strategy.pdf[1] (Feb 2011)
17. Langmead B, S.S.: Fast Gapped Read Alignment with Bowtie 2. Nature Methods 9(357–359) (2012)
18. Li, H., Durbin, R.: Fast and Accurate Short Read Alignment with Burrows-Wheeler Transformation. Bioinform 25, 1754–1760 (2009)
19. National Institute of Standards and Technology: The NIST Definition of Cloud Computing: Recommendations of the National Institute of Standards and Technology. NIST Special Publication 800-145 (Sep 2011)
20. OpenVPN Technologies, Inc.: Site-to-Site Layer 3 Routing Using OpenVPN Access Server and a Linux Gateway Client. https://docs.openvpn.net/[1] (Feb 2012)
21. Plattner, H., Schapranow, M.P. (eds.): High-Performance In-Memory Genome Data Analysis: How In-Memory Database Technology Accelerates Personalized Medicine. Springer-Verlag (2014)
22. Ryan, M.D.: Cloud Computing Privacy Concerns on Our Doorstep. Commun. ACM 54(1), 36–38 (Jan 2011)
23. SAP SE: Add Hosts Using the Command-Line Interface. http://help.sap.com/saphelp_hanaplatform/helpdata/en/0d/9fe701e2214e98ad4f8721f6558c34/content.htm[1] (2014)
24. Schaffner, J.: Multi Tenancy for Cloud-Based In-Memory Column Databases: Workload Management and Data Placement. Springer (2014)
25. Schapranow, M.P., et al.: In-Memory Computing Enabling Real-time Genome Data Analysis. Int'l Journal On Advances in Life Sciences 6(1 and 2), 11–29 (2014)
26. Srinivasan, S.: Cloud Computing Evolution. In: Cloud Comp Basics, pp. 1–16. Springer (2014)
27. The UNICORE Forum e.V.: UNICORE - Documentation. https://www.unicore.eu/documentation/[1] (Jul 2015)
28. Wicks, P., et al.: Sharing Health Data for Better Outcomes on PatientsLikeMe. Journal Med Internet Res 12(2), e19 (Jun 2010)
29. Zhang, Q., Cheng, L., Boutaba, R.: Cloud Computing: State-of-the-art and Research Challenges. Journal of Internet Services and Applications 1(1), 7–18 (2010)

[1] All online references were checked on July 28, 2015.