frontiers in Complex Systems

TYPE Brief Research Report
PUBLISHED 17 February 2026
DOI 10.3389/fcpxs.2026.1724679

OPEN ACCESS

EDITED BY
Chandra Krintz,
University of California, Santa Barbara,
United States

REVIEWED BY
Bakkiyaraj Kanthimathi Malamuthu,
Morgan Stanley Financial Services,
United States
Seetharami Seelam,
IBM Research - Yorktown Heights,
United States

*CORRESPONDENCE
Germán Moltó,
gmolto@dsic.upv.es

RECEIVED 14 October 2025
REVISED 29 January 2026
ACCEPTED 06 February 2026
PUBLISHED 17 February 2026

CITATION
Moltó G, Caballer M, Parcero E and Rodríguez V (2026) Deployment of transatlantic computational testbeds via the infrastructure manager.
Front. Complex Syst. 4:1724679.
doi: 10.3389/fcpxs.2026.1724679

Deployment of transatlantic computational testbeds via the infrastructure manager

Germán Moltó*, Miguel Caballer, Estíbaliz Parcero and Vicente Rodríguez

Instituto de Instrumentación Para Imagen Molecular (I3M), Centro Mixto CSIC - Universitat Politècnica de València, Valencia, Spain

Transatlantic scientific collaborations require computational testbeds that can be provisioned on demand and reconfigured rapidly while spanning institutions in different regulatory and operational domains. During the DISCOVER-US exchange program, we integrated the Infrastructure Manager (IM), a TOSCA-based orchestrator for the computing continuum, with the Chameleon cloud infrastructure. The workflow combined federated identity management, delegated project administration, and an IM plugin that targets Chameleon's OpenStack endpoints through application credentials. We validated the approach by deploying single virtual machines, a production-ready Galaxy environment, distributed OSCAR-based serverless clusters that offload an AI-based fish detection pipeline, a workflow for flood impact modeling, and a hybrid SLURM cluster. Transatlantic computational testbeds included dynamically provisioned computational resources from EGI Federated Cloud and Chameleon. The study also documents operational constraints encountered with lease automation, bare-metal introspection, and the exposure of Kubernetes services across wide-area networks. The resulting blueprint demonstrates a reproducible path to deploy secure, elastic, and scientifically useful transatlantic computational testbeds.

KEYWORDS
cloud computing, cloud orchestration, computational testbeds, infrastructure as code, serverless computing

1 Introduction

The DISCOVER-US program promotes transatlantic collaborations in distributed computing and swarm intelligence by funding research exchanges that pursue, among many other topics, shared testbed developments (DISCOVER-US Consortium, 2025) to foster scientific experimentation. Establishing these collaborations demands more than coordinated travel and project planning. Teams must instantiate well-governed computing environments that bridge regulatory domains while guaranteeing reproducibility and accountability. Achieving that goal requires orchestrating virtualized and bare-metal resources that are subject to distinct identity providers, security postures, and operational policies.

Chameleon[1] (Keahey et al., 2020) operates a nationally funded, reconfigurable platform based on OpenStack that exposes both virtualized and bare-metal resources for computer

1 Chameleon - https://chameleoncloud.org

science at scale. Its emphasis on configurability and experiment repeatability has made it a reference platform for systems research in the US. At the same time, Chameleon's policies, resource reservation model, and authentication flows differ from those of European e-infrastructures such as the EGI Federated Cloud[2]. Integrating it into existing European orchestration pipelines demands dedicated engineering around credential brokerage, quota management, and the orchestrated deployment and configuration of resources across infrastructures.

This article presents the integration work undertaken during the April–May 2025 DISCOVER-US research exchange from Prof. Germán Moltó to Argonne National Laboratory. Building on preliminary bilateral discussions, we executed a hands-on program that paired IM developers in Europe with Chameleon operators in the US. The collaboration produced a reusable provider plugin, tested a spectrum of scientific workloads, and documented the procedures required to deploy joint transatlantic computational testbeds.

