A mobile computing platform for data and virtual machines migration between datacenters and private clouds: Design and use cases

Dan AVRAM[1,2]*, Ioana MATEI[1], Ioana APOSTOL[1], Ion BICA[1], Florin POP[2]

[1] Military Technical Academy "Ferdinand I", Romania

dan.avram@mta.ro (*Corresponding author), ioana.matei@mta.ro, ioana.apostol@mta.ro, ion.bica@mta.ro

[2] National University of Science and Technology Politehnica Bucharest, Romania

constantin.avram@stud.acs.upb.ro, florin.pop@upb.ro

Abstract: This paper presents a mobile computing platform for secure data and virtual machine migration between different datacenters and private cloud environments. Traditional network-based methods are often constrained by bandwidth, connectivity, and security requirements, especially in isolated or air-gapped scenarios. The proposed solution allows secure offline data transfer by physically moving data while integrating compute, storage, networking, and virtualization capabilities within a mobile architecture. This approach allows conversion, migration and execution of virtual machines between different infrastructures, providing interoperability and continuous operation. The platform is autonomous and designed for use in disaster recovery, field deployments and secure environments with no Internet connectivity. Our work shows that mobile datacenter architectures can be applied in efficient infrastructure-independent migration processes.

Keywords: Data migration, VM migration, VM conversion, Mobile datacenter.

Platformă mobilă de calcul pentru migrarea datelor și a mașinilor virtuale între centre de date și infrastructuri cloud private: proiectare și scenarii de utilizare

Rezumat: Această lucrare prezintă o platformă mobilă de calcul destinată migrării securizate a datelor și a mașinilor virtuale între centre de date diferite și medii cloud private. Abordările tradiționale bazate pe rețea sunt adesea limitate de constrângeri legate de lățimea de bandă, conectivitate și cerințe de securitate, în special în scenarii care presupun izolare fizică. Soluția propusă permite transferul securizat offline al datelor prin transportul fizic al acestora, integrând capabilități de calcul, stocare, interconectare în rețea și virtualizare într-o arhitectură mobilă. Platforma suportă conversia, migrarea și rularea mașinilor virtuale în infrastructuri eterogene, asigurând interoperabilitatea și continuitatea operațională. O astfel de platformă funcționează în mod autonom, fiind destinată utilizării în scenarii de recuperare în caz de dezastru, în implementări în teren și în medii securizate, indiferent de disponibilitatea conexiunii la Internet. Abordarea noastră demonstrează faptul că arhitecturile mobile de centre de date sporesc eficiența proceselor de migrare, desfășurate independent de infrastructura existentă.

Cuvinte-cheie: migrare date, migrare mașini virtuale, conversie mașini virtuale, centru de date mobil.

1. Introduction

Digital transformation and the rapid adoption of Cloud services have produced significant changes in the way organizations cover data management, storage, and the migration of IT workloads. Higher demands for flexible architectures and distributed and hybrid cloud models used by companies have raised new challenges in data transfer. Data security, both in transit and at rest, has become a main requirement, particularly in environments where regulatory compliance and constraints on protecting confidential information are imposed.

Organizations are now more interested in adopting private cloud solutions to obtain better control over their own data, improve security, and meet regulatory compliance and organizational requirements. Private cloud environments, unlike public clouds, allow organizations to retain full ownership of their infrastructure and data, which is an important requirement in sectors such as

https://doi.org/10.33436/v36i2y202608

defense or government. However, switching to private clouds introduces a higher degree of complexity in managing distributed and heterogeneous environments, where workloads must be migrated between different virtualization platforms, isolated infrastructures, or air-gapped setups.

Data migration is the process of transferring data, applications, or virtual machines (VMs) from one storage system, computing environment, or private cloud infrastructure to another, while preserving data integrity, consistency, and operational continuity (Jamshidi, Ahmad & Pahl, 2013; Strauch et al., 2014). Usually, migration processes are conducted over network connections between datacenters or cloud platforms. While network exposure provides flexibility and ease of use, it also raises vulnerabilities, increases the risk of data leaks and introduces other security concerns (Stoleriu, Petre & Pop, 2025). When large volumes of data, limited bandwidth, strict security policies, or isolated environments are involved, network-based transfers may be infeasible or inefficient. In these cases, where data is physically moved using dedicated storage solutions, offline data migration is a better alternative for the secure and reliable transfer of large amounts of data. Although major cloud providers offer mature data transfer solutions, such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP), these services are typically tied to their proprietary ecosystems and are primarily designed for data ingestion into their own cloud infrastructures.

Existing approaches to cloud migration often rely on software-level orchestration mechanisms or storage transfer devices, without providing an integrated, autonomous and mobile platform that can operate independently without the need for permanent infrastructure. Integrating heterogeneous hypervisors and planning VM migrations poses significant challenges, especially in environments with strict service-level requirements (He, Toosi & Buyya, 2021; Moravcik et al., 2024).

To address these limitations, this paper introduces a unified, mobile and secure platform that supports offline data and VM migration across different on-premises infrastructures. The proposed system aims to integrate computing, storage, networking, and virtualization components within a portable architecture that can operate autonomously and provide multi-protocol storage. Data transfer is performed offline by physically transporting the appliance, which eliminates the need for high-bandwidth network connectivity between sites. The platform is designed to operate independently of public network connectivity, making it suitable for field operations deployments, air-gapped infrastructures, disaster recovery scenarios, or security-constrained environments.

The novelty of the proposed solution lies in combining high-performance computing resources, self-encrypting storage systems, and flexible virtualization technologies into a compact and portable format. In contrast to traditional transfer devices, the proposed architecture integrates migration, storage exposure, virtualization, clustering and business continuity functions into a single, operationally autonomous system.

The remainder of this paper is organized as follows. Section II reviews the state of the art in data and VM migration, mobile datacenters and the limitations of existing approaches. The objectives, requirements and layered architecture of the proposed system design are outlined in Sections III and IV. The main use cases and operational scenarios of the proposed system are covered in Section V, along with a discussion and comparative analysis in Section VI, and conclusions in Section VII.

