OpenCloud

Simplified and stable integrations for modern IT

Integration

https://opencloud.eu

Inhalt

03 OpenCloud: Simplified and stable integrations for modern IT

04 The limitations of traditional integration models

05 The challenges of integration in day-to-day business

06 Architecture: OpenCloud's approach to integration

10 Practical advantages for integration and operation

11 Overview: How OpenCloud simplifies integrations

12 The result: digital sovereignty

OpenCloud: Simplified and stable integrations for modern IT

In most IT environments, integrations do not arise as part of a clear architectural framework, but rather in response to new requirements. New applications or devices require new connections, leading to a collection of numerous ad-hoc solutions accumulating over time in many organisations. Each component brings its own specific requirements regarding formats, storage or environments, and each integration is developed as a standalone project.

As a result, an environment gradually emerges that is complex, time-consuming and difficult to manage. Ad-hoc integrations increase security risks, and even minor changes to an application can have unexpected consequences for other systems because the integration paths are closely interlinked.

OpenCloud takes a different approach. The platform simplifies integrations by aligning all applications with a common POSIX-based file tree, providing a clearly structured API layer and enabling federated collaboration via open protocols. Its open-source foundation ensures full transparency and control over all data flows. As a result, integrations remain reliable and can be easily adapted to new requirements at any time.

OpenCloud is based on a fork of the open source software ‘ownCloud Infinite Scale’ (OCIS), whose components were co-developed by developers from the science organisation CERN and other active contributors. OpenCloud is now being further developed by the Heinlein Group with new ideas and a clear focus on data protection, interoperability and sustainable digitalisation.

The limitations of traditional integration models

Integrations are one of the most challenging tasks in modern IT environments nowadays. Applications, analytics tools, AI modules and devices access different data sources, use their own formats and each come with specific requirements regarding authentication, versioning and the operating environment. A separate integration is created for each of these components, which is developed and maintained individually.

In practice, this involves a great deal of effort: teams search for suitable APIs, develop their own interface modules, prepare data for exchange, set up authentication methods and continuously check versions and compatibilities. As soon as an application receives an update, unexpected side effects can occur that affect other systems. This results in a complex web of dependencies that appears difficult to control and poses significant operational risks.

These traditional integration approaches have limited scalability. With every new application, maintenance costs rise, whilst time and security requirements increase simultaneously. Organisations are investing ever more resources in stabilising integrations rather than being able to utilise new functions or services. The effort increases linearly, but the complexity often grows exponentially.

The challenges of integration in day-to-day business

Many IT environments present a range of recurring integration issues. The following issues arise in virtually all heterogeneous environments – regardless of their size or the technology used:

1. **Custom development costs:**
Each new integration requires its own interface design, additional code and ongoing maintenance.

2. **Compatibility issues:**
Differences in software, hardware and version levels make consistent integration difficult.

3. **Points of failure between applications:**
Different data formats, storage structures and access methods lead to fragile handover points.

4. **Cascading updates:**
Changes to one application affect other systems and create unpredictable dependencies.

5. **Lack of standardisation:**
Different data formats, storage structures and access methods lead to fragile interfaces and prevent consistent procedures.

6. **High maintenance costs:**
Integrations need to be continuously adapted, tested and stabilised.