We make three contributions reflected in this paper. First, we describe an end-to-end identity and access management workflow that leverages federated login, delegated project administration, and application credentials to comply with policies from EGI Federated Cloud and Chameleon. Second, we extend the IM code base with a Chameleon-specific plugin that discovers resources, provisions and configures virtual machines, and manages their lifecycle. Third, we validate the resulting integration by deploying five demonstrator environments ranging from single-node virtual machines to serverless AI (Artificial Intelligence) inference pipelines and hybrid high-performance computing (HPC) virtual clusters. Together, these contributions deliver a blueprint to deploy elastic, secure, and scientifically productive transatlantic computational testbeds.

2 Background and related work

2.1 Federated testbeds for distributed computing

International collaborations often rely on federated testbeds that amalgamate resources from multiple administrative domains. European initiatives such as the EGI Federated Cloud have refined mechanisms for identity management, accounting, and Virtual Organization (VO)-based governance, while US programs, including Chameleon and Fabric[3], focus on deeply reconfigurable infrastructure. Harmonizing those approaches remains a challenge because European policies rely heavily on community-managed identity providers, whereas U.S. research clouds frequently require institution-backed project leadership. The DISCOVER-US vision addresses this gap by funding short exchanges that prototype shared operational models, with a particular focus on the computing continuum that spans cloud, edge, and specialized devices (DISCOVER-US Consortium, 2025). Our work extends that agenda by demonstrating how a mature European Cloud orchestrator can consume a prominent US research cloud infrastructure without diluting either side's policy requirements.

2.2 Infrastructure Manager in the computing continuum

IM[4] is an open-source Cloud orchestrator that provides TOSCA-compliant orchestration capabilities across virtualized data centers, public clouds, and edge devices (Caballer et al., 2023). There exist other orchestration frameworks, such as Terraform[5] and Cloudify[6], that offer similar capabilities. However, IM's design emphasizes multi-cloud portability, adoption of standards (TOSCA) and integration with contextualization engines popular in research communities (e.g., Ansible), apart from supporting multiple interfaces (API, CLI and a web-based dashboard) and being completely open-source.

Indeed, IM supports declarative infrastructure blueprints and a vault for managing sensitive credentials. It has been adopted in projects targeting scientific gateways, digital twins, and large-scale data processing. The platform's modular architecture encourages the development of provider-specific drivers that translate generic lifecycle events into concrete API calls. Prior integrations have covered OpenStack, OpenNebula, Kubernetes, AWS, and edge orchestrators. Extending IM to Chameleon, therefore, required capturing the particularities of Chameleon's Keystone endpoints, flavour catalogue, bare-metal leasing system, and monitoring stack while preserving IM's abstractions for node states and orchestration workflows.

2.3 Domain workloads motivating the integration

The demonstrators selected for this work reflect recurring demands from European and US collaborators. Bioinformatics teams frequently rely on the Galaxy[7] platform to expose reproducible analysis pipelines. Environmental scientists are building digital twins, such as the DT-Flood workflow, that orchestrate hydrodynamic models across geographically distributed resources (Tromp et al., 2024). Research into serverless computing has produced frameworks like OSCAR that enable highly parallel event-driven data processing (Perez et al., 2019; Risco et al., 2021). The OBSEA submarine observatory[8] and its AI-based fish detection service[9] can benefit from bursting between European and US resources to meet demand. Finally, hybrid HPC

2 EGI - https://www.egi.eu

3 Fabric - https://portal.fabric-testbed.net

4 Infrastructure Manager (IM) - https://www.grycap.upv.es/im

5 Terraform - https://www.terraform.io

6 Cloudify - https://cloudify.co

7 Galaxy Project - https://usegalaxy.org/

8 OBSEA Underwater Observatory - https://www.obsea.es

9 OBSEA Fish Detection Service - https://dashboard.cloud.imagine-ai.eu/marketplace/modules/obsea-fish-detection

experiments combine SLURM clusters across multiple clouds to allocate workloads near data sources or specialized accelerators (Caballer et al., 2021). By demonstrating these workloads on a joint EGI–Chameleon testbed, we cover a representative spectrum of data-intensive and compute-intensive scenarios.