2. Background and related work

Given the challenges previously outlined, existing approaches to data and VM migration require further examination. This section covers the main approaches presented in the existing literature and technologies that are related to data and VM migration across distributed infrastructures, outlining the major limitations that motivate our approach. It covers data transfer methods, VM migration techniques and tools, new infrastructure paradigms, and the limitations of existing solutions.

2.1. Data migration between private clouds

Data migration between private clouds raises specific challenges compared to traditional inter-datacenter transfers. Most current approaches assume the availability of high-capacity and stable network connectivity; however, this condition is not always met inside resource-constrained environments, where connectivity limitations or service outages can have significant operational and security implications (Luo et al., 2020; Shen, Van Beek & Iosup, 2015).

In accordance with the U.S. National Institute of Standards and Technology (NIST), a private cloud is provisioned exclusively to a single organization, providing it with full control over infrastructure, data and security policies (Mell & Grance, 2011). Nonetheless, these environments often include heterogeneous virtualization platforms, custom configurations and isolated network domains, which affect interoperability and standardization. In addition, strict security and compliance requirements limits the use of public network infrastructure, requiring controlled or fully segregated data transfer mechanisms. When it comes to sensitive or large-scale workloads preserving confidentiality and data integrity during transit is of primary concern (Majigi et al., 2025). These problems are further intensified by requirements for data sovereignty, rigorous access control, and operations in isolated or air-gapped environments. Hence, data migration between private clouds has a higher level of complexity and risks related to integrity, downtime and interoperability (Majigi et al., 2025).

2.2. Inter-cloud VM migration

Transferring the entire system state, including memory, storage, and configuration, makes inter-cloud VM migration much more complex than data transfer. The conversion of a virtual machine from one hypervisor format to another is known as Virtual-to-Virtual (V2V). Even though it is important for interoperability across heterogeneous environments, V2V conversion generates challenges related to compatibility, driver adaptation, and configuration consistency (Pant, 2025).

Cold migration and live (hot) migration are the two main approaches to VM migration. Cold migration involves shutting down the VM to guarantee consistency and incurs service downtime, whereas live migration transfers a running system with minimal service interruption. Live migration relies heavily on network performance and can result in extra costs, mainly in remote locations (Pant, 2025). Homogeneous environments can be easily migrated using platform-specific solutions, such as VMware vMotion, Xen Motion, and Microsoft Hyper-V Live Migration (Hu et al., 2013; Kargatzis, Sotiriadis & Petrakis, 2017; Kaur, 2025). OpenStack offers mechanisms for cold and live migration. However, these mechanisms are limited to host-to-host migration within a single deployment. Cross-hypervisor migration can be achieved using tools such as Coriolis, which are typically designed for operation within local area networks (LANs), consequently limiting their usefulness to inter-datacenter transfers and air-gapped environments. Our proposed approach uses these tools while extending their applicability range.

The performance of migration over wide-area-networks (WANs) is also a significant challenge, as it is influenced by transfer duration, downtime, and resource consumption. Performance assessment relies on metrics such as total migration time, downtime, and transferred data volume, which can differ significantly based on dynamic network factors (Kaur, 2025; Zhang et al., 2018).

In large-scale data transfer scenarios, these limitations are even more significant, creating the need for alternative approaches. Physical data migration solutions were introduced by cloud providers as a means to overcome these constraints, such as the AWS Snow Family (Snowball, Snowmobile), Microsoft Azure Data Box, and Google Transfer Appliance. These systems allow the transfer of large amounts of data by means of secure, high-capacity storage devices with hardware-level encryption. Industry guidelines indicate that when network transfer times exceed 7–10 days, physical shipment becomes more efficient, particularly for datasets ranging from tens to hundreds of terabytes. These solutions are often tightly integrated with particular cloud ecosystems, but they only act as one-way transfer mechanisms, limiting interoperability and heterogeneous environments support while also addressing bandwidth constraints. Hence, they cannot handle

flexible, cross-platform virtual machine and data migration scenarios that demand interoperability and independence from vendor-specific infrastructures.

2.3. Mobile datacenter

Mobile Data Centers (MDCs) have emerged as a disruptive paradigm for rapid infrastructure deployment and infrastructure lifecycle management (Vishwanath et al., 2009). In contrast to traditional "brick-and-mortar" facilities that require years for design and commissioning, MDCs use modularity to achieve operational capability within months or even weeks (Vishwanath et al., 2009).

The concept behind cloudlets and edge-based datacenters, which are deployed near data sources or end users, is similar. Cloudlets are designed to reduce latency and the reliance on centralized cloud infrastructure by allowing for local processing and storage. Similar to mobile datacenters, they provide localized computing resources. Typically, they are smaller and more closely associated with edge computing scenarios (Satyanarayanan et al., 2011). The introduction of distributed mechanisms for VM migration across cloudlets has helped develop this concept by enabling workloads to be dynamically relocated between edge nodes while still maintaining service continuity. However, such approaches remain primarily focused on network-based live migration and are, therefore, dependent on connectivity and network performance (Singhai et al., 2018).

2.4. Limitations of existing approaches

The current approaches to migrating data and VMs are mostly network-centric and are limited by bandwidth and connectivity limitations. While methods such as DCCast (Noormohammadpour et al., 2017), QuickCast (Noormohammadpour et al., 2018), or Multiple Bulk Data Transfers Scheduling (MBDTS) (Wang et al., 2014) improve transfer efficiency, they depend on a stable WAN infrastructure. Similarly, NetStitcher (Laoutaris et al., 2011) and the Time-Shifted Multilayer Graph (TS-MLG) (Lin et al., 2016) introduce an in-network storage mechanism but do not support workload execution in disconnected environments.

Industrial solutions like the AWS Snow Family or Azure Data Box are capable of transferring large-scale data through physical transport, but are tightly connected to specific cloud ecosystems. Consequently, they offer limited support for direct migration of VMs between heterogeneous private clouds. Interoperability remains a major challenge (Cheng et al., 2016), as differences in VM formats, storage architectures, and networking are setbacks in migration and may lead to vendor lock-in. Additionally, these approaches do not deal with scenarios that require continuous connectivity, highlighting the requirement for integrated, mobile, and autonomous solutions to reliably and safely migrate complete VM systems across heterogeneous private cloud environments. These limitations show the necessity for integrated solutions that combine portability with enhanced capabilities for secure offline data and VM migration in heterogeneous environments.

