MIKELANGELO

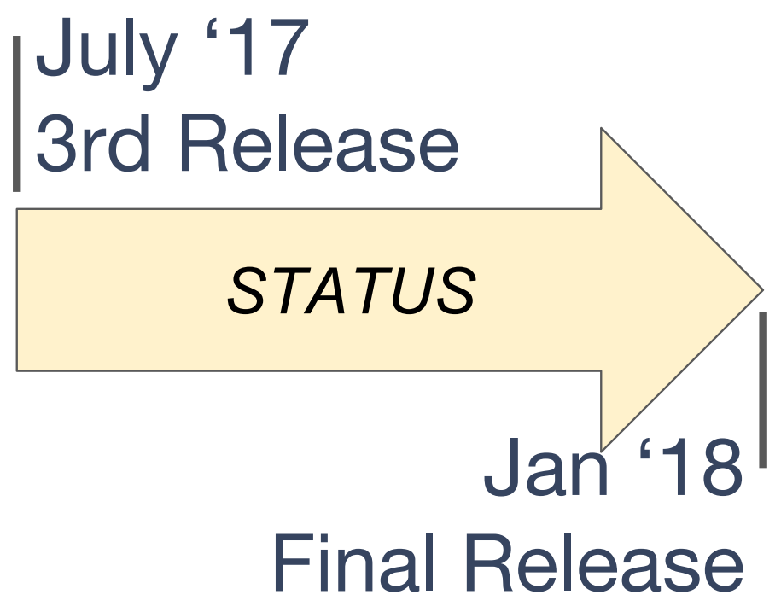
January 2015 - December 2017

Problem Statement

Cloud and HPC architectures are inefficient. We are currently running at the trade-off between efficiency, stability and security, trading between simplicity of the overall application management, their execution capabilities (e.g backwards compatibility, libraries) and the overall security (built-in security vs. secure zones, etc.).

MIKELANGELO Goals

- ➢ Optimise virtual IO, guest OS, hypervisor
- ➢ Speed up in HPC and Cloud
- ➢ Hypervisor security
- ➢ OpenStack compatible
- ➢ Package applications