2.4 Operational challenges in transatlantic orchestration

Operating across continents introduces latency, regulatory, and cultural considerations that are frequently absent from single-region testbeds. Indeed, European projects typically rely on community-driven VOs, whereas Chameleon foregrounds project leadership and allocation accountability. Aligning those governance models demands tooling that can articulate provisioning, enforcing least privilege, and providing descriptive deployment logs. Our integration work therefore emphasized not only API compatibility, but also operational artifacts that lower the barrier for future collaborations to navigate these challenges. This is exemplified by contributing upstream the integration with Chameleon in the production instance of the IM[10] offered under the EGI catalog of services.

3 Methodology

At a high level, our methodology consisted of four main steps: (1) users gather temporary, scoped credentials to access the underlying Cloud platforms (in our case EGI Federated Cloud and Chameleon); (2) extending the Infrastructure Manager with a provider plugin that encapsulates Chameleon's OpenStack endpoints and resource models; (3) developing automation artifacts that enable researchers to deploy complex workloads with minimal manual intervention; and (4) evaluating the integration through a series of demonstrator deployments.

3.1 Credential brokering and project onboarding

We adopted a staged credential workflow that couples Chameleon's federated login with the EGI Check-in[11] service. Visiting researchers authenticated through Chameleon's identity federation and requested Principal Investigator status from the Chameleon operations team. Once approved, the PI created a dedicated project, configured Service Unit quotas, and invited collaborators with role-based access assignments. Each collaborator generated OpenStack application credentials restricted to the project scope. These credentials were stored in the IM Vault service and exposed to automation. The arrangement delivered traceability, revocation capabilities, and a clear mapping between human identities and automated actions, satisfying the accountability requirements of both infrastructures.

Note that these credentials can be created with an expiration time and be easily revoked by the user at any time via the OpenStack portal and can have restricted roles assigned to them, following the principle of least privilege. Also, no trust assumptions are made between EGI and Chameleon regarding identity management, as each side relies on its own identity federation. It is the user who is in charge of bridging both sides by obtaining the necessary credentials from each side and configuring them in the IM. This approach preserves the independence of each infrastructure while enabling seamless cross-cloud operations.

3.2 Infrastructure Manager integration with Chameleon

Integrating IM with Chameleon required extending the former with a plugin that encapsulates the Keystone endpoints and region labels exposed by Chameleon sites (e.g., the KVM@TACC site, operated by the Texas Advanced Computing Center). The plugin authenticates via application credentials, fetches flavour and image metadata, and maps IM's state transitions of Pending, Running, Configuring, and Configured to Nova's provisioning lifecycle. Special handling was introduced for resources that require reservation identifiers, such as bare-metal nodes leased through Chameleon's Blazar service. We also enhanced IM's error-handling routines to surface Chameleon's verbose fault messages, enabling researchers to diagnose quota breaches, missing allocations, or hypervisor-side issues. The integration work followed a test-driven approach. We first mapped IM's existing OpenStack driver capabilities and identified deltas required to support Chameleon's configuration, including region-specific endpoints and project-scoped quotas.

The provider plugin adheres to IM's modular design, where each driver implements a minimal interface for authentication, resource discovery, provisioning, and lifecycle management. We decomposed the plugin into components responsible for Keystone interactions, Nova and Glance queries, networking, and optional Blazar reservations. Each component exposes clear error semantics so that failures propagate with actionable diagnostics. Caching mechanisms minimize redundant API calls by storing flavour and image metadata for the duration of a deployment. We instrumented the plugin with structured logging that captures request identifiers, execution timings, and retry counts. The architecture leaves room for future extensions, such as support for bare-metal introspection data, without requiring disruptive refactoring. The resulting driver delivers parity with IM's EGI Federated Cloud OpenStack integrations while preserving the ability to evolve independently as Chameleon introduces new capabilities.

3.3 Operational workflow and automation

We implemented a repeatable automation workflow that allows researchers to progress from template selection to deployment verification with minimal manual intervention. Templates are version-controlled as TOSCA YAML files that describe compute nodes, network attachments, security groups, and contextualization scripts expressed via Ansible Roles. We contributed examples to a shared GitHub repository[12] that stores TOSCA templates and Ansible playbooks.