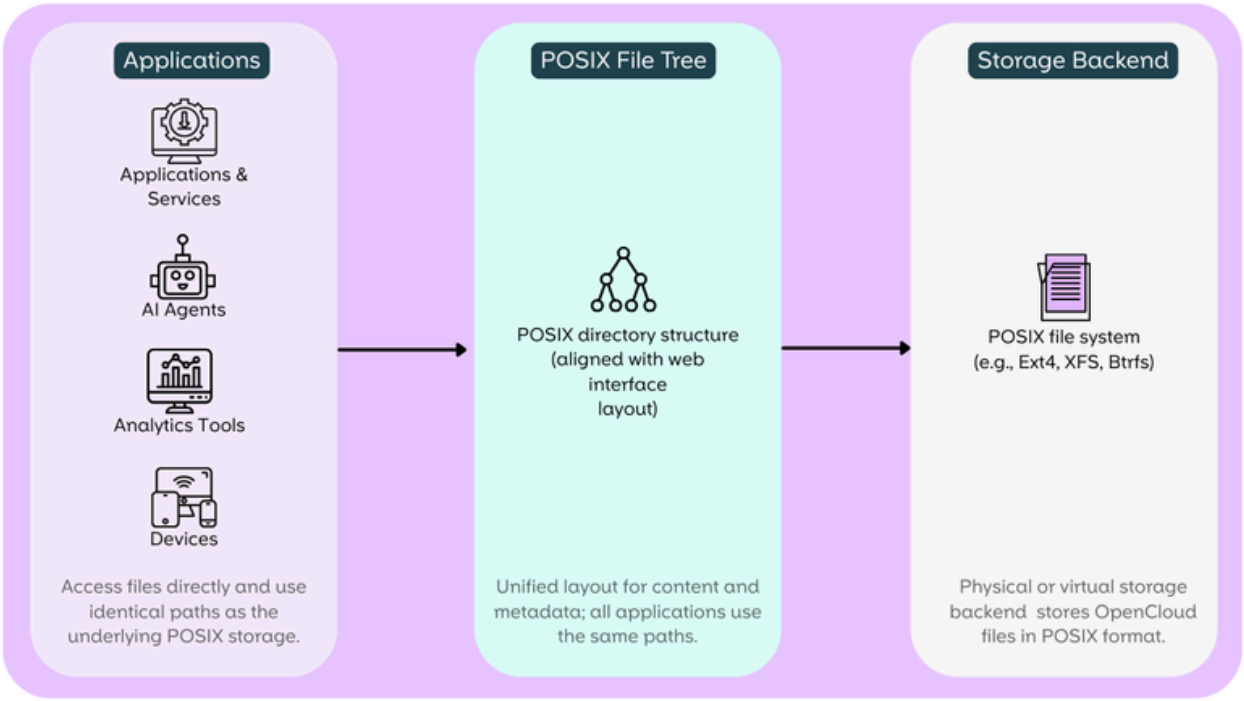
7. **Security risks:**
Ad-hoc solutions and isolated integrations create additional entry points into the system that cannot be reliably secured.

Architecture: OpenCloud's approach to integration

To ensure that integrations are stable, traceable and maintainable in the long term, OpenCloud is based on three fundamental technical principles: a common POSIX foundation, clearly defined APIs and federated instances. Together, these enable an integration model that does not require data duplication, dedicated integration modules or tight coupling. The following sections explain how this architecture works.

POSIX-based integration

One of OpenCloud's key strengths is its consistent focus on POSIX as a common technical framework. Rather than moving, copying or converting data into proprietary formats between systems, all applications and devices access the same POSIX file tree. The files exist only once, and every component works with the same paths, permissions and metadata – regardless of whether they are analytics tools, AI agents, mobile devices or desktop applications.



POSIX-based integration: Applications access the same file tree directly

Architecture: OpenCloud's approach to integration

Using this shared file system avoids a major source of integration errors. Data does not need to be transferred between systems or converted into other formats. Applications see the files as if they were stored locally on their own system, whilst OpenCloud reliably provides and secures the underlying storage backend.

POSIX provides a stable framework: the standard has been established for decades, is widely implemented and well documented. This reliability enables an integration architecture that requires no custom logic and proves robust, transparent and easily extensible in day-to-day operations. OpenCloud extends this foundation with fine-grained permission models and access rules, enabling services, individuals and teams alike to utilise the same data set.

API-first approach

OpenCloud consistently follows an API-first approach to integration. All core functions are available via a clearly defined and fully documented REST API. The libre-graph interface adheres to an OpenAPI specification, thereby creating a unified integration model that remains stable, predictable and easy to test – regardless of the frameworks or programming languages used.

