OpenCloud

Maximum performance with minimum energy requirement

Performance

https://opencloud.eu

Table of contents

OpenCloud: Maximum performance with minimum energy requirement

Slow software costs time and electricity. Powerful IT is not a luxury, but a prerequisite for smooth operations. However, many platforms slow themselves down: interpreter-based applications and complex database structures generate unnecessary overhead, increase energy consumption and worsen response times. The result is sluggish interfaces, dissatisfied users and rising costs for cooling and hardware in the data centre.

OpenCloud takes a different approach. The platform is developed entirely in Go – a compiled language that does not require an interpreter and runs directly on the machine. This eliminates the typical runtime and query delays that occur with PHP or SQL-based systems. Combined with a database-free architecture and clearly separated microservices, the result is a system that uses resources efficiently and responds quickly at all times – even under high load.

The result: noticeably shorter response times, lower energy consumption and a platform that grows with its tasks instead of slowing them down. For IT managers, this means predictable performance and sustainable operation – for users, it simply means smooth working.

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.

When platforms consume energy instead of delivering performance

Many file management systems quickly reach their limits as usage increases. What still works in small installations leads to delays, increased energy consumption and growing operating costs in large environments.

The causes usually come down to the tech base:

Interpreter instead of machine code:

- PHP-based systems reinterpret the code every time it's called. This takes up computing time, ties up CPU resources, and makes response times unpredictable.

Complex tech stacks:

- Multiple layers of web servers, interpreters, databases, and caches increase dependencies and maintenance costs. Each additional component consumes energy – and each interface can become a bottleneck.

SQL databases as a bottleneck:

- Metadata and file information must be permanently managed in relational databases. The more users and files there are, the slower queries, backups and synchronisations become.

High resource requirements:

- To mitigate performance issues, caching systems, load balancers and similar solutions are often retrofitted. This increases energy and cooling requirements – and makes operation even more complex.

For IT departments, this means more maintenance, more hardware, more energy consumption, but no sustainable performance improvement.

OpenCloud – Performance due to clear architecture

High performance is no coincidence, but rather the result of a clear technical decision. OpenCloud was developed to use computing power efficiently and distribute load evenly – without the typical bottlenecks of conventional platforms.

The performance of OpenCloud stems from the interaction of a clearly defined, resource-efficient system architecture:

Short data paths
All microservices are compiled in Go and run directly on the machine. This avoids runtime overhead and reduces the energy requirement per request.

Asynchronous communication
Services exchange information in parallel via the internal event broker NATS. Processes do not block each other, and the system remains responsive even under load.

Parallel file operations
Uploads, scans and versioning run in parallel so that IO-intensive processes do not slow each other down.

Targeted resource utilisation
Each service runs in isolation in a container and can be scaled individually depending on the load.

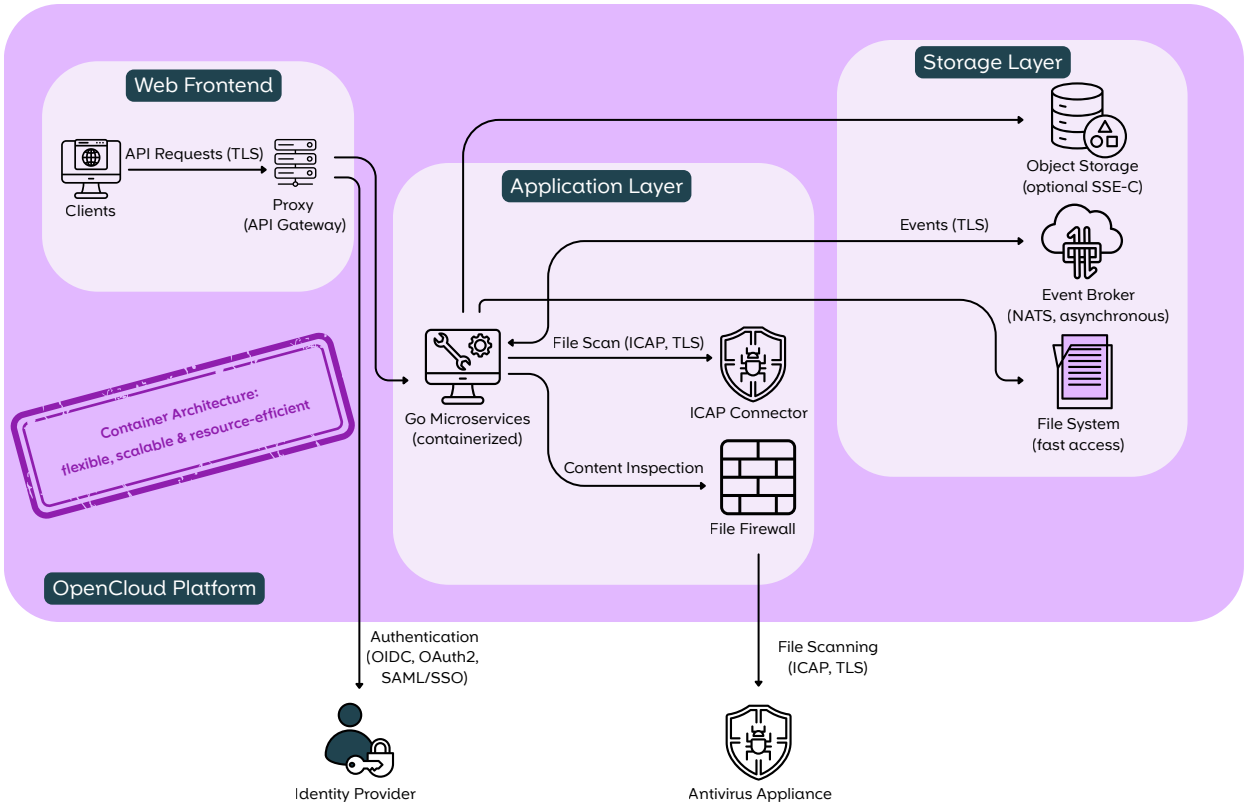
TLS-encrypted communication
All data flows between services are encrypted throughout. The architecture ensures that security and performance are not mutually exclusive – encryption takes place without any noticeable latency losses.