10 EGI's IM service - https://im.egi.eu

11 EGI Check-In - https://www.egi.eu/service/check-in/

12 TOSCA Templates GitHub repository - https://github.com/grycap/tosca

Developing reusable TOSCA templates demanded a disciplined approach to parameterization and dependency management. We adopted a layered structure in which core infrastructure components (networks, security groups, base images) are coupled with workload-specific details. Parameters capture zones and image identifiers, enabling the same artifact to target both European and Chameleon sites with minimal overrides. Configuration scripts are packaged as Ansible roles that are versioned independently to guarantee reproducibility. This artifact management discipline ensures that the investment made during the exchange continues to deliver value to subsequent projects.

Researchers select a template and edit parameter files to specify hardware, software, and configuration details. The IM CLI or the IM Dashboard is then used to submit the template for deployment. IM orchestrates the provisioning of resources on Chameleon, executes contextualization scripts, and monitors progress through its state machine. Upon completion, IM provides SSH access details and logs that document each step of the deployment. This workflow abstracts away the complexities of Chameleon's API while preserving flexibility for advanced users to customize deployments as needed.

3.4 Evaluation approach

Our evaluation combined qualitative observations with quantitative measurements collected during the exchange. For qualitative insights, we conducted structured debriefings after each deployment, capturing operator effort. Quantitative data included provisioning times recorded by IM, resource utilization metrics gathered from Chameleon's metrics, and network latency measurements. We also monitored failed authentication attempts to ensure that automation respected Chameleon's guardrails.

4 Demonstrator deployments

4.1 Virtualized baseline deployments

We began with baseline scenarios that exercise fundamental provisioning capabilities. Single-node virtual machines were deployed with custom images pre-configured for scientific Python stacks. IM's contextualization scripts configured software and synchronized SSH keys. Provisioning times averaged 4 minutes from template submission to SSH availability, aligning with IM's performance on European OpenStack sites. These baseline deployments surfaced minor differences in default security group policies and metadata naming conventions, which we addressed by updating the IM templates accordingly. The baselines now serve as smoke tests that verify credential validity and quota health before launching more complex infrastructures.

4.2 Galaxy bioinformatics platform

The Galaxy demonstrator showcases how IM deploys complex applications. During contextualization, Ansible roles configured Galaxy and imported a curated set of reference datasets. We benchmarked common bioinformatics pipelines by observing execution times comparable to European deployments when data resided locally. To mitigate transatlantic transfer overheads, we prototyped caching strategies with S3 buckets replicated across geographical regions. Indeed, Galaxy can serve as a shared scientific gateway, with IM handling lifecycle management and Chameleon providing elastic compute capacity.

4.3 Serverless fish detection across the atlantic

To evaluate event-driven workloads, we orchestrated two OSCAR[13] clusters, one in the EGI Federated Cloud (LIP site, in Portugal, EU) and one on Chameleon (KVM@TACC site, in Texas, US), as shown in Figure 1. Each cluster comprised Kubernetes nodes, an OSCAR control plane, and storage back-ends. IM coordinated the deployment by instantiating virtual machines, installing Kubernetes, configuring OSCAR, and loading service descriptors for the fish detection pipeline. The pipeline ingests underwater imagery, applies AI model inference, and publishes annotated outputs to shared storage based on MinIO. We configured workload delegation so that idle invocations on the European cluster burst to Chameleon during demand spikes. This demonstrated that federated serverless platforms can tolerate long-distance links when offloading job executions. The artifacts developed for this demonstrator are publicly available[14].

Two tests were performed to verify the latency and performance of the testbed. Initially, independent invocations of the fish-detector service were launched on the OSCAR cluster deployed in the US and the same number of independent invocations were submitted on the OSCAR cluster deployed in the EU. The images to be processed were on the order of 100 kB. The processing times for each service were 47 s on the OSCAR@US cluster and 74 s on the OSCAR@EU cluster. The difference in processing time was due to the superior computational resources of the OSCAR@US cluster. However, the download times for the event (the jpg file to be processed) on each cluster were around 120 ms. This is because the events were located on the local MinIO of each OSCAR cluster.