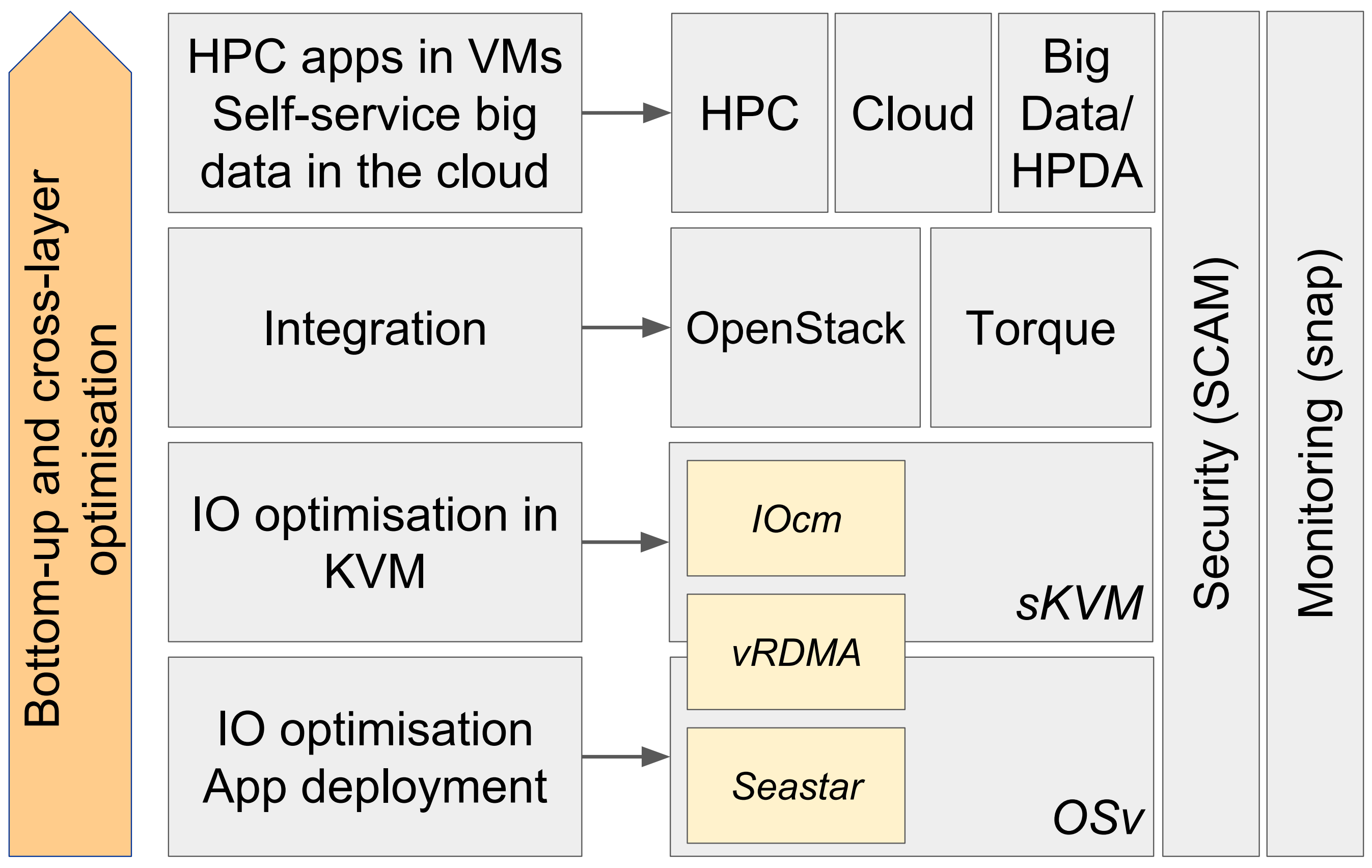


July ‘17
3rd Release

STATUS

Jan ‘18
Final Release

Our Approach



Bottom-up and cross-layer optimisation

HPC apps in VMs Self-service big data in the cloud → HPC | Cloud | Big Data/ HPDA

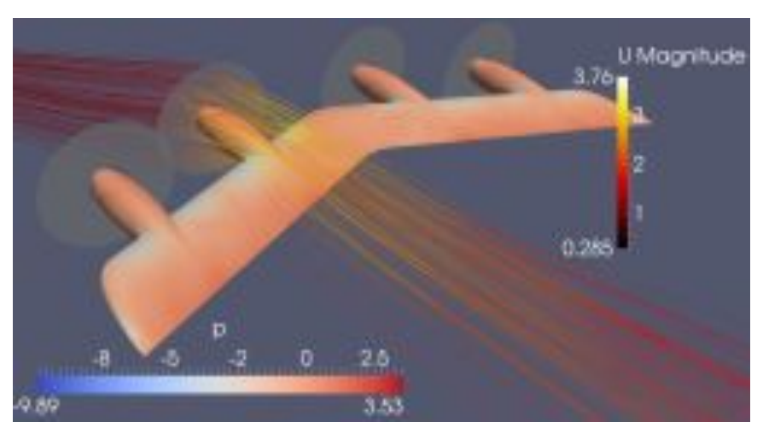
Integration → OpenStack | Torque

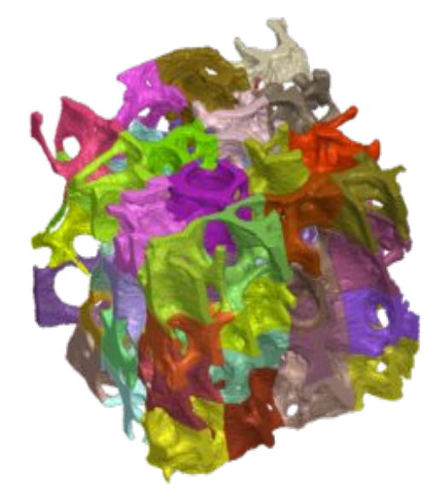
IO optimisation in KVM → IOcm | vRDMA | sKVM

IO optimisation App deployment → Seastar | OSv

Security (SCAM)

Monitoring (snap)





From concept to concrete implementation and evaluation on use cases

Key efficiency improvements are shared between hypervisor (IOcm - IO Core Manager), guest (OSv and Seastar) and between them (vRDMA - Virtual RDMA); As for stability, OSv has been improved and can execute more applications; Security is focused on SCAM (hypervisor level), but extends also to other parts of the stack.