3. System design goals and requirements

A key requirement for the proposed platform is to manage the migration of data and virtual machines between dispersed datacenters, including both private cloud architectures and different on-premises computing environments. The proposed system allows bidirectional transfer of workloads and large data volumes, so that resources can be ingested from external infrastructures or sent to other local sites. This process is complex, since there is no direct interoperability between different hypervisors, which often requires special format conversions or manual fixes to ensure architectural compatibility.

Virtual machine migration and conversion should implement the following objectives: (1) Ingestion from external virtualized environments: VMs originating from a source virtualization platform are converted (using a "migration and conversion" software component) into the

platform's internal VM format, allowing them to be stored and executed locally without dependence on the source infrastructure, (2) Migration to external virtualized environments: VMs stored within the platform will be converted into the specific format required by the destination virtualization platform.

The solution should also support mobile high-performance computing and business continuity scenarios. Achieving this requires transferring and hosting imported VMs within the platform's compute environment. These operations include transferring the VM from the external virtualization platform to its storage, converting the VM to the platform's native virtualization format, and configuring the platform's networking and firewall policies to replicate the security and connectivity parameters of the source environment.

The proposed solution should also function as an external storage device. It should support the following: file-based storage – the solution should be able to share storage resources through file-sharing protocols, such as NFS and SMB; object-storage – it should expose an S3-compatible Application Programming Interface (API) to support modern object-based workflows; and block-storage – the platform should be capable of providing high-performance block storage, exposing Logical Unit Numbers (LUNs) to external initiators. This direct mapping allows external hosts to treat the platform's storage as local raw disks, bypassing the overhead associated with network file system layers.

The system is expected to be transportable across a range of operational environments, with particular emphasis on reliability and fault tolerance. These qualities should be addressed at the architectural level through redundant power supplies on each device, a dual-controller storage system backed by fault-tolerant RAID arrays (e.g., RAID 5 or RAID 10), redundant network paths, and the selection of hardware capable of operating reliably at ambient temperatures of up to 40° C. To preserve hardware integrity during transport and field operation, all components should be housed within ruggedized transit enclosures to prevent shock and vibration damage. Data confidentiality and integrity constitute further essential concerns. These should be addressed through data-at-rest encryption on the storage system, which preserves confidentiality even during physical transport, together with a site-to-site VPN for inter-site data transfer and a remote-access VPN for secure user connectivity.

Scalability is another important capability of this system. Since the resources needed may vary significantly across operational contexts, the architecture must support implementations of different scales, allowing computing and storage resources to be provisioned as needed. This implies both the integration of additional hardware and the provision of sufficient network capacity to accommodate further expansion. Equally important is the ability of the virtualization platform and supporting system applications to adapt to these variations, enabling new resources to be added into the existing environment with minimal reconfiguration effort.

4. System architecture

In our approach, the design of the system architecture is logically partitioned into four layers: hardware, virtualization, service, and orchestration, as depicted in Figure 1. This layered representation provides an overview of the system's main components while abstracting away the implementation details, thereby decoupling the architectural design from any specific implementation.

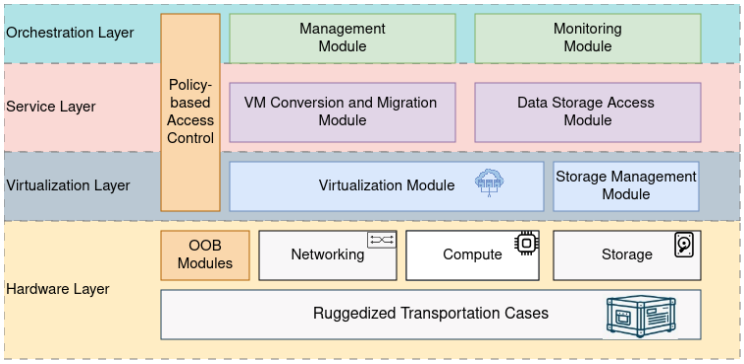

Figure 1. High-level architecture of the system

The present chapter offers a detailed presentation of each layer. For every layer, we discuss its role within the overall architecture, the functionalities it provides, its implementation choices, and the trade-offs that guided our design decisions.

4.1. Hardware layer

This layer is designed around a set of high-performance computing nodes interconnected with an all-flash storage array through a dedicated Fibre Channel Storage Area Network (SAN), to provide low-latency, high-throughput storage access. All hardware is intended to be distributed across a number of ruggedized transit enclosures conforming to the standardized 19-inch rack form factor. This enables the use of high-performance commercial off-the-shelf (COTS) servers and ensures compatibility with a wide range of standard equipment. The enclosures are envisioned to preserve operational mobility and structural integrity under field deployment conditions, allowing manual handling and short-range displacement by a small team without the need for mechanical lifting assistance, while remaining compatible with designated vehicular platforms for extended transport operations. Their number should be determined so as to achieve a balanced distribution of the system's total mass while accommodating the number and physical dimensions of the components.

As illustrated in Figure 2, the hardware architecture can be further divided into two subsystems: The Compute, Storage and Network subsystem, and the Out-of-Band (OOB) access subsystem, which incorporates a dedicated OOB firewall.