Direct memory access
Metadata is not stored in an SQL database, but in the file system. This speeds up queries and eliminates the need for transactions.

Linear scaling
New services can be easily added without burdening existing components. This saves hardware, reduces energy consumption and keeps performance constant.

OpenCloud – Performance due to clear architecture

The secure architecture of OpenCloud



OpenCloud's architecture generates performance not through tuning, but through efficiency – with noticeably lower computing load and measurably lower energy consumption.

It combines compiled Go code, containerised microservices and asynchronous communication via NATS. This results in short data paths, minimal latency and high energy efficiency – even with growing loads.

Numbers that convince

Theory is good – measurable results are better. To realistically assess the performance and resource efficiency of OpenCloud, extensive K6 load tests were carried out. K6 is an open-source tool that uses virtual users (VUs) to simulate realistic access scenarios and measures response times, error rates and system utilisation.

The tests were based on dedicated virtual machines with different configurations – from one to eight CPU cores and memory sizes between 2 and 32 GB. At the same time, K6 simulated between 8 and 630 simultaneously active users who triggered typical file operations and API calls.

Response times in milliseconds, error rates in percent and system resource utilisation were recorded – precisely the key figures that determine energy efficiency, stability and user experience in the data centre. The aim was to determine the optimal balance between performance and energy consumption and to test how efficiently OpenCloud works under increasing load.

How OpenCloud behaves under load

The calculations based on these measurements paint a clear picture: OpenCloud scales linearly – with consistent efficiency. Additional users increase the load, but the typical slumps experienced by traditional systems (due to database overhead or interpreter limits, for example) do not occur.

Compared to interpreter-based systems, OpenCloud requires around 30% less energy and computing power for the same user load. The compact architecture avoids overhead from databases and script processing and keeps performance constant even as the number of users grows.