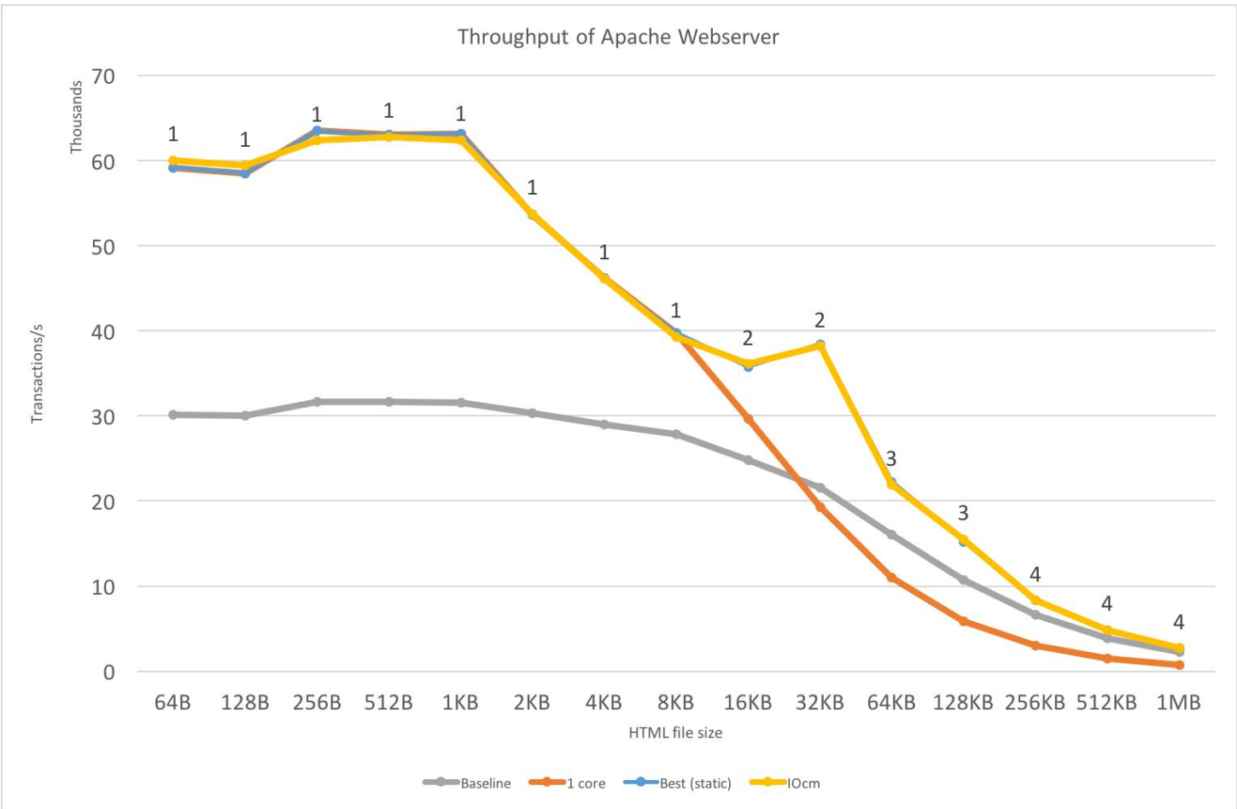
Unikernel App Management

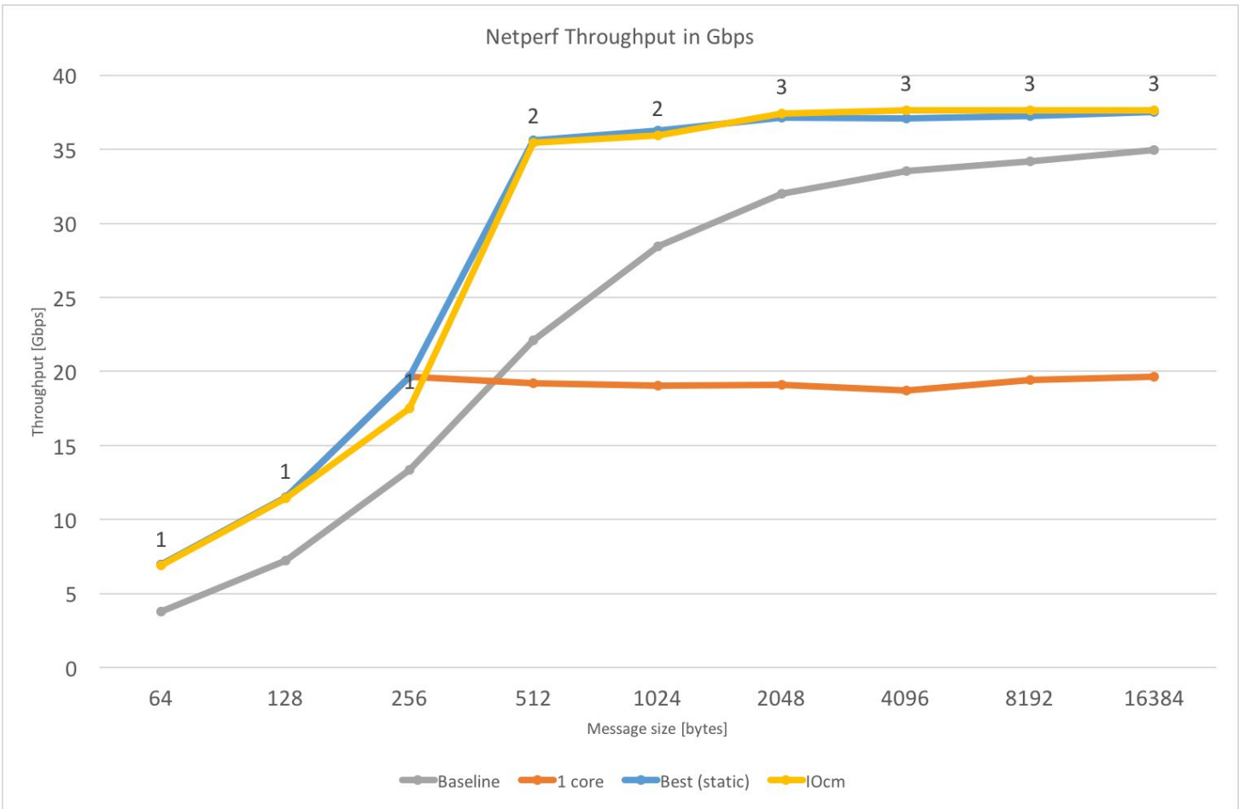
MIKELANGELO utilises lightweight unikernels to wrap workloads into virtual machines with minimal execution and storage overheads. The composition and contextualisation processes are fully integrated with Torque/PBS resource manager. Numerous pre-built open source application packages are freely available. OpenStack, Amazon and Google clouds are supported out of the box.

OpenFOAM® OPEN MPI MySQL OpenJDK hadoop HDFS ERLANG cassandra unik OSv designed for the cloud amazon web services EC2 openstack node js

Hypervisor IO Management

IO core manager dedicates CPU cores adapting for optimal outcomes based on the demand for I/O and compute resources at all times. This results in improved overall throughput of virtualised network stack.