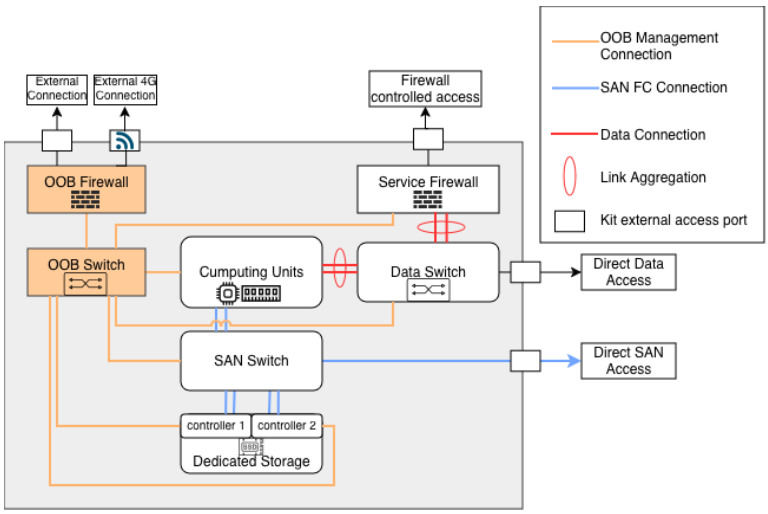

Figure 2. System architecture

The Compute, Storage, and Network subsystem may be implemented using one or more COTS servers, dimensioned according to the computational and memory demands of the operational context. Each server is required to provide multi-core processors with hardware virtualization support (Intel VT-x or AMD-V), a local SSD-based RAID 1 array for the operating

system and configuration data, a dedicated out-of-band management interface with remote KVM capabilities, and a minimum of two high-speed network interfaces and two Fibre Channel interfaces for connectivity to the service and storage fabrics, respectively.

The storage component may be implemented using a dedicated dual-controller storage appliance, which provides high availability and fault tolerance through controller redundancy. This appliance should incorporate high-speed, high-capacity Non-Volatile Memory Express (NVMe) Solid-State Drives (SSDs) with Self-Encrypting Drive (SED) capability, ensuring both storage performance and data confidentiality at the hardware level. A dedicated storage appliance, decoupled from the computing component, enables independent scaling of storage and computing resources.

The compute nodes should be interconnected with the storage appliance through a dedicated Storage Area Network (SAN) Fibre Channel (FC) switch. The SAN fabric facilitates interconnection with source or destination datacenters, allowing high-throughput, low-latency block-level storage access and efficient data migration to and from the system.

The Networking component provides connectivity between system components while enforcing strict traffic control through network segmentation and firewall policies. All inter-network traffic transmitted via Ethernet must pass through the system's primary firewall, where filtering and security policies are applied. This design ensures that all external communications are monitored and controlled before reaching internal services, preventing unauthorized lateral movement within the system. Access policies must be configured to meet the needs of the operational context. The firewall can also provide external access to the services through Port Forwarding/Virtual IPs configured on the external interface.

The OOB subsystem serves as the primary management plane, with operators performing configuration and administrative tasks either via a dedicated out-of-band management port or through a secure 4G LTE VPN tunnel. This subsystem can be implemented using a dedicated firewall with integrated 4G/5G cellular capabilities and VPN support, and one or more management switches, scaled according to deployment requirements.

4.2. Virtualization layer

The compute nodes host the virtualization infrastructure, which provides efficient resource allocation, workload isolation, and flexible deployment of the VMs that are required for running the system services and execution of imported workloads. The platform supports the creation and management of VMs, provisioning and administration of storage volumes, and scalability through the addition or removal of compute nodes. VMs are classified into two primary categories based on their functional role. The first category contains the permanent instances needed for the platform's services. These instances manage the internal functionality of the platform, including VMs dedicated to VM migration and conversion, storage management services, and a dedicated monitoring and management component. This separation is illustrated in Figure 3.

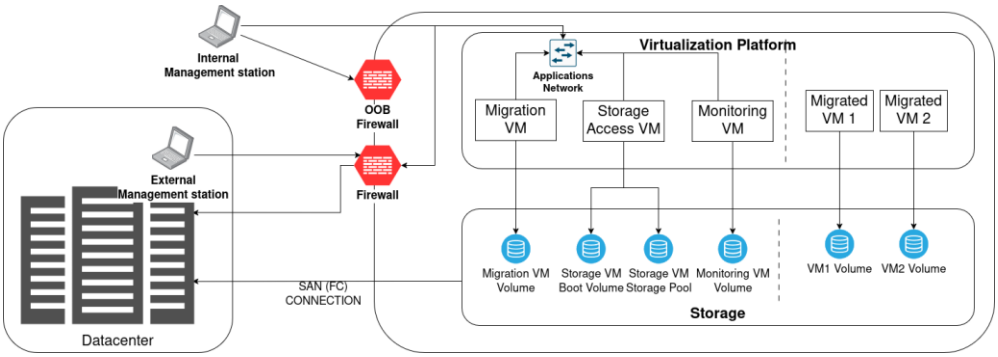

Figure 3. Reference connectivity scheme between the solution and an external datacenter

The second category comprises mission-specific VMs. These instances are represented by workloads migrated from external infrastructures to support business continuity, as well as temporary instances created during VM conversion.

The system is designed for compatibility with major virtualization platforms, including VMware ESXi, Proxmox, KVM, Hyper-V, Oracle, and OpenStack. Our approach uses OpenStack as the virtualization solution due to its widespread adoption as an open-source private cloud platform. It is compatible with Coriolis, a migration and conversion software solution capable of functioning as both the source and destination platform for the VM migration process. We recommend installing Openstack via Kolla Ansible, which provides containerized, production-ready deployment of OpenStack services and simplifies installation, upgrades, and maintenance. OpenStack manages block storage access through a service called Cinder. The hardware storage appliance should be selected based on its compatibility with Cinder to fully utilize OpenStack storage capabilities. This is achieved through vendor-provided Cinder drivers.

4.3. Service layer

This layer builds upon the virtualization layer. Its functionalities are modular and are deployed as permanent VMs.

Although the migration of a VM instance may be conceptually reduced to the transfer of a virtual disk from a source to a destination host through direct disk copy and VM recreation, or via the export and reimport of an OVA-formatted image, this process introduces a set of non-trivial compatibility challenges. These arise from discrepancies in underlying hardware architecture, hypervisor type and version, virtualized device configurations and network topology, all of which may impede the successful execution of the migrated workload in the destination environment.