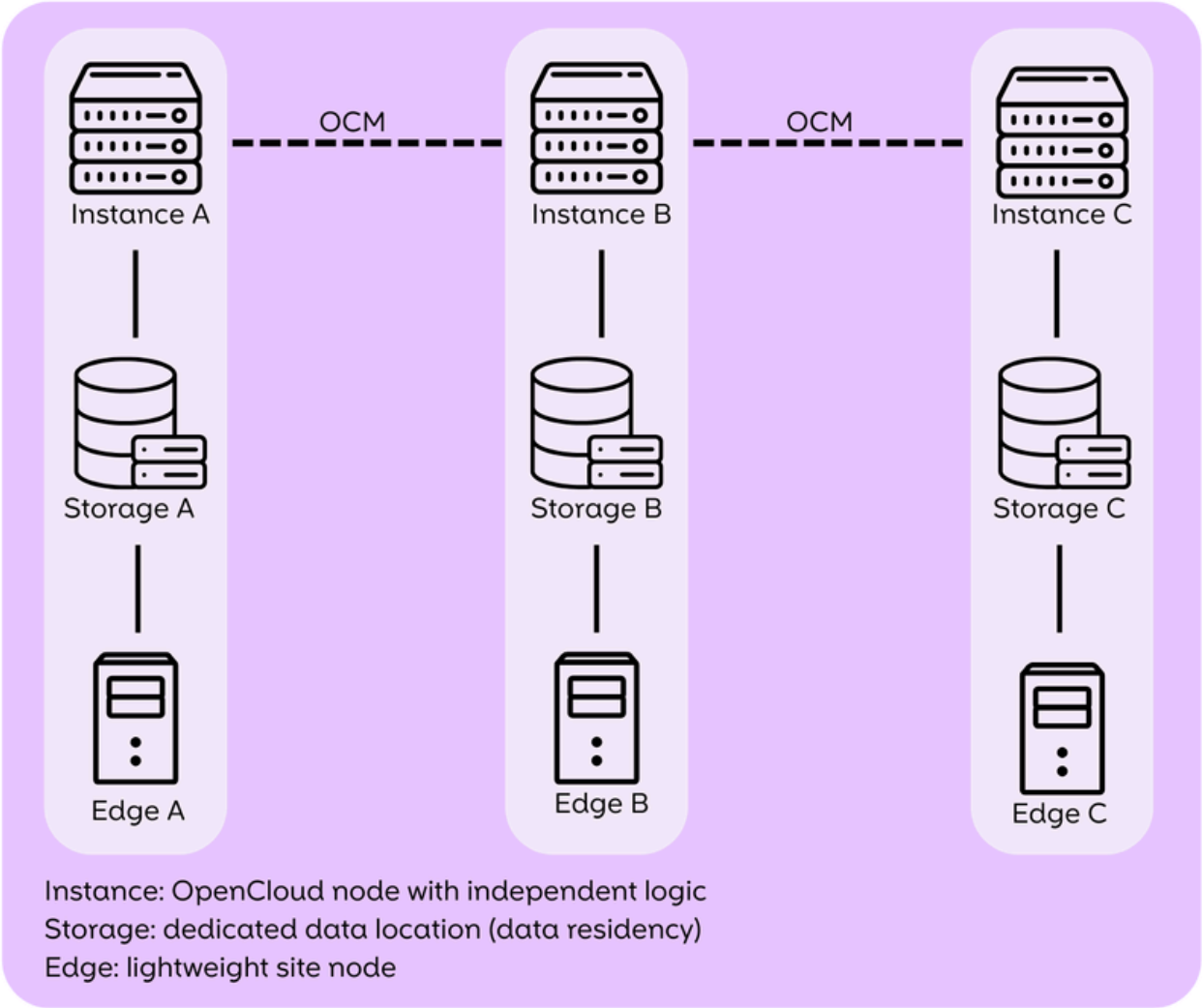
Based on this, a dedicated Web Extensions framework has been developed. Extensions use the same APIs, run in isolation and can be integrated into the platform in a controlled manner. New features can be added without disrupting existing workflows.

In addition to direct POSIX access, OpenCloud also supports uploads via the platform. Files undergo defined validation and processing steps, including metadata management and access control checks. This results in consistent, traceable upload processes that require no additional implementation and enhance security.

Architecture: OpenCloud's approach to integration

OpenCloudMesh, federation and edge deployment

OpenCloud supports the federation of instances via the OpenCloudMesh (OCM) protocol (in accordance with IETF Draft 04). This allows multiple independent instances to be connected to one another: across regions, locations or organisational boundaries. Each instance retains its own storage and permissions structure, whilst the connections enable secure data exchange and the sharing of content.



Architecture: OpenCloud's approach to integration

The architecture is particularly well suited to distributed locations and so-called edge deployments – that is, lightweight OpenCloud nodes operated close to users or data sources. Such instances require only minimal resources and can be deployed quickly without placing a burden on existing systems. This significantly reduces infrastructure costs and entry barriers: new nodes can be integrated with little effort, and organisations gain flexible, scalable operating models – from campus clouds and branch offices to decentralised government structures.