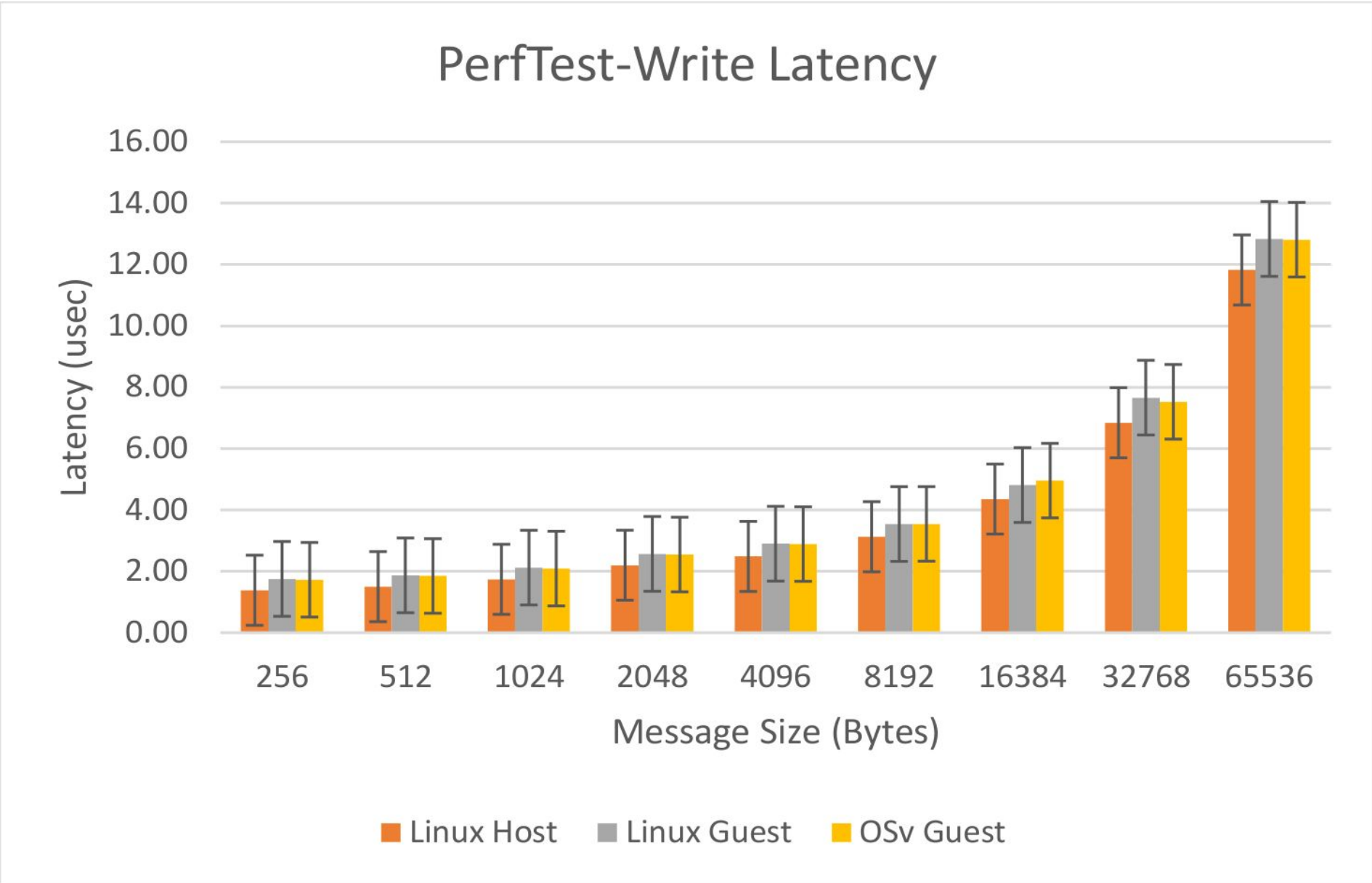


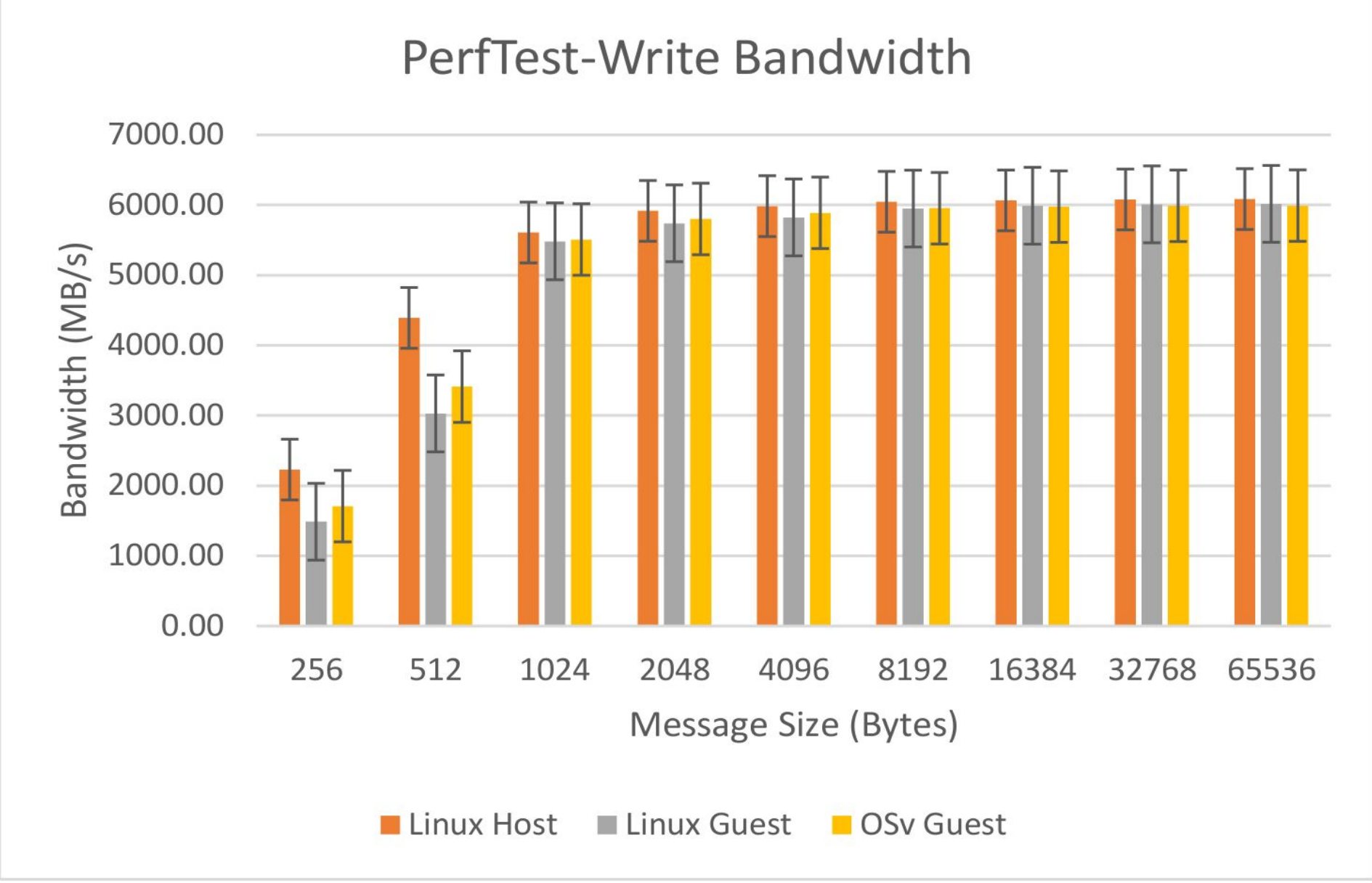


The adaptiveness of the IOcm to the IO load is completely automated. The inbuilt heuristics follows the behavior of the guests and selects the optimal number of CPU and IO dedicated cores, based on IO and CPU needs of the virtualised guests under host.