The VM Conversion and Migration service is responsible for the reliable transfer of virtual machines between private datacenters and the platform and is required to accommodate the full range of VM migration scenarios identified in this work. It must support both using the platform as an intermediate storage location for VM transfers between datacenters and migrating VMs into the platform's own virtualization environment for local execution. Both scenarios may entail modifications to the VMs' resources — such as disk image conversion across hypervisor formats — which fall within the responsibility of this service. For this purpose, we propose the use of Coriolis, a commercial migration solution that is compatible with the most widely adopted virtualization platforms and meets all the requirements identified in the design.

The system is designed to present storage space to the user in three different forms: block, object, and file. The block storage part should be implemented in the Storage Management Module inside the Virtualization Module. The design of the system includes dedicated storage equipment capable of delivering block storage via Fibre Channel (FC) and/or iSCSI.

The File and Object storage capabilities should be implemented at the Service Layer, a VM running the necessary applications that implement the required protocols (SMB, NFS, rsync, S3). This VM will also allow the creation of different storage pools to further organize data. It will have at least two volumes attached, a permanent volume containing the operating system, the applications and configurations needed to provide storage access, and the others for the actual data storage.

As a solution to these requirements, we propose deploying TrueNAS SCALE as a VM. This will allow dynamically attaching volumes to it, as each operational context requires, while exposing the necessary file-level (NFS, SMB, FTP, SCP) and object-level (S3) access protocols over the network. TrueNAS natively supports the required file-level protocols (NFS, SMB, FTP, SCP); however, to expose an S3-compatible interface, we propose the deployment of RustFS or Garage within the same TrueNAS instance, thereby complementing the file-level access with a dedicated object-storage endpoint while using the same data pool.

4.4. Orchestration layer

The orchestration layer manages the platform state before, during, and after its operations, and can be implemented as a series of microservices running on permanent Virtual Machines. It includes a management component responsible for the configuration and management of the underlying hardware and system components, and a monitoring component that provides visibility

into the current state of the system, presenting to the operator the resource availability and utilization, error conditions and notifications, and the status of system services and hardware.

The management component should offer a centralized web interface, with links to the management endpoints of the platform's hardware and software components (e.g., server management interfaces, Coriolis transfer appliance, etc.). It should also offer configuration options for the solution, implemented through existing APIs: creating and attaching volumes to VMs, managing TrueNAS pools and volumes, increasing capacity for the Data Storage service, activating mission-specific services (S3, NFS), defining access policies from external networks, and allowing for direct console access to services VMs.

It should allow creating, managing and accessing new VMs for scenarios not covered by this work that could require more functionalities running on the service layer. This enables an operator to install specific tools inside the platform to further increase its capabilities. The Orchestration layer should also allow for quick platform setup before being deployed, adjusting platform configurations (services configuration, IPs, VPN), downloading the OpenStack openrc.sh file for direct OpenStack access, resource cleanup (resetting the platform to a default state, removing volumes, deactivating services, etc.), and complete client data and VM removal at the end of a mission. An additional section containing documentations and guides can also be added to support the operator.

The monitoring component collects data from different sources, such as traffic volume, latency, services status, and resource utilization using protocols such as SNMP or dedicated metrics APIs. We recommend Prometheus as the data collection engine with InfluxDB as storage for the time-series. This data should be presented to the operator through personalized Grafana dashboards composed of charts, tables and other components. Additionally, the monitoring component should detect problems and anomalies from the collected data by comparing the current state with predefined thresholds and alert the operator through notifications.

5. Use cases and scenarios

The proposed solution addresses three primary use cases. Although its flexibility and reconfiguration capabilities enable a broader range of applications, this work focuses on these three scenarios.

5.1. Data migration

This scenario addresses the challenge of migrating large volumes of data between two remote locations, where network-based transfer is either not practical or cost-prohibitive. This limitation requires a trustworthy physical transportation method, with data encryption at rest on the physical carrier throughout the whole transit process, mitigating risks associated with data loss, data breach, or unauthorized access.

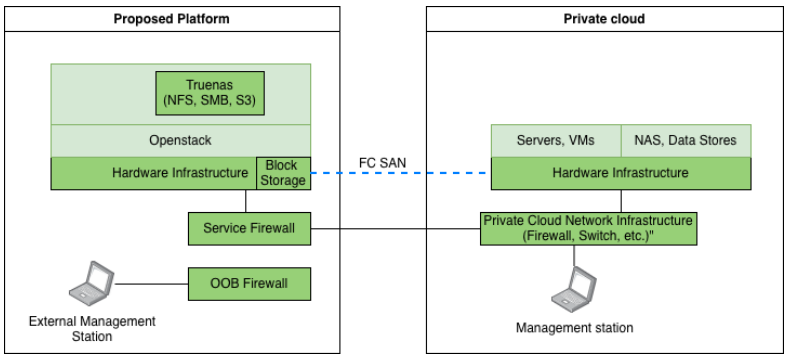

Figure 4. Connecting to the source/destination datacenter for data migration

The connection diagram is presented in Figure 4. Optionally, it can include a direct SAN Fibre Channel (FC) interconnect between the platform and the source datacenter, allowing direct

block-level access to the source storage system. File and object storage migration is performed by an authorized operator who, having access to the source datacenter, transfers data to the platform via the firewall-exposed NFS, SMB, and S3 endpoints, populating the pre-configured storage pools provisioned within the platform's local infrastructure.

Upon completion of the data copying process, the platform is physically transported to the destination site, where the inverse migration procedure is performed, restoring the replicated data into the destination infrastructure. This use case is suited for finance and healthcare companies or government agencies that have large amounts of data stored locally and want to move it into a centralized private cloud without using a public network, while maintaining strict data privacy requirements.

5.2. VM migration

This study considers the issue of moving multiple VM instances between two remote locations — all performed in a secure, flexible manner, supporting a broad range of use cases regardless of network availability or access to external infrastructure. Similar to the scenario described in Section 5.1, a physical connection must be established between the solution and the Source/Destination Datacenter.

The firewall enforces access control by exposing only the required services. An operator connects to the Coriolis appliance via the External Management Station to configure the source and destination hypervisors, target VM, and network mappings. The Coriolis software then performs the replication of the designated VM into the platform's OpenStack instance, producing a fully operational VM replica within its local infrastructure.