Then, a test was launched in which 7 invocations of the fish-detector service were performed on the OSCAR@EU cluster. Since it could not process all the jobs simultaneously, it delegated 3 to the OSCAR@US cluster. Job execution times on the OSCAR@EU cluster are similar to the first test because the images to be processed were the same (around 100 kB). However, on the OSCAR@US cluster, the execution time averaged 49 s. This is because the event now needs to be downloaded from the OSCAR@EU MinIO server instead of the local server, as in the previous test. Therefore, the communication between OSCAR@US and the MinIO where the event (the JPG file to be processed) is responsible for this slight increase in processing time on the OSCAR@US cluster.

4.4 Digital twin flood modeling workflow

We integrated the DT-Flood workflow developed within the interTwin[15] project, using Common Workflow Language (CWL)

13 OSCAR - https://oscar.grycap.net

14 OSCAR Fish Detector Replicas - https://github.com/grycap/oscar/tree/master/examples/fish-detector-replicas

15 interTwin – https://www.intertwin.eu

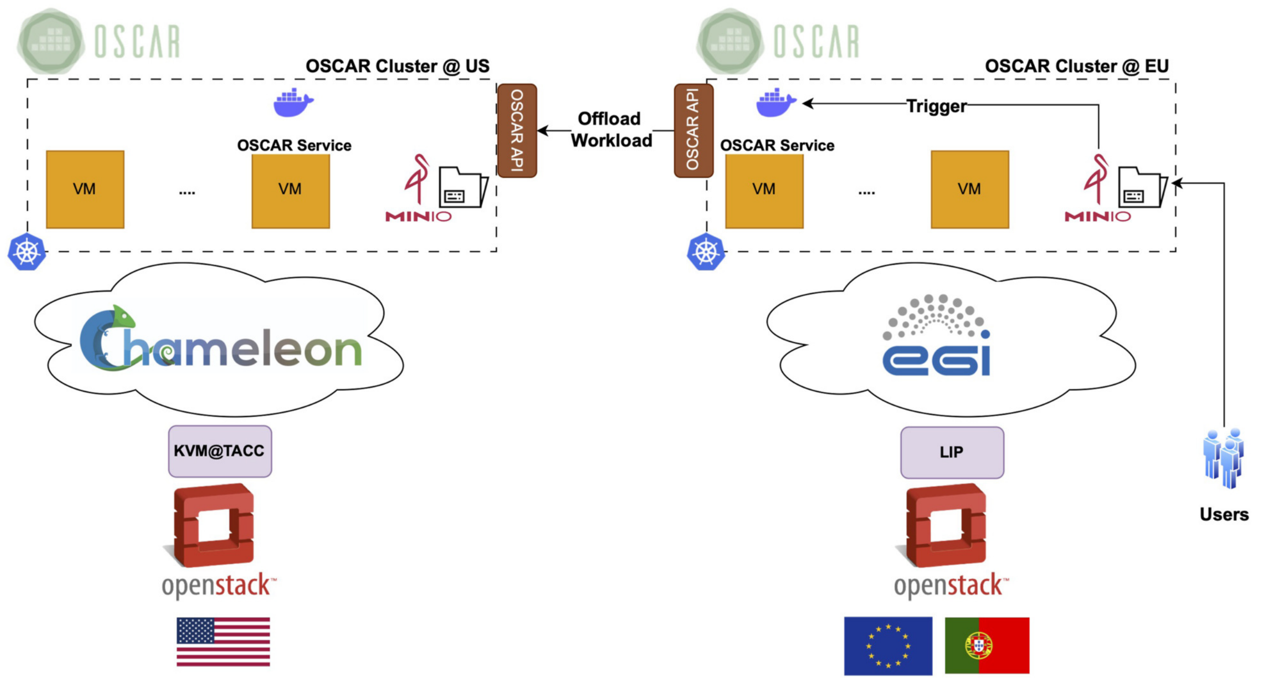

FIGURE 1
Deployment of OSCAR clusters as a transatlantic computational testbed for workload offloading.