Virtualised RDMA

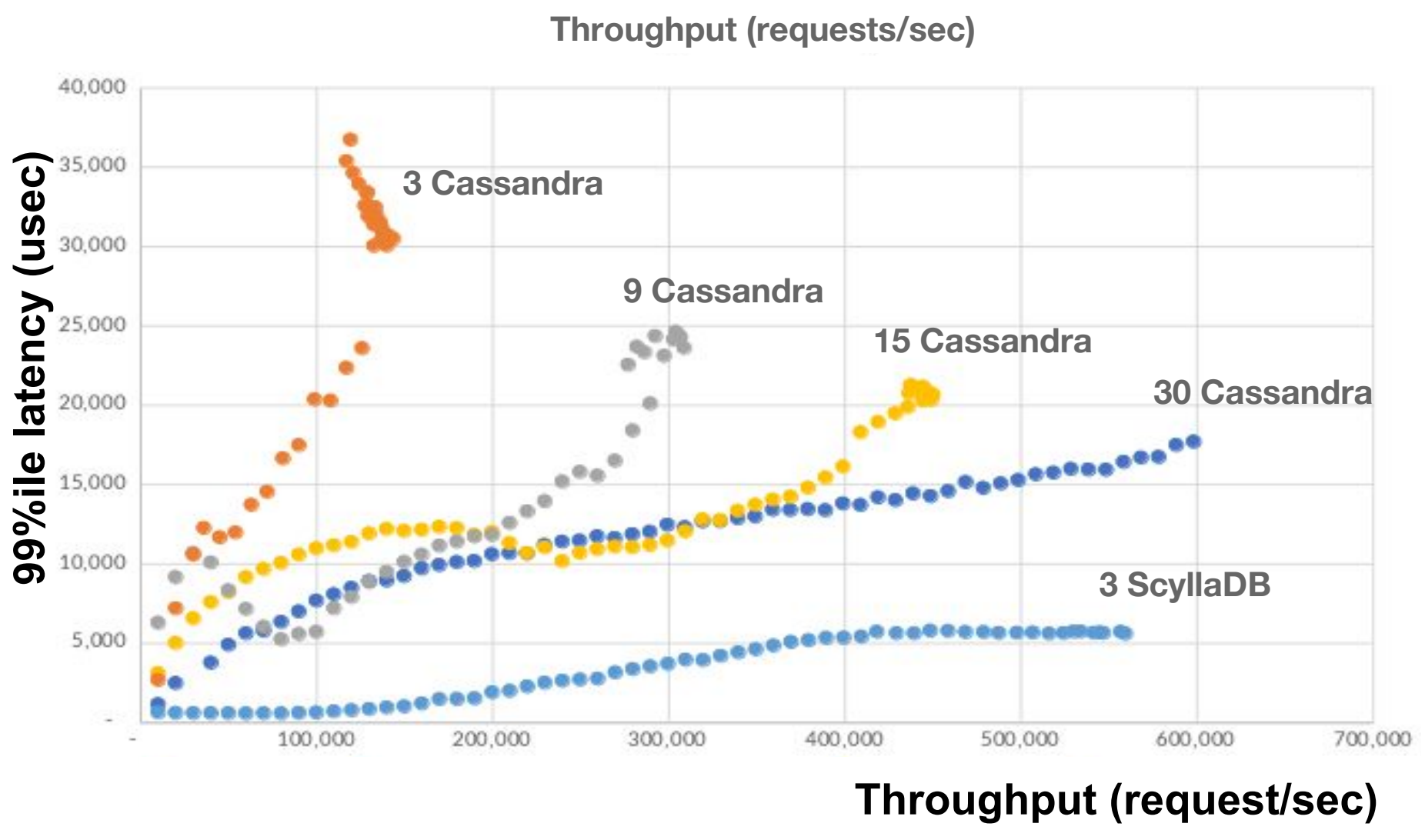
Paravirtualised virtio compliant device driver for Linux and OSv enables unmodified InfiniBand communication.
This results in near-native (maximum 98%) performance while at the same time allowing sharing of the InfiniBand device between multiple virtual machines.
Figures show native throughput and latency, compared with virtualised Linux and OSv guests over different message sizes.