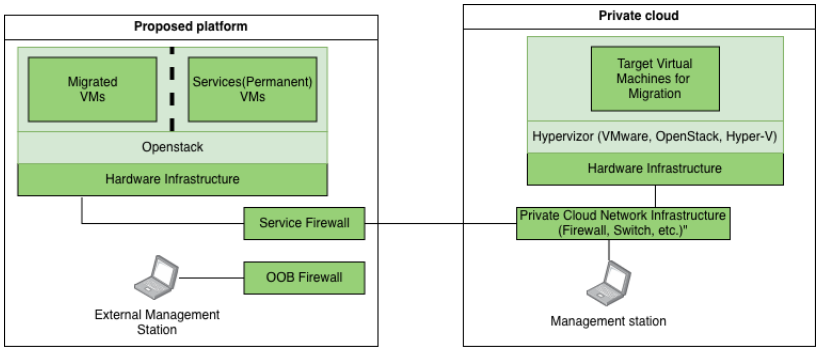

Figure 5. Connecting to the source/destination datacenter for VM migration

Following the completion of the VM replication phase, the platform is physically transported to the destination site, where an operator executes the same steps but in reverse order to transfer the VMs to the destination datacenter. These capabilities can be valuable for private cloud providers to support their customers in migrating workloads from on-premises infrastructure to the cloud.

5.3. Workload execution

This scenario investigates the sustained execution of computational workloads in the context of business continuity and edge computing paradigms, where the reliance on centralized cloud infrastructure is either impractical or unavailable. The platform can be quickly transported and deployed at the required location to run the necessary workloads, with the only requirement being supplying sufficient electrical power.

This scenario builds on Scenario 2. The destination is the platform itself in Coriolis' orchestration stage, which will replace the remote datacenter as the target environment for the replicated VM instances. An OpenStack private cloud platform is a proven production tool for effectively orchestrating and managing all processes involved with VM instances and making them accessible through a configurable virtual network layer. After the migration to OpenStack, the

platform can be relocated and function as an independent mobile datacenter. This scenario is particularly suitable for Disaster Recovery situations, allowing a company to have a rapid, field-deployable mini-datacenter to temporarily host critical services until full resource availability is restored. Additionally, this solution may be used as a remote, fully autonomous solution that can be rapidly deployed in remote locations where services and computing power are needed.

6. Discussion

The proposed architectural design presents several advantages over existing solutions:

- *Flexibility*. Each architectural layer concentrates on a specific capability and can evolve independently of the others;
- *Scalability*. The modular composition of each layer allows adjustment of computing, storage, and network resources to the requirements of each operational context;
- *Operational self-sufficiency*. The platform integrates within a single self-contained system the full set of services discussed in earlier chapters — virtualization, multi-protocol storage, networking, and orchestration. There is no need for reliance on external infrastructure during deployment.
- *Enhanced data security*. Achieved through network isolation from public infrastructures and data-at-rest encryption.

The proposed design is readily extensible: additional services may be incorporated into the service layer in the form of VMs, exposed either through the firewall or directly via the Ethernet switch. By design, the layered architecture does not enforce a single configuration; rather, it provides the structural flexibility to support multiple deployment configurations. As an illustration, the architecture accommodates both a dedicated out-of-band (OOB) firewall for management traffic and a consolidated single-firewall configuration, in which segmentation and security policies are unified within a single device.

Scalability constitutes another key strength of the proposed solution. Depending on the operational context, the architecture can also be implemented using a single transit case with a minimal set of devices or extended to as many cases as needed to provide considerably greater storage and compute capacity. Figure 6 presents two representative hardware configurations.

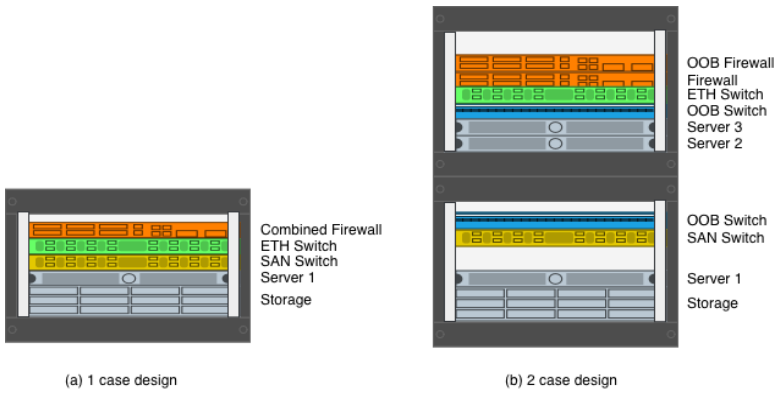

Figure 6. Proposed physical designs: (a) one case and (b) two cases

We consider the strongest contribution of our design to lie in the breadth of operational scenarios it can support within a single, coherent, scalable architecture, ranging from data migration and VM transfer or conversion to the execution of complete workloads in business continuity, mobile datacenter, and edge computing contexts. Existing solutions usually address subsets of this range: appliances for bulk data transfers to the cloud (e.g., AWS Snowball, Azure Data Box), mobile cloud platforms tied to specific ecosystems (e.g., Dell EMC Tactical Azure Stack Hub), or specialized edge computing appliances. The proposed design is distinctive in integrating these capabilities in a single, vendor-neutral platform. This positioning is summarized in Table 1.

Table 1. Comparison of existing solutions and the proposed design

Capability	**AWS Snowball Edge**	**Azure Data Box/Next gen**	**Google Transfer Appliance**	**Dell EMC Tactical Azure Stack Hub**	**Proposed platform**
Data Storage	42 / 80 / 210 TB	80/770TB, 120/525TB	7 / 40 / 300 TB	Configurable all-flash	Configurable all-flash
Storage Protocols	S3, NFS, EBS-compatible	SMB, NFS, REST(HTTP)	NFS, SMB, SCP, SFTP	SMB, iSCSI, Azure Blob	NFS, SMB, FC, FTP, SCP, S3
VM Migration	Limited (AMI/EC2 import)	Only VM backup, using external solution	No	Yes (Azure VMs)	Yes
VM Conversion	Limited (AMI ↔ EC2)	No	No	Limited (VHD ↔ Azure)	Yes
VM Execution	EC2 compatible	No	No	Yes (Hyper-V)	Yes
Vendor Neutral	No (AWS lock-in)	No (Azure lock-in)	No (GCP lock-in)	No (Microsoft Azure + Dell)	Yes

7. Conclusion and future work

This paper presents the design of a mobile computing platform intended to support the migration and storage of large volumes of data across heterogeneous private infrastructures. The proposed platform enables local execution and format conversion of VM workloads, providing a self-contained environment for hosting virtualized services. The primary contribution of this work is the design of a system that addresses a compound gap identified in the related work: the absence of a solution that integrates VM migration, local VM execution, automated format conversion, integrated data storage, and a field-deployable form factor within a single system.