A key advantage of the federation remains control over the storage location: each instance defines for itself where data is permanently stored and processed. This data residency control not only meets regulatory requirements but also supports strategic goals in the area of digital sovereignty.

Practical advantages for integration and operation

OpenCloud significantly reduces the effort required for integration. Applications either write directly to the shared file tree via POSIX or upload files via OpenCloud. Both methods ensure stable, consistent integrations without the need for custom intermediary logic, without additional points of failure, and without custom-developed integration code. This speeds up projects, reduces costs and minimises long-term maintenance efforts.

Uploading via OpenCloud offers additional security and operational functions that would otherwise need to be implemented and maintained separately. Files undergo server-side security and processing procedures: comprehensive auditing, ICAP-based malware and virus scanning, automatic metadata maintenance, full-text indexing, access protection via MFA, encryption, and transparent logging of all operations. Applications can therefore operate securely and transparently without having to implement their own security or validation logic.

The integration layer remains stable and version-independent. Updates do not trigger chain reactions, and scaling works both centrally and in a distributed manner. This reduces operational risks and creates reliable operating conditions.

Overview: How OpenCloud simplifies integrations

Function	Benefits for integrations
POSIX-based Integration	A shared tree structure with no duplicates or transformations, and consistent paths for all applications
API-first architecture	Predictable, documented integration points; robust extensibility via Web Extensions
Upload via OpenCloud	Malware scanning (ICAP), auditing, logging, metadata management, indexing, MFA, encryption
Federation (OCM)	Connect instances, keep data local (data residency), work across locations
Stability and version independence	Updates to individual applications do not trigger any knock-on effects in other systems
Scaling & Edge Support	Scalable growth in centralised, distributed or edge environments
Reduced complexity	Less integration code, fewer sources of error, lower maintenance costs

The result: digital sovereignty

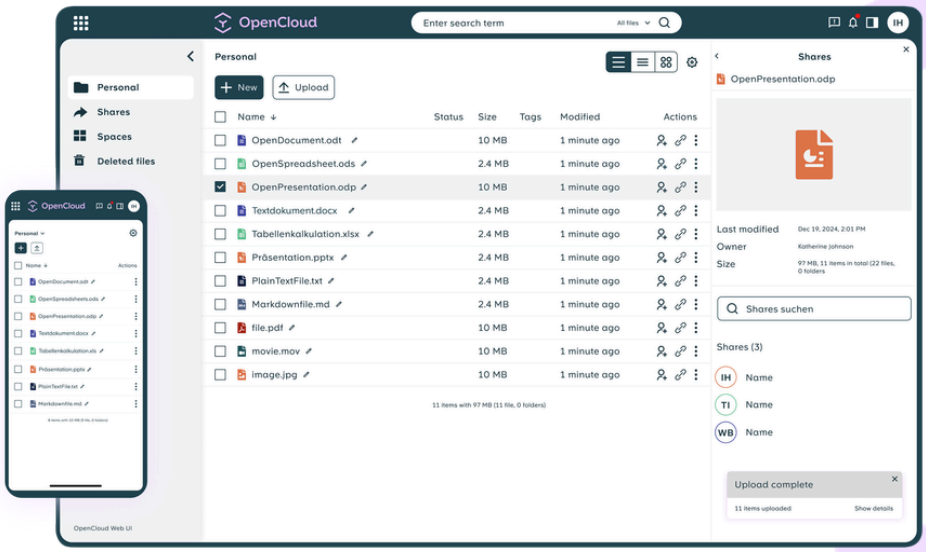
The combination of POSIX-based integration, open APIs and federated instances creates an architecture that operates without proprietary dependencies. Applications run on a shared file system, integrations follow clearly documented interfaces, and data remains in defined locations. This enables organisations to retain control over their data and infrastructure.

For organisations that prioritise digital sovereignty, this approach is crucial: they have complete control over the choice of storage locations, access methods and extensions. New services can be integrated without having to modify existing systems or develop complex integration logic. This reduces effort, shortens project timelines and avoids long-term dependencies.

OpenCloud demonstrates that modern integrations do not have to be complicated. Open standards, transparent architectural principles and a consistent operating model result in a platform that remains stable, scalable and independent – thereby creating precisely the framework that organisations need today.

Would you like to simplify integrations and reduce costs? We can support you with architecture, planning and implementation.

Get in contact with us at sales@opencloud.eu.
We look forward to hearing from you.



OpenCloud

https://opencloud.eu