descriptions to distribute model steps across the two OSCAR clusters (Risco et al., 2024). DT-Flood, part of the FloodAdapt Digital Twin by Deltares, combines three numerical models—WFLOW for river discharge, SFINCS for flood-extent mapping, and RA2CE for assessing disruptions to infrastructure networks—making it a representative multi-stage workflow for evaluating distributed execution. In this setup, also described in the interTwin success story[16], we deployed OSCAR clusters on the EGI Federated Cloud in Europe and on Chameleon in the USA to form a transatlantic computational testbed. IM handled the orchestration of the deployment and configuration of the OSCAR clusters across both sites, enabling us to exploit resource availability across regions, maintain data-locality constraints, and increase resilience by allowing workflow steps to run on either cluster when needed. Hydrological preprocessing and the WFLOW and SFINCS simulations were executed on the European cluster to remain close to regional datasets, while the RA2CE impact-analysis stage was offloaded to the US cluster to leverage additional capacity. The CWL definition indicated the script to be executed for each step and the target execution site. A customized Python script, invoked by the CWL workflow, used the `oscar-python` library to perform all necessary operations within the OSCAR environment, including data staging, triggering model execution, monitoring progress, and retrieving outputs across both clusters. As shown in Figure 2, this approach enabled seamless, distributed execution of the DT-Flood workflow across the transatlantic testbed.

16 OSCAR and DT-Flood in the DISCOVER-US project - https://www.intertwin.eu/case-study/success-story-transatlantic-computational-testbeds-enabling-seamless-workflow-executions-of-flood-simulations

4.5 Hybrid SLURM cluster

We provisioned a hybrid SLURM cluster that placed the head node and one worker on the EGI Federated Cloud while scaling additional workers on Chameleon, as an example of cross-cloud deployments as initially introduced in (Caballer et al., 2021). The TOSCA template described the network connectivity, NFS exports, and SLURM daemon registration across both sites. Batch jobs submitted via the front-end node were seamlessly executed on resources from both continents. Performance analysis revealed that compute-bound tasks scaled nearly linearly, whereas I/O-heavy workloads experienced throughput degradation because the shared filesystem traversed the transatlantic link.

5 Discussion

The integration of Chameleon into IM's orchestration workflow shortens the time required to build transatlantic computational testbeds from days to hours. Automating credential management and lifecycle operations reduces the operational overhead borne by Cloud infrastructures. The demonstrators cover virtual machines, platform services, serverless pipelines, and hybrid clusters, revealing that a single orchestration framework can meet heterogeneous collaboration needs. Importantly, the collaboration established shared efforts that survived the conclusion of the exchange, enabling subsequent remote cooperation.

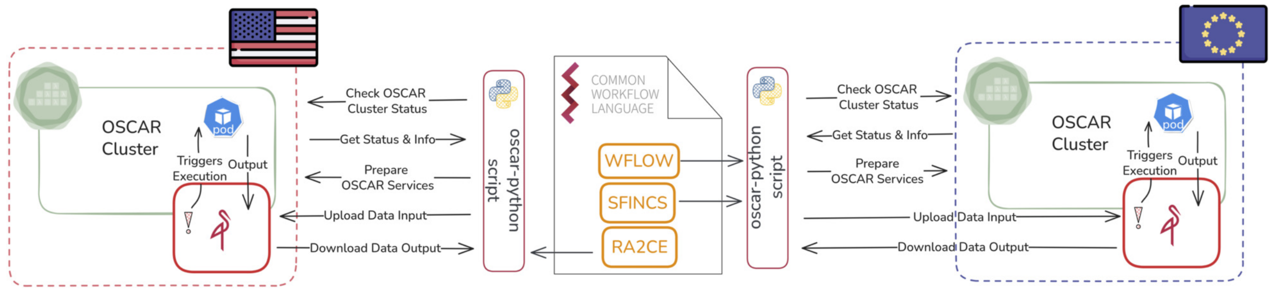

FIGURE 2
Overview of the workflow for executing the tasks of the DT-Flood models on OSCAR clusters across Europe and the USA.