The applicability of this design has been illustrated through a set of representative operational scenarios, ranging from the bidirectional migration of data and workloads between geographically dispersed sites to the local execution of complete virtualized environments under restricted or absent external connectivity.

Several directions remain open for future development. The implementation of an automation tool for the reconfiguration and decommissioning of the platform across operational contexts would substantially reduce the effort required to repurpose the system between deployments. A second direction concerns the validation of architectural choices through a suite of benchmarks that measure data transfer speeds, VM migration times, conversion overheads, and recovery and deployment times under conditions representative of the targeted scenarios.

Author contributions

Conceptualization: A.D., I.C., I.P., B.I., F.P.; Data Curation: A.D., I.C., I.P., B.I.; Project administration: B.I.; Supervision: B.I., F.P.; Validation: A.D., I.C., I.P., B.I., F.P.; Writing original draft: A.D., I.C., I.P., B.I.; Writing—review and editing: A.D., I.C., I.P., B.I., F.P.; All authors have read and agreed to the published version of the manuscript.

Acknowledgments

This work was supported by a grant of the Ministry of Research, Innovation and Digitization, CCCDI - UEFISCDI, project number PN-IV-P6-6.3-SOL-2024-0068, within PNCDI IV.

Submission: 11 May 2026; Revised: 26 May 2026; Accepted: 31 May 2026; Published: 30 June 2026.

REFERENCES

Cheng, C., Deng, Z., Gu, Z. & Xu, D. (2016) vMocity: Traveling VMs across heterogeneous clouds. *2016 IEEE 35th Symposium on Reliable Distributed Systems (SRDS)*. pp. 101–110. https://doi.org/10.1109/SRDS.2016.022.

He, T., Toosi, A. N. & Buyya, R. (2021) SLA-aware multiple migration planning and scheduling in SDN-NFV-enabled clouds. *Journal of Systems and Software*. 176, 110943. https://doi.org/10.1016/j.jss.2021.110943.

Hu, W., Hicks, A., Zhang, L., Dow, E. M., Soni, V., Jiang, H., Bull, R. & Matthews, J. N. (2013) A quantitative study of virtual machine live migration. *Proceedings of the 2013 ACM Cloud and Autonomic Computing Conference*. pp. 1–10. https://doi.org/10.1145/2494621.2494622.

Jamshidi, P., Ahmad, A. & Pahl, C. (2013) Cloud migration research: A systematic review. *IEEE Transactions on Cloud Computing*. 1(2), 142–157. https://doi.org/10.1109/TCC.2013.10.

Kargatzis, D., Sotiriadis, S. & Petrakis, E. G. M. (2017) Virtual Machine migration in heterogeneous clouds: From openstack to VMWare. *2017 IEEE 38th Sarnoff Symposium*. pp. 1–6. https://doi.org/10.1109/SARNOF.2017.8080393.

Kaur, M. (2025) VM migration strategies: A review of approaches and challenges. *International Journal of Innovative Research in Technology*. 11(10), 7411–7421. https://ijirt.org/IJIRT | An UGC Compliant Peer reviewed Journal | International Open Access Journal

Laoutaris, N., Sirivianos, M., Yang, X. & Rodriguez, P. (2011) Inter-datacenter bulk transfers with netstitcher. *Proceedings of the ACM SIGCOMM 2011 Conference*. pp. 74–85. https://doi.org/10.1145/2018436.2018446.

Lin, X., Sun, W., Veeraraghavan, M. & Hu, W. (2016) Time-shifted multilayer graph: A routing framework for bulk data transfer in optical circuit-switched networks with assistive storage. *Journal of Optical Communications and Networking*. 8(3), 162. https://doi.org/10.1364/JOCN.8.000162.

Luo, L., Yu, H., Foerster, K.-T., Noormohammadpour, M. & Schmid, S. (2020) Inter-datacenter bulk transfers: Trends and challenges. *IEEE Network*. 34(5), 240–246. https://doi.org/10.1109/MNET.011.1900632.

Majigi, M. U., Idris, I., Abdulhamid, S. M. & Ikuesan, R. A. (2025) Big data transfer service architecture for cloud data centers: Problems, methods, applications, and future trends. *Discover Computing*. 28(1), 163. https://doi.org/10.1007/s10791-025-09682-3.

Mell, P. M. & Grance, T. (2011) The NIST definition of cloud computing (NIST SP 800-145; p. NIST SP 800-145). *National Institute of Standards and Technology*. https://doi.org/10.6028/NIST.SP.800-145.

Moravcik, M., Segec, P., Kontsek, M. & Zidekova, L. (2024) Model-driven approach to cloud-portability issue. *Applied Sciences*. 14(20), 9298. https://doi.org/10.3390/app14209298.

Noormohammadpour, M., Raghavendra, C. S., Kandula, S. & Rao, S. (2018) QuickCast: fast and efficient inter-datacenter transfers using forwarding tree cohorts. *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications* pp. 225–233. https://doi.org/10.1109/INFOCOM.2018.8486324.

Noormohammadpour, M., Raghavendra, C. S., Rao, S. & Kandula, S. (2017) *DCCast: Efficient Point to Multipoint Transfers Across Datacenters*. https://doi.org/10.48550/ARXIV.1707.02096.