Numbers that convince

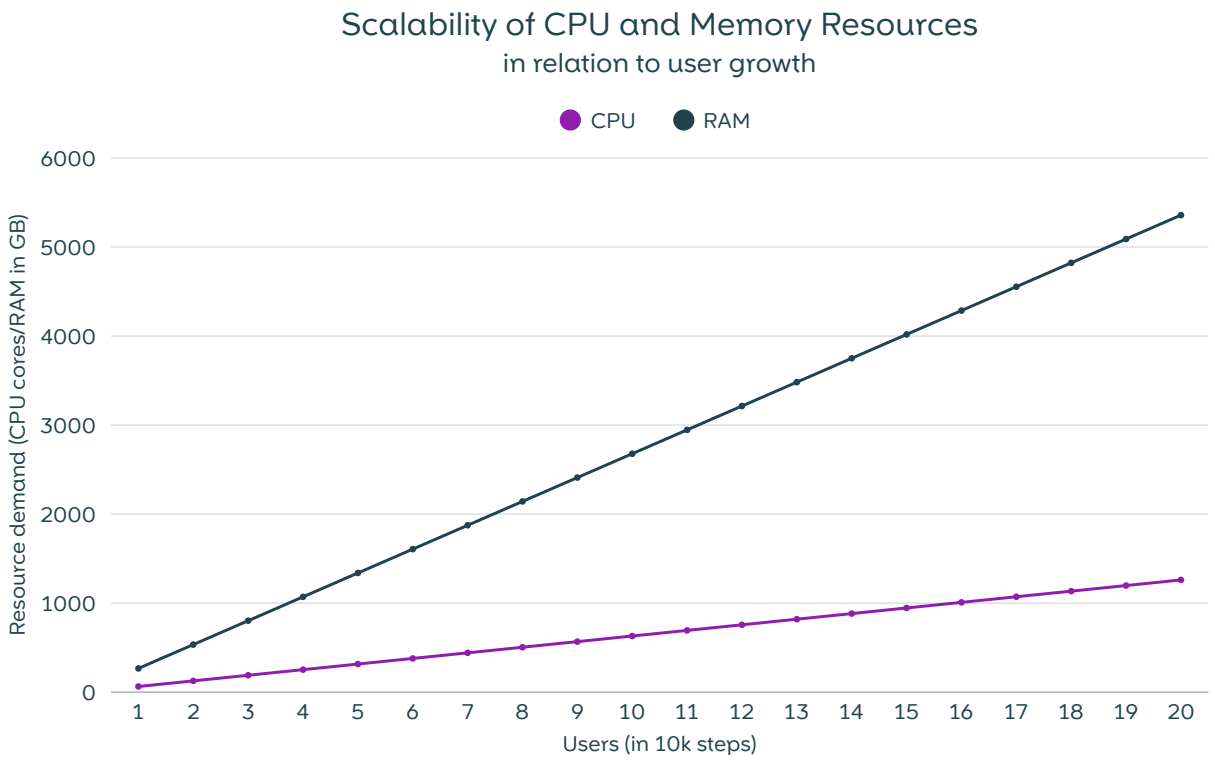


Diagram 1: Linear scaling of CPU and RAM utilisation. The measurements show that OpenCloud uses resources proportionally to the number of users – without any drop in performance or energy efficiency.

Long-term efficient memory usage

In addition to the load tests, the development of memory requirements during typical use was also examined. This was based on a practical scenario involving office workers, students and private individuals who regularly work with common Office files.

The calculations are based on a typical working environment with around 250 working days per year. On average, a user creates around 520 to 720 new files per year – mostly in the kilobyte to lower megabyte range. Word documents have an average size of 2 MB, Excel spreadsheets 1.5 MB and presentations around 8 MB. In addition, there are PDFs and scans with an average size of 2 MB per file. This results in a total annual increase of around 1.1 GB of new files per person.

Numbers that convince

Since OpenCloud versions every change, the storage requirements increase additionally due to multiple versions. Approximately 60% of all files are revised at least once, 30% even ten times or more. In total, this results in an average of 1.86 GB of additional data per year. Together with the new files, this results in an annual storage increase of just under 3 GB per user.

The upload frequency also follows a realistic pattern: two to three files per day, ten to fifteen per week. Peaks occur mainly after meetings, at the end of the month or during project phases.

Predictable storage requirements

The model calculation shows that storage requirements grow continuously but moderately over five years. Even with intensive use and numerous versions, the annual increase remains predictable. Organisations can therefore plan their storage capacities for the long term.