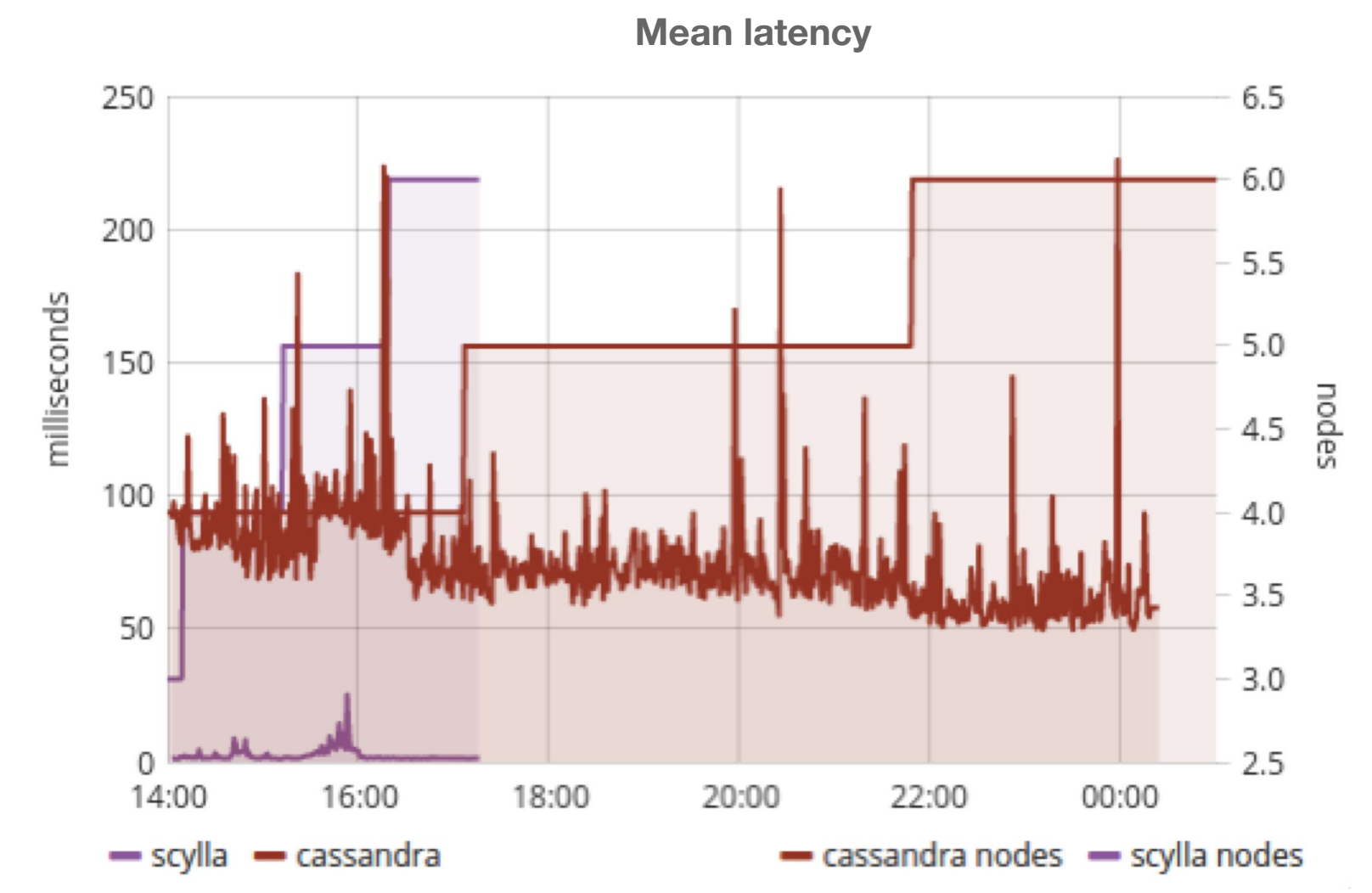




Seastar Asynchronous APIs

The analysis of bottlenecks of several workloads (memcached, Cassandra) has shown that traditional programming interfaces (e.g. POSIX) are preventing significant performance gains. Seastar’s shared-nothing design, with high performance networking, exploits modern infrastructure to the maximum.





Application redesign and rewrite is usually required.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 645402

Project coordinator: *daniel.vladusic@xlab.si*

Technical coordinator: *gregor.berginc@xlab.si*

www.mikelangelo-project.eu

@mikelangelo_eu

Mikelangelo

XLAB NOT IDLE · HLRS · IBM · intel · GWDG Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen · PIPISTREL · SCYLLA · Ben-Gurion University of the Negev · HUAWEI