TABLE 1 **Deployment times for OSCAR and Galaxy at two sites.**

Site	OSCAR	Galaxy
KVM@TACC	12:07	13:37
EGI LIP site	39:35	39:26

Nevertheless, the deployment surfaced limitations. Creating reservation leases on behalf of end users still requires manual intervention via the Chameleon helpdesk, delaying access to specialized hardware. Bare-metal deployments do not yet expose resource metadata through the OpenStack flavour catalog, hindering IM's ability to tailor contextualization scripts automatically. Kubernetes clusters that span continents struggle with NodePort-based service exposure, causing liveness probes to fail under high latency. Addressing these issues will require deeper integration with reservation APIs, enhanced discovery mechanisms for bare-metal nodes, and the adoption of overlay networking solutions better suited to high-latency links.

Regarding deployment times, two of the described demonstrators have been measured: the OSCAR cluster and the Galaxy Platform, both on Chameleon and at the EGI LIP site. In the case of the OSCAR cluster, this implies the installation and configuration of a Kubernetes cluster with one front-end and one worker node, both with 4 CPUs and 8 GB of RAM. In the case of the Galaxy portal, a single VM is deployed with 2 CPUs and 4 GB of RAM.

As shown in Table 1, there is a significant difference in deployment time between the two sites. After analysis, an important difference was detected in disk access performance: the Chameleon site reported write access 3 times faster and read access 4 times faster than the EGI LIP site.

We also observed cultural and organizational challenges. European teams are accustomed to VO-based governance structures, whereas Chameleon emphasizes PI-led allocations. Bridging those expectations demanded clear communication about responsibilities, escalation paths, and acceptable use.

Another insight concerns reproducibility and measurement practices. The availability of version-controlled templates and automated validation scripts enabled us to recreate complex deployments weeks after the exchange concluded. We recommend that future collaborations adopt immutable artifact references. These practices not only facilitate scientific reproducibility but also accelerate operational recovery when regressions occur.

Looking ahead, we envision extending the integration to additional U.S. sites, including facilities that expose specialized accelerators or edge devices (e.g., CHI@EDGE). We also aim to explore cost-aware orchestration strategies that consider credit budgets alongside performance and policy constraints. These avenues will further mature the transatlantic testbed blueprint and support an expanding community of researchers.

To maximize reuse, we packaged templates, scripts, and best-practice guides as a publicly accessible collection. Artifacts are licensed under permissive terms and accompanied by metadata describing dependencies and responsible maintainers. An example is the entry in Chameleon's Trovi artifact repository[17]. We encourage other teams to adapt and extend these resources, fostering a virtuous cycle of shared learning.

6 Conclusion

We reported on the integration of the Infrastructure Manager with Chameleon Cloud to support the deployment of transatlantic computational testbeds for scientific collaborations. The joint effort delivered support in the Infrastructure Manager for a new provider plugin that encapsulates Chameleon's operational specifics, and a portfolio of demonstrator deployments spanning bioinformatics, serverless computing, digital twins, and hybrid HPC. Operational evaluation confirmed that the combined toolchain delivers reproducible deployments and manageable overheads. While challenges remain around bare-metal leasing, high-latency service exposure, and cross-institutional governance, the work establishes a solid foundation for future joint experimentation. By sharing artifacts, documentation, and lessons learned, we invite other teams to replicate and extend the integration, accelerating the creation of secure and elastic transatlantic testbeds.

Note that this integration illustrates how research infrastructures from different continents can interoperate without requiring trust relationships between their identity management systems. This

17 Chameleon Trovi entry - https://www.chameleoncloud.org/experiment/share/ea0b7d1e-cacc-4f9d-8660-6f1827fdef92

required complex systems engineering that involve the coordinated orchestration and deployment of resources across distinct regulatory and operational domains, without any changes to the underlying infrastructures. This approach preserves the independence of each infrastructure while enabling seamless cross-cloud operations, which is a key requirement for transnational scientific collaborations. The wide range of TOSCA-based templates available for deployment demonstrates the versatility of the Infrastructure Manager in supporting diverse scientific workloads across continents, including cross-disciplinary complex topics that can benefit from highly distributed computing platforms.