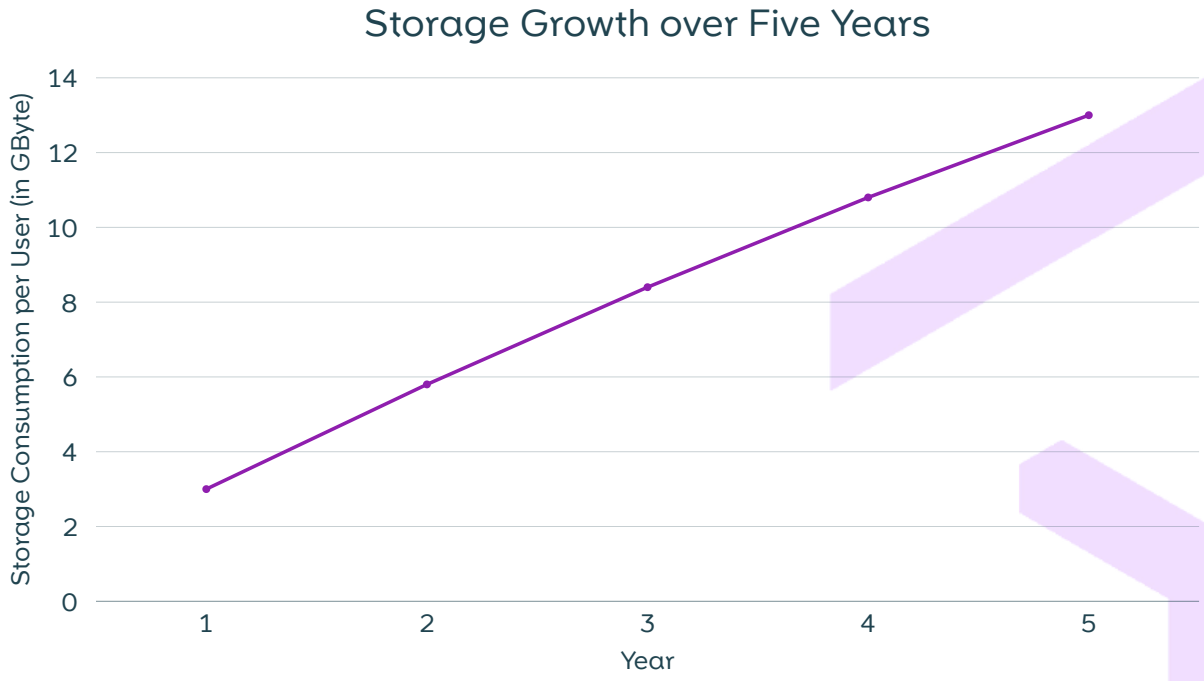


Diagram 2: Storage development per user over five years: Storage requirements are growing continuously, but remain predictable and calculable – an important factor for long-term stability.

Practice-proven: Performance for real workloads

The performance advantages of OpenCloud are evident not only in tests, but also in ongoing operation. Thanks to its compact code compiled in Go, the platform operates with significantly lower energy requirements than traditional web applications. Processes run directly on the machine without interpreter or database overhead, which reduces CPU load and lowers power consumption by around 30% for comparable loads.

Since OpenCloud consists of modular microservices, the infrastructure can be precisely dimensioned. Instead of expanding entire server clusters, only those components that are actually under load can be scaled. This saves hardware and simplifies maintenance and updates – a clear advantage for the total cost of ownership (TCO). Unused services are automatically put into idle mode, which significantly reduces energy consumption even at low utilisation.

The benchmarks also show that OpenCloud already scales linearly with around 40% of users active at the same time. Even when several hundred users are working in parallel, response times and energy consumption remain constant. This allows the capacity of a system to be planned precisely – without overprovisioning, i.e. without permanently unused hardware reserves.

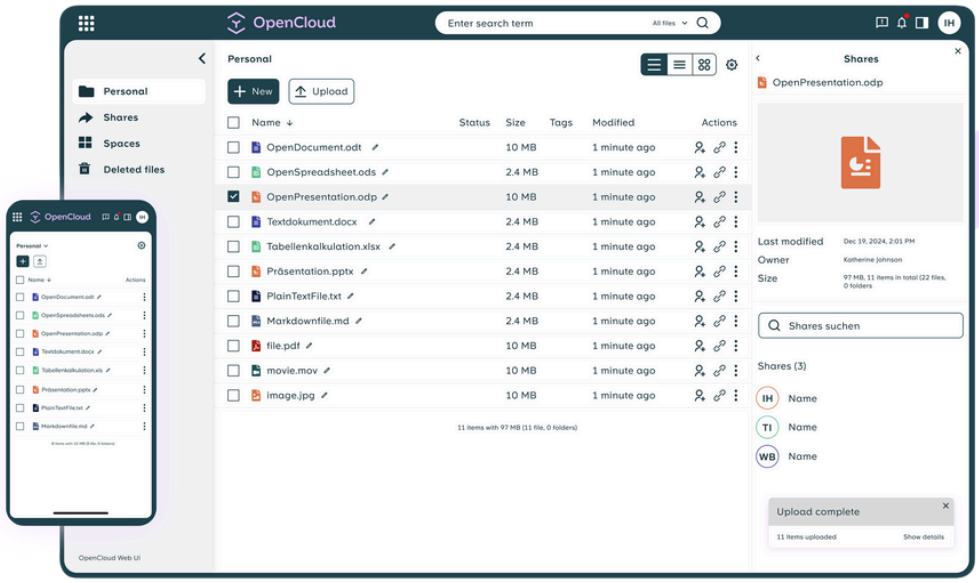
Efficient IT architecture for maximum performance and sustainability

OpenCloud demonstrates that high performance is not a matter of oversized hardware, but rather of well-designed architecture. Compiled code, asynchronous processes and a database-free structure ensure that computing power is delivered where it is needed – without wasting energy.

The results from the benchmarks and projections confirm that OpenCloud scales linearly, remains responsive even under load, and uses resources efficiently. For organisations, this means predictable capacity, lower energy consumption, and lower long-term operating costs.

OpenCloud connects high performance with efficient use of resources – for predictable load distribution, low operating costs and sustainable data centre operation. Talk to us about your scalability and energy efficiency requirements.

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



OpenCloud

https://opencloud.eu