Pant, A. (2025) A comparison of AWS, Google Cloud, and Azure services perceptivity for cloud data engineering. In Dr. V. Shrivastava, Dr. A. Pandey, Dr. V. Pathak & Er. R. B. Buri, *IIP Series Iterative International Publishers (IIP)*. Vol. 6, 66–74. https://doi.org/10.58532/nbennurFTHPSW9N1.

Satyanarayanan, M., Bahl, P., Caceres, R. & Davies, N. (2009). The case for VM-based cloudlets in mobile computing. *IEEE Pervasive Computing*. 8(4), 14–23. https://doi.org/10.1109/MPRV.2009.82.

Shen, S., Van Beek, V. & Iosup, A. (2015) Statistical characterization of business-critical workloads hosted in cloud datacenters. *2015 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*. pp. 465–474. https://doi.org/10.1109/CCGrid.2015.60.

Singhai, V., Damania, K., Holmukhe, S. & Bhavathankar, P. (2018) A Distributed API for live VM migration in cloudlets. *International Journal of Computer Applications*. 180(52), 12–18. https://doi.org/10.5120/ijca2018917357.

Stoleriu, R., Petre, I. & Pop, F. (2025) Cybersecurity Governance in Large-scale Infrastructures. *Romanian Journal of Information Technology and Automatic Control*, ISSN 1220-1758, vol. 35(1), pp. 51-66, 2025. https://doi.org/10.33436/v35i1y202504

Strauch, S., Andrikopoulos, V., Karastoyanova, D., Leymann, F., Nachev, N. & Stäbler, A. (2014) Migrating enterprise applications to the cloud: Methodology and evaluation. *International Journal of Big Data Intelligence*. 1(3), 127. https://doi.org/10.1504/IJBDI.2014.066319.

Vishwanath, K. V., Greenberg, A. & Reed, D. A. (2009) Modular data centers: How to design them? *Proceedings of the 1st ACM Workshop on Large-Scale System and Application Performance*, pp. 3–10. https://doi.org/10.1145/1552272.1552275.

Wang, Y., Su, S., Liu, A. X. & Zhang, Z. (2014) Multiple bulk data transfers scheduling among datacenters. *Computer Networks*. 68, 123–137. https://doi.org/10.1016/j.comnet.2014.02.017.

Zhang, F., Liu, G., Fu, X., & Yahyapour, R. (2018) A survey on virtual machine migration: Challenges, techniques, and open issues. IEEE Communications Surveys & Tutorials, 20(2), 1206–1243. https://doi.org/10.1109/COMST.2018.2794881

* * *

Dan AVRAM is a specialist within the Center of Excellence for Advanced Cyber Security Technologies (CETASC), Military Technical Academy. He is currently pursuing a Ph.D. at the National University of Science and Technology Politehnica Bucharest. His research interests focus on cloud computing, resource management and energy efficiency.

Dan AVRAM lucrează ca specialist în cadrul CETASC din Academia Tehnică Militară. În prezent, urmărește studii doctorale la Universitatea Națională de Știință și Tehnologie Politehnica București. Interesele sale de cercetare se concentrează pe cloud computing, gestionarea resurselor și eficiența energetică.

* * *

Ioana MATEI completed her doctoral studies in Computer and Information Technology. She is currently a specialist at CETASC, Military Technical Academy "Ferdinand I", Bucharest. Her research interests focus on cybersecurity and Internet of Things (IoT) security.

Ioana MATEI a finalizat studiile doctorale în domeniul Calculatoare și Tehnologia Informației. În prezent, activează ca specialist în cadrul CETASC, Academia Tehnică Militară „Ferdinand I" din București. Activitatea sa de cercetare este orientată către securitatea cibernetică și securitatea IoT.

* * *

Ioana APOSTOL holds a Ph.D. degree in Computer and Information Technology and is currently Head of the "Computer Network Security" Laboratory at CETASC, within the Military Technical Academy "Ferdinand I". Her research interests focus on cybersecurity and machine learning for network security.

Ioana APOSTOL deține titlul de doctor în domeniul Calculatoare și Tehnologia Informației și este în prezent șef al laboratorului „Securitatea Rețelelor de Calculatoare” din cadrul CETASC, în Academia Tehnică Militară „Ferdinand I”. Domeniile sale de interes în cercetare includ securitatea cibernetică și utilizarea tehnicilor de învățare automată pentru securitatea rețelelor.

* * *

Ion BICA is a full professor at the Military Technical Academy “Ferdinand I”, Romania. His main research interests are applied cryptography and cybersecurity. He has published six books and more than 100 papers in journals and conference proceedings, co-edited three conference volumes published by Springer, and delivered invited talks at numerous universities and international conferences. He is an active member in several NATO and EDA working groups on cybersecurity.

Ion BICA este profesor titular la Academia Tehnică Militară „Ferdinand I” din România. Principalele sale domenii de cercetare sunt criptografia aplicată și securitatea cibernetică. A publicat 6 cărți și peste 100 de lucrări în reviste și lucrări de conferințe, a fost editor la 3 volume de conferințe publicate de Springer și a susținut prezentări la numeroase universități și conferințe internaționale. Este membru activ în mai multe grupuri de lucru NATO și EDA privind securitatea cibernetică.

* * *

Florin POP is a Full Professor at the Computer Science and Engineering Department of National University of Science and Technology Politehnica Bucharest, Romania. His main research interests are in large-scale distributed systems, adaptive and autonomous methods, optimization methods, and applications in Big Data, IoT and Smart Cities. He is a Senior Researcher at ICI Bucharest, a member of the Academy of Romanian Scientists and a Senior Member of IEEE (Institute of Electrical and Electronics Engineers).

Florin POP este profesor titular la Departamentul de Informatică și Inginerie al Universității Naționale de Știință și Tehnologie Politehnica București, România. Principalele sale interese de cercetare sunt în sisteme distribuite la scară largă, metode adaptive și autonome, metode de optimizare, aplicații Big Data, IoT și Smart Cities. Florin Pop este cercetător științific gradul I în cadrul Institutului Național de Cercetare-Dezvoltare – ICI București, membru al Academiei Oamenilor de Știință din România și membru senior al IEEE (Institutul Inginerilor Electrotehnicieni și Electroniști).