Future work will concentrate on formalizing service-level objectives for cross-site deployments. We also plan to extend our cooperation to support models that recognize the operational burden of maintaining shared computational testbeds. Such models may include dedicated helpdesk pathways, pooled allocation budgets, or joint fellowships that onboard scientists across institutions. Progress on these fronts will ensure that the IM–Chameleon integration evolves from a successful exchange experiment into a durable pillar of transatlantic deployable computational testbeds.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

GM: Investigation, Methodology, Writing – review and editing, Funding acquisition, Conceptualization, Writing – original draft, Resources. MC: Writing – original draft, Software, Conceptualization, Writing – review and editing. EP: Validation, Writing – review and editing, Writing – original draft. VR: Validation, Writing – original draft, Writing – review and editing.

Funding

The author(s) declared that financial support was received for this work and/or its publication. The research exchange was funded by the DISCOVER-US project, financed by the European Union's Horizon Europe research and innovation funding programme under grant agreement number 101135064. Also, Grant PID2024-155662OB-I00 funded by MICIU/AEI/10.13039/501100011033. The results presented were obtained using the Chameleon testbed supported by the National Science Foundation.

Acknowledgements

We thank the Chameleon Cloud operations team at Argonne National Laboratory and the University of Chicago for their guidance during the exchange, and the EGI Federated Cloud for providing computational resources. We also acknowledge the interTwin consortium and the OBSEA observatory staff, as part of the iMagine project, for sharing domain workflows and datasets that motivated the demonstrators. GM would like to thank Prof. Kate Keahey for hosting the research stay and for her mentorship during the exchange.

Conflict of interest

The author(s) declared that this work was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declared that generative AI was not used in the creation of this manuscript.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

References

Caballer, M., Antonacci, M., Šustr, Z., Perniola, M., and Moltó, G. (2021). Deployment of elastic virtual hybrid clusters across cloud sites. *J. Grid Comput.* 19, 4. doi:10.1007/s10723-021-09543-5

Caballer, M., Moltó, G., Calatrava, A., and Blanquer, I. (2023). Infrastructure manager: a tosca-based orchestrator for the computing continuum. *J. Grid Comput.* 21, 51. doi:10.1007/s10723-023-09686-7

DISCOVER-US Consortium (2025). Distributed computing and swarm intelligence – developing a vision for transatlantic collaboration. *Zenodo.* doi:10.5281/zenodo.14940197

Keahey, K., Anderson, J., Zhen, Z., Riteau, P., Ruth, P., Stanzione, D., et al. (2020). "Lessons learned from the Chameleon testbed," in *Proceedings of the 2020 USENIX annual technical conference (USENIX ATC '20)* (United States: USENIX Association). doi:10.5555/3489146.3489161

Perez, A., Risco, S., Naranjo, D. M., Caballer, M., and Moltó, G. (2019). "On-premises serverless computing for event-driven data processing applications," in *2019 IEEE 12th international conference on cloud computing (CLOUD)*, 414–421. doi:10.1109/CLOUD.2019.00073

Risco, S., Moltó, G., Naranjo, D. M., and Blanquer, I. (2021). Serverless workflows for containerised applications in the cloud continuum. *J. Grid Comput.* 19, 30. doi:10.1007/s10723-021-09570-2

Risco, S., Alarcón, C., Langarita, S., Caballer, M., and Moltó, G. (2024). Rescheduling serverless workloads across the cloud-to-edge continuum. *Future Gener. Comput. Syst.* 153, 457–466. doi:10.1016/j.future.2023.12.015

Tromp, W., Winsemius, H., Eilander, D., Weerts, A., and Backeberg, B. (2024). "Workflow composition for compound flooding events and adaptation measures," in EGU general assembly 2024 *(vienna, Austria and online)*. doi:10.5194/egusphere-egu